# DYNAMIC MODELLING AND SIMULATION OF A POWER GAS TURBINE USING ARTIFICIAL NEURAL NETWORK; A COMPARATIVE STUDY

**BY**

**ATTAMAH CHIKAODILI STEPHNIE (B.Eng.)**

**20174076438**

**A THESIS SUBMITTED TO THE DEPARTMENT OF CHEMICAL ENGINEERING, SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY FEDERAL UNIVERSITY OF TECHNOLOGY OWERRI,**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF MASTER OF ENGINEERING (M.ENG) DEGREE IN CHEMIAL ENGINEERING**

**DECEMBER, 2022**

## CERTIFICATION

This is to certify that this research work "***Dynamic Modelling And Simulation Of a Power Gas Turbine Using Artificial Neural Network; a Comparative Study***" was carried out by **Attamah Chikaodili Stephnie**, with registration number **20174076438**. In partial fulfilment of the requirements for the award of Masters of Engineering (M.Eng.) degree in Chemical Engineering, Federal University of Technology Owerri.

| | |
|---|---|
| **Engr. Dr. E. C. Osoka**<br>**(Project Supervisor)** | 08-02-2023<br>**Date** |
| **Engr. Dr. I. S Ike**<br>**(Co Supervisor)** | 08-02-2023<br>**Date** |
| **Dr. C. I. O. Kamalu**<br>**(Head of Department)** | 9/2/2023<br>**Date** |
| **Engr. Prof. J. C. Ezeh**<br>**(Dean, SEET)** | 23/03/2023<br>**Date** |
| **Prof. B. O. Esonu**<br>**(Dean, PGS.)** | **Date** |
| **Engr. Prof. O. D. Onwukwuli**<br>**(External Examiner)** | 30/3/2023<br>**Date** |

## DEDICATION

This project is dedicated to the sovereign God who gave me unrivalled, unconditional emotional, and financial support.

## ACKNOWLEDGEMENT

My appreciation goes to my supervisor, Dr. E.C. Osoka and Engr. Dr. I.S. Ike my co-supervisor for their painstaking effort (guidance and correction) to the project. My heartfelt appreciation goes to the lecturers of the Department of Chemical Engineering for the transfer of knowledge and experience they whole heartedly released on me during the phase of my course work. I will also extend my gratitude to the Head of Department Dr. I.S. Ike for his leadership charisma. Furthermore I will not fail to be grateful to the Dean of School of Engineering and Engineering Technology, Engr Prof. J.C. Ezeh, and the Dean of Postgraduate School, Prof. C.C. Eze. I also extend my gratitude to Engr. Opeyemi Onarinde (Head of Control Engineering, Egbin Power Plant), Engr. Obah Izuchukwu (First Independent Power Limited, Omoku) and Engr. Samuel Duruibe (FIPL, Afam) for their support in retrieval of research data from Afam Power Plant. Finally I would not fail to appreciate my beloved parents Mr. Attamah Christopher and Mrs. Attamah Mary.

## TABLE OF CONTENTS

## LIST OF TABLES

**LIST OF FIGURES**

**NOMENCLATURE**

| | |
|---|---|
| ANN | Artificial neural network |
| APP | Afam power plant |
| BPP | back propagation |
| CCGT | Closed circuit/cycle gas turbine |
| CCGT | Combined cycle gas turbine |
| CovNets | Convolution neural networks |
| CHP | Combined heat and power |
| FIPL | First independent power limited |
| GT | Gas turbine |
| IEEE | Institute of electrical and electronics engineers |
| IIOT | Industrial internet of things |
| IPGT | Industrial power gas turbine |
| LAN | Local area network |
| LayRecNet | Layer recurrent neural network |
| LHV | Lower heating value |
| MAPE | Mean of absolute percent error |

| | |
|---|---|
| MLP | Multi layer perceptron |
| MSE | Mean squared error |
| n.d | No date |
| NAIC | Normalized akaike information criterion |
| NARX | Non linear autoregressive input |
| NBIC | Normalized bayesian information criterion |
| OCGT | Open cycle/ciruit gas turbine |
| OEM | Original equipment manufacturer |
| OGT | Omoku gas turbine |
| OPC | Open process control |
| PLC | Programmable logic controller |
| RNN | Recurrent neural network |
| RTU | Remote terminal unit |
| SCADA | Supervisory control and data acquisition |
| SFC | Static frequency converter |
| SSD | Static starting device |
| T-MATS | Toolbox for the modelling and analysis of thermodynamic systems |

## ABSTRACT

The dynamic modelling and simulation of a power gas turbine by comparing three forms of Artificial Neural Network was adopted given the complexities of the physics and mathematical based turbine models. Layer Recurrent Neural Network (Layrecnet), Feed Forward Back Propagation (FF BPP) Network and Non Linear Autoregressive Network with Exogenous Input (NARX) were selected for the dynamic modelling of the turbine. The start-up data was trained with these networks and Multiple Input Multiple Output (MIMO) and Multiple Input Single Output (MISO) models were developed for the machine. Furthermore; the selected models were validated with operational data from the turbine similar in manouver to the data adopted for modelling. It is observed that "Layrecnet" has the least Mean Squared Error (MSE) of 1.12 and Mean of Absolute Percentage Error (MAPE) of 0.7310 in the MIMO model while "FF BPP" network comes a close second with MSE of 1.74 and MAPE of 1.4249. "LayRecNet" MIMO and MISO models were used to simulate the start-up of the gas turbine because it ranked the highest among the three networks with the use of MSE and MAPE error performance metrics. However; the "FF BPP" network also performed well as it had the best performance for the Turbine Outlet Temperature MISO model with MSE of 0.296 and MAPE of 0.495. The research showed that the "Layrecnet" Network is a better tool for dynamic time series modelling as the network had the least MSE and MAPE with FF BPP coming a close second, while the much acclaimed NARX Network is the least performing network. It was shown that neural networks can be considered a reliable alternative to conventional mathematical driven techniques. Therefore, by using the developed tool, an optimization of the plant operation and maintenance is rendered possible.



***Keywords:*** gas turbine, neural networks, blackbox models, start-up phase, Load, simulation, Plant, manouver

# CHAPTER ONE

# INTRODUCTION

## 1.1 Background Information

Today the most widely used and flexible turbo machinery is the gas turbine. Different process in the engineering industry adopts the use of the gas turbine such as the domestic, small scale, oil and gas, aviation, and the process plant industries (Soares, 2014).The crucial role of gas turbine in the energy industry has motivated researchers to investigate new techniques in order to be able to prognosticate dynamic behaviour of these complex systems as reliably as possible. For these power generating machines to be precisely modelled using the physics based approach confidential Original Equipment Manufacturers (OEM) data such as information on the Internal Guide Vanes(IGV) control and bleed flows has to be released by their OEM's which in most cases are usually not accessible. Today, historical analysis is carried out by collecting a sizable but useful number of parameters which are concurrently stored in databases. However, a thorough evaluation of this data demands immense resources and it can be affirmed that the usefulness of the storage are not yet completely exploited. An exhaustive utilization of the opportunities/capabilities provided by the historical records of the system behaviour could be achieved with the use of a data driven, non-parametric modelling approach which exploits the measured data to a more effective use (Fast, 2010).

Gas turbine are subject to highly non-linear flow conditions which makes it problematic to obtain underlying information for physics based modelling. Physics and thermodynamic equations make up the bulk of most heat and mass balance programs in most simulation tools (Olausson, 2003). However, in order to monitor the condition of the plant and locate faults, long man-hours and a remarkable level of expertise are required. An alternative approach that could be adopted is the use of the so-called black box models which fundamentally

makes use of the process measurement in order to engineer models of the system. One overriding benefit of black box modelling to the physics based approach is that the model can be formulated in a time saving and economical manner and its preciseness can be validated, explicitly juxtaposed against the measured data (Fast, 2010). Therefore a more efficient and easy to use simulation tools and condition monitoring systems are indispensable for use in modern power plants.

Today a wide range of open source artificial intelligence (AI) exist (Russell and Norvig, 1995) and, artificial neural network (ANN) is a the popular AI based algorithms which could be utilized for gas turbine modelling and simulation. ANN is an information driven computer algorithm which is able to learn behavioural patterns through adoption of the right training, i.e. They are adaptable making them well-suited for modelling nonlinear and complex operation of real physical plants. This is an alternative to the conventional simulation and condition monitoring tools as it has grown to become a contemporary, reliable, and low cost IT–based intelligent tools (Haykin, 1999).

Until a new technology is proven to be reliable and trustworthy it will always be received with a certain amount of hesitation. Moreover, the end users of these black box models are plant supervisors and operators which typically have only a lean grasp of the gas turbine design. The usability of this ANN model could be demonstrated to plant operators through the development of a graphical user interface (GUI). Due to the gaps in the modelling and simulation of the heavy duty gas turbines there is a need to carry out this research on dynamic modelling and simulation of a power gas turbine using Artificial Neural Network; a comparative study. A comparative study with Non Linear Autoregressive network with exogenous Input, Layer Recurrent Network and Feed forward back propagation networks will be made and the model with the best performance will be used to simulate the heavy duty

power gas turbine so that a good tool for online/offline system identification and sensor calibration could be developed and deployed.

## 1.2 Statement of Problem

The (heavy duty) gas turbine is capable of operating uninhibitedly over prolonged periods between its usage to the scheduled/corrective maintenance period. However; the overall life cycle of a gas turbine from conception, design, production and followed by its usage and maintenance is inundated with complexities. Incessant and unplanned shutdowns have plagued Nigeria's turbine plants due to ineffective maintenance culture and absence of condition monitoring tools. The Overall Equipment Efficiency (OEE) and operability of moving/rotating equipment deteriorate gradually with respect to time. Real and ideal thermodynamic correlations of the turbine components are prone to lose their credibility after a prolonged period of use and, are therefore subject to major modification. The GT13E2( formerly Alston) turbine of Afam power plant lacks developed models for condition monitoring and if its maintenance culture persist, downtime will remain an upward trend thereby throwing the consumers of this service into prolonged darkness and subsequently leading to loss of revenue from the owners of these plants. There has been a glut of literature developed to depict the steady state model of gas turbine using both the physics based and black box approach but very few works has been developed on the start-up or transient phase of the gas turbines which is key to finding faults that may go undetected even during normal operation of the machine. Modelling physics and thermodynamics-based models can be challenging because of the proprietary nature of the models, such as turbine and compressor maps, provided by original equipment manufacturers (OEMs). Researchers in this field often have to make assumptions when creating these models.

Fortunately, black box models are very valuable in this scenario due to their ease of adaptability and flexibility to new conditions. Training and using up-to date ANN based model for condition monitoring based on new data sets of GT parameters could solve the proprietary confidentiality. Furthermore; literature lacks enough source regarding comparisons of the different black box approach methodology such as the Feed Forward Back propagation network, Layer RNN and the NARX network models in terms of their deviation from the real system. Therefore the research is geared towards the use of ANN's to appropriately model and simulate the GT13E2 Heavy duty turbine of Afam power station.

## 1.3 Objectives

The main focal point for this research is to develop dynamic models for the simulation of the start-up of the GT13E2 gas turbine (Cited at Afam Power Plant, Rivers State). Other more specific objective includes:

- To obtain data from Afam Power Plant (APP) for modelling and simulation of the gas turbine.
- To identify real life aspects associated with the development of ANN models using operational data: What are the independent and dependent parameters? How many runs of start-up data should be utilized? How large is the data size?
- To identify and adopt a good data wrangling strategy to ensure unit dimensional consistency and to remove corrupt data for example obvious erroneous sensor signals, not a number (NAN) and not available (NA), so as not to disrupt the accuracy of the ANN model.
- To develop dynamic ANN models with FF-BPP, LayRecNet and NARX network.
- To determine the performance of developed ANN models and identify the most suitable ANN model for the MIMO and MISO system
- To simulate the selected ANN model for the MIMO and MISO system using unseen data from the plant similar in manouver to the start-up manouver

## 1.4 Justification of the Study

The thesis will enable APP to utilize a new tool for condition monitoring and sensor validation which previously has never been used by the plant for possible online deployment; because the model to do so has never been developed. Furthermore, the model will change the culture of preventive maintenance applied by the company in running their plant as predictive maintenance will be applied which could help to reduce the running and maintenance cost of the plant. The end result of this modelling approach, will enable the creation and deployment of a machine learning tool to be run in parallel with the considered gas turbine for real-time condition monitoring and gas turbine sensor diagnostics.

## 1.5 Scope of the Study

The research will develop MIMO models of the transient behaviour of APP PGT using three different networks: Feed forward BPP, layer RNN, and NARX. Comparisons in the performance of the networks will be made. Furthermore the study will develop dynamic MISO models of the chosen ANN network with the best performance, and thereafter test/simulate the trained MISO and MIMO models of the best performing network. Thermodynamics and physics based model will not be considered in this research. Furthermore, this thesis will not investigate and classify the various kinds of faults present in the turbine, although there are various kinds of black box modelling and ANN available this research will be restricted to the three adopted networks.

# CHAPTER TWO

# LITERATURE REVIEW

## 2.1 Gas Turbine in Power Generation Industries

A gas turbine is a rotating machine that contains its own turbine chamber, compressor, and combustion chamber. A turbine plant, combustion chamber and compressor unit connected to one or more shafts is called a gas generator (Soares, 2014).

Heat is generated due to the ignition and combustion of the air and fuel mixture in the combustion section of the turbine. These installations and design makes them similar to internal combustion engine (ICE) (Gas Turbine Dictionary Definition, 2016).The gas turbine has gained key dominance in the past forty years in the power industry as a major energy generator and also has been the most popular energy supplier in the petrochemical, process industries and utilities around the world. Offshore platforms adopt the use of the gas turbine as a primary source of power due to its compactness, low weight and multi fuel flexibility which makes it a top preference in these remote locations. They supply the energy that propels propellers in ships and air crafts; their shafts drive compressors for pumping natural gas through pipelines.

Today gas turbines are designed to be flexible to accept light distillate fuels: naphtha, diesel, and gasoline, natural gas: propane and methane, biomass fuels and gases, bituminous and pulverized coal. However highly flammable and combustive fuels are often used specifically tailored to the design so as not to reduce its thermal efficiency (Meherwan, 2006).

For the turbine to attain design speed and run independently a distinct starter module is used to provide the first rotor motion. The gas turbine working principle is based on production of mechanical energy through the reaction of air-fuel mixture in the combustion chamber as a result of compressed air drawn in from the compressor and uninterrupted fuel supply regulated by the fuel valve. The hot gas is drawn through the turbine expander and the blades

of the turbine expander begin to spin as the energy from the hot gas is extracted by the blades. The spinning blades and rotor causes the shaft connected to the compressor to begin to rotate and rod attached to the generator begin to spin leading to electricity production.

## 2.1 Simple Cycle Gas Fired Turbines

### 2.1.1 Open cycle gas turbine

Peak-load utilities were the principal purpose for which open cycle gas turbines (**OCGT**) for electricity production were developed decades ago. Fundamentally OCGT plants comprises chiefly of the turbine expander, an air compressor arranged on one shaft connected to an electricity generator. Approximate 66.6% of the overall power output produced by the OCGT is consumed by the air compressor and the residual 33% is channelled to drive the electricity generator. Open cycle gas turbines have fairly low power electrical efficiency varying between 35 – 42% while aero derivative turbines are able to give efficiency ranging between 41-42% but they are only able to produce a maximum of 40-50 MW of electricity due to their compact size. Atmospheric air is first treated in the air filters and its pressure raised by passing it through the turbines compressor. Compressed air is exuded to the combustive reactors in the combustion chamber where combustion reaction with the flammable fuel occurs. As the temperature and pressure of the combustion products increase, the gas enters the turbine expansion unit due to the increase in pressure and temperature. The gas is expanded by the turbine's blades and that energy is used to drive or generate electricity. Exhaust gases are released into the atmosphere by emptying the turbine. It is an open cycle system as no exhaust gases are recycled back into the process. It is theoretically assumed that open circuit turbine plant does not follow the Brayton cycle principle but in practise modern gas turbines designed as internal combustion engines operate by this thermodynamic

principle. Moreover; its performance is frequently evaluated by considering it as analogous to closed circuit power plant, however care should be taken with such proposition (Paoli, 2009).

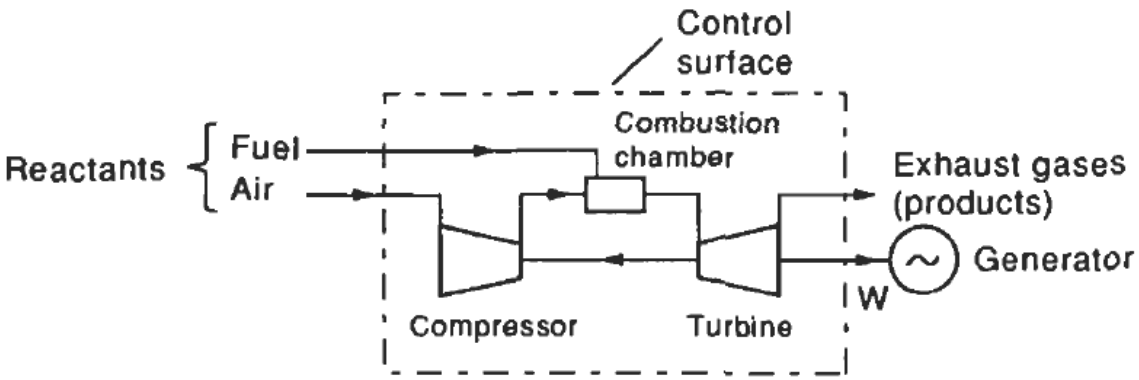



**Figure 2.1:** An open cycle turbine (Paoli, 2009).

### 2.1.2 Closed cycle turbines

The closed circuit turbine plant has been designed in theory but its real life implementation has not been fruitful. The fluid is usually an inert gas (not usually air) that is transmitted continuously around the process. Divergent working fluids like neon, carbon IV oxide, helium and argon can be used. The turbine is also able to accept solid fuels and dirty fuels like biomass and coal without damaging or leading to erosion of the blades. The pressure of the fluid is increased with the aid of a compressor after it is channelled to a heat exchanger/heater (QA) where its temperature and pressure is further raised without internal combustion. The fluid is passed through an expander where power is extracted from it. This energy created is utilized to power the compressor and the remaining power output (W) is channelled to drive the generator shaft (Paoli, 2009). The fluid is recirculated through the exhaust to the second heat sink (QA) or cooler where the fluid is regulated to meet the inlet fluid conditions of the compressor (Mishra, 2016).

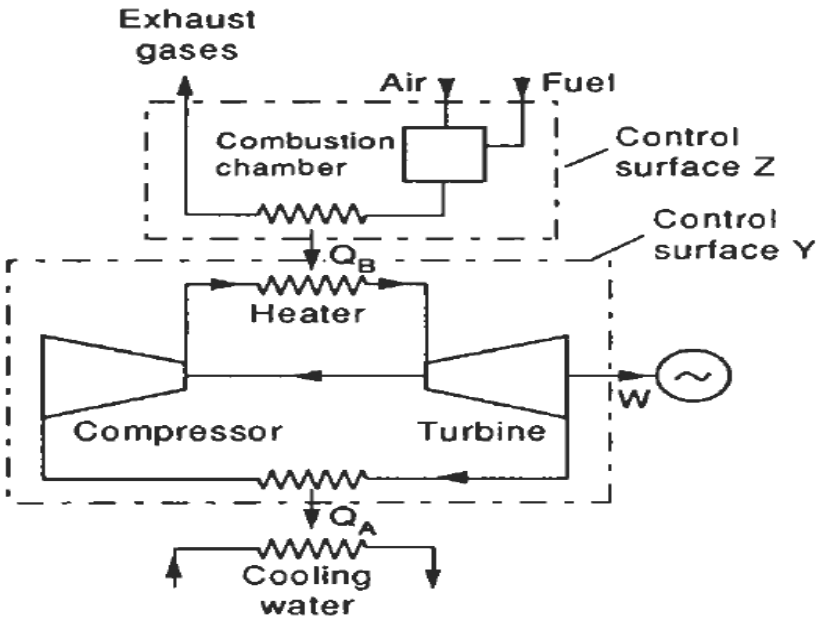



**Figure 2.2:** closed cycle turbine (Paoli, 2009).

## 2.2 Combined Cycle Power Plant

The combined cycle power plant (CCPP) is one of the most relevant focus for research for power plant manufactures vital area of research for power plant manufacturers. The aim of the CCPP is to produce higher work output and increase efficiency by transforming the heat ejected in the upper cycle to thermal energy in the bottoming cycle. A broad definition of a combined power plant (Figure 2.3). A "top" (top or top) thermodynamic cycle is a cycle in which energy is produced but some or all of the heat removal is used to supply heat to the "bottom" or bottom cycle.. The upper part of the turbine operates by the Brayton thermodynamic principle and the fluid moves through an open circuit configuration; while the lower part of the plant operate by the Rankine thermodynamic cycle. The fluid flow is closed circuited as steam is the preferred fluid for energy generation. The hot gases ejected through the exhaust are routed to the lower bottoming cycle to boilers, evaporators and super heaters. The boilers raise the temperature of the water to its boiling point. In the evaporator duct burners are used to supplement the heating process. The evaporator helps further

evaporate the saturated liquid to steam. Finally in the super heater, the steam becomes superheated and traces of moisture are all converted to (superheated) steam. Superheated steam is channelled through the turbine blades as mechanical energy extracted from this process is converted by a generator to electricity. Steam exuded from the turbine is transmitted to a condenser where it is made to undergo condensation. This fluid is recirculated back to the boilers to continue the process of power generation. The overall process ensures that the CCPP operation efficiency ranges between 50-60%. (Seebregts, 2010).

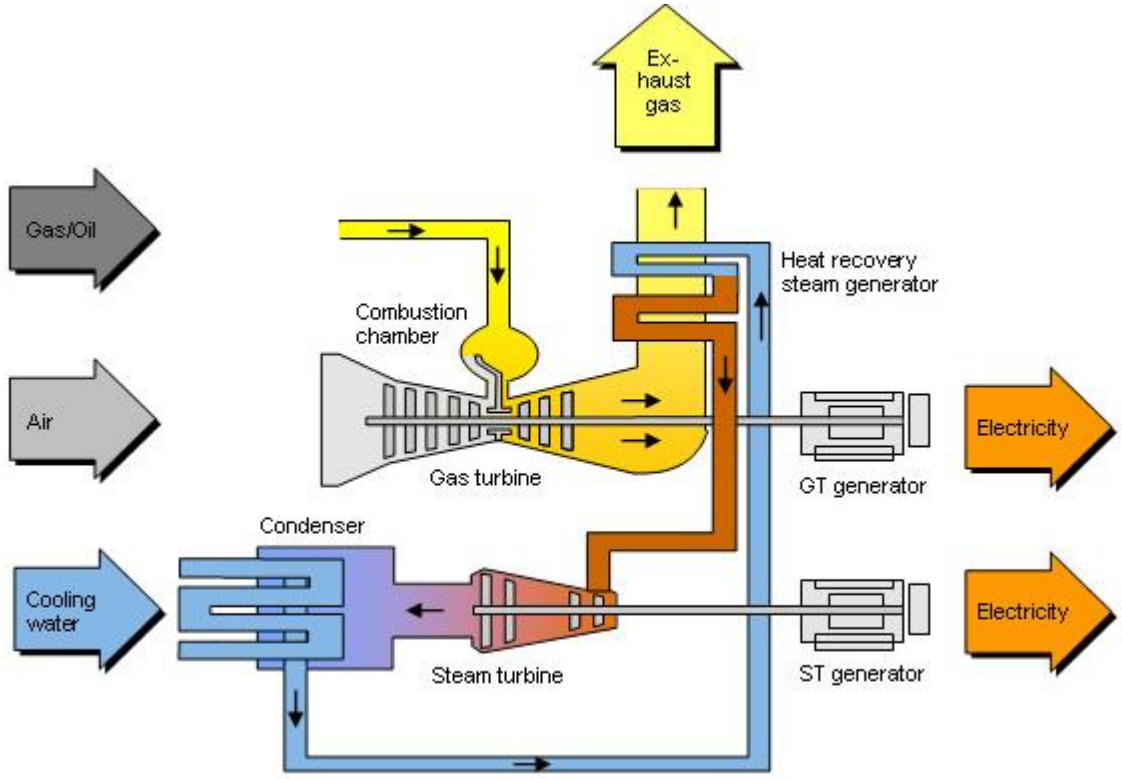



**Figure 2.3**: Gas-fired CCGT plant (Siemens, 2008).

## 2.4 Categories of Gas Turbines

### 2.4.1 Frame type heavy-duty gas turbines

The inspiration for the first set of designs of the heavy duty industrial gas turbine was drawn from the steam turbine concept. The first obsolete designs did not conceptualize size and

units the early heavy-duty gas turbine design was largely an extension of steam turbine design. Restrictions of weight and space were not important factors for these ground-based units, and so the design characteristics included heavy-wall casings split on horizontal centrelines, sleeve bearings, large-diameter combustors, and thick air foil sections for blades and stators and large front surfaces. The total weight reduction for this unit ranges from 5:1 for the previous unit to 35:1 for the unit currently in operation. The turbine inlet temperature increased and some of these units reached 1371 °C (2500 °F). This makes gas turbines one of the most efficient power engines on the market today, achieving 50% efficiency. Expected temperatures are approaching 3000 degree Fahrenheit (1649$^{o}$C), which will make gas turbines even more efficient. An industry-leading gas turbine program sponsored by the US Department of Energy targets these high temperatures. To achieve these high temperatures, modern designs use steam cooling to achieve the goal of keeping sheet metal temperatures below 1300°F (704°C) and avoiding high-temperature corrosion problems (Meherwan, 2006).

High-power industrial gas turbines use axial compressors and turbines. Industrial turbines consist of 15-17 phase axial compressors, each with several annular combustion chambers connected by crucible tubes. The transmission hose spreads the flame from one combustion chamber to all other chambers and ensures pressure equalization between each combustion chamber. Previous European industrial designers used single-stage side combustion chambers. The turbo expander has an axial compressor and a 2-4 phase axial flow turbine that drives the generator. The wide front of this unit reduces the suction speed, which reduces air noise. Weight gain is reduced with each compressor stage, creating a large and stable workroom.

The latest frame-type unit included is a 480 MW unit using steam cooling with a combined cycle that can reach combustion temperatures of 2600°F (1427°C). This increases the efficiency of the combined cycle by up to 60% or more.

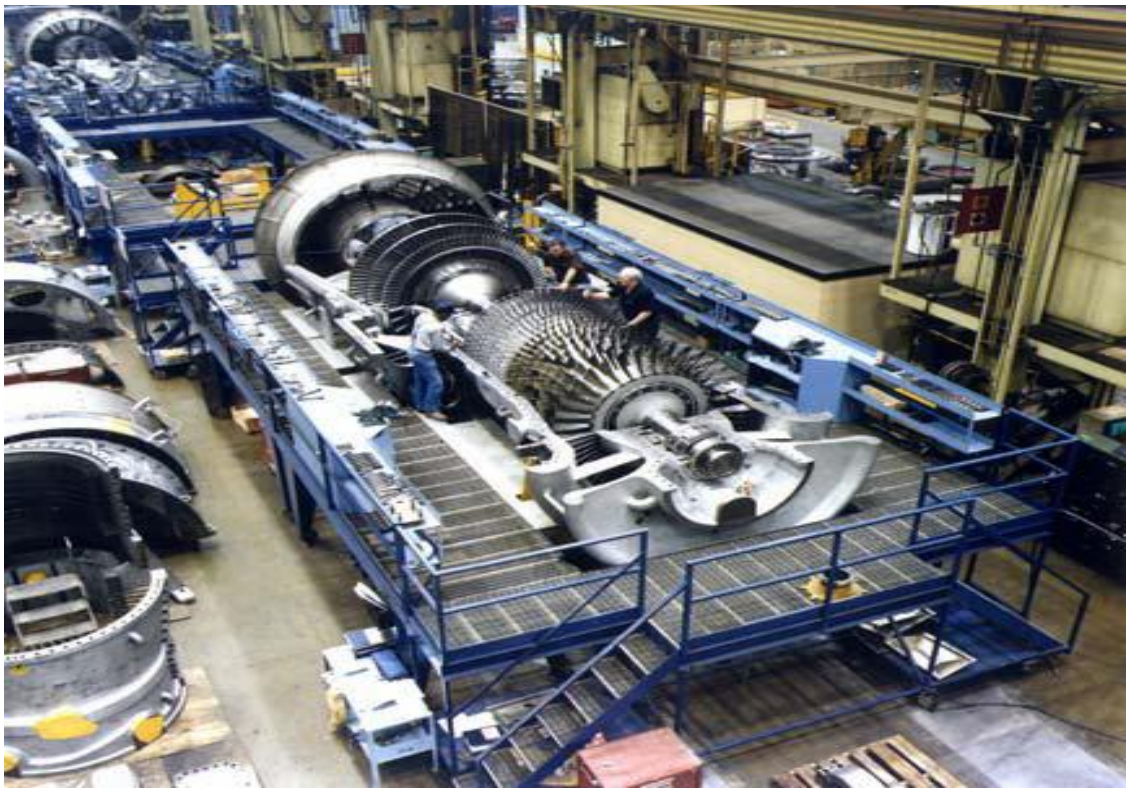


**Figure 2.4:** GE Frame 7F turbine during manufacture on a test bed in an OEM facility showing rotor in half the casing (Source: GE Power Systems).

### 2.4.2 Aero-derivative gas turbines

Aero derivative gas turbines are a popular choice for energy generation thanks to their reliability, efficiency and flexibility. Based on advanced aircraft engine technology and materials, it is significantly lighter, more responsive and takes up less space than heavy-duty RV offerings. With up to 45% efficiency compared to up to 35% for heavier GTs, these turbines are often seen as good choice in smaller-scale (up to 100 MW) energy generation. The turbines are also popular due to their flexibility –they allow a combination of natural gas

and liquid fuel operation. These turbines are lightweight and thermally efficient, but cost more than products designed and built for standalone applications. The largest turbines available for the production of aircraft engines range from 40 to 50 MW. Many standalone gas turbines operate under compression conditions in the 30:1 range requiring an external high-pressure gas compressor. With advanced system developments, larger aero derivative turbines (>40 MW) are approaching 45% simple-cycle efficiencies (LHV) (Energy Solutions Centre, n.d.).

As such the global aero derivative GT market was expected to grow at an annual growth rate of nearly 5% between 2016 and 2020, according to a 2016 study by Technavi. Asia, in particular deploy many of these machines in power trains for Liquified Natural Gas (LNG) plants. In the United States, aero derivatives are mainly being used in peak operations, or to compensate for fluctuations in the grid caused by renewable or extreme weather conditions (Robb, 2017).

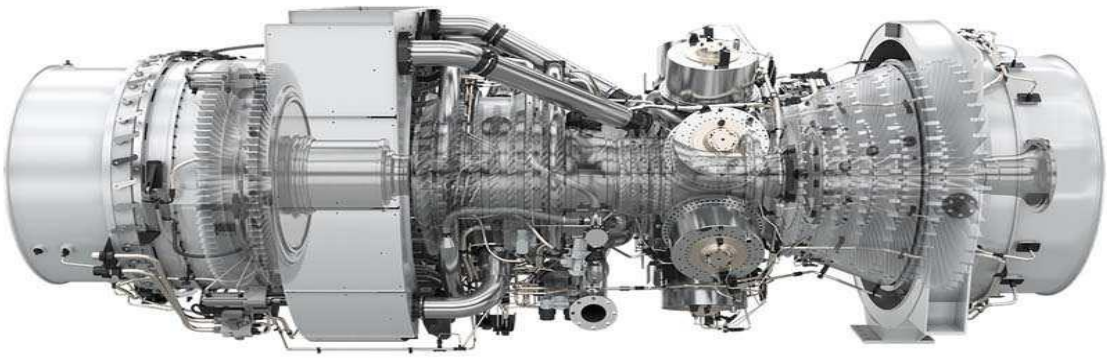


**Figure 2.5:** An aero-derivative gas turbine engine. (Courtesy: Siemens Power Generation).

### 2.4.3 Industrial type gas turbines

Industrial turbines are specifically designed for stationary power generation applications and lie in power range from 1 to 350 MW. It is relatively cheaper, durable, can last longer

between inspections, and is better suited to operating continuous base loads with longer inspection and maintenance periods than aircraft turbines. Its demerits, lies in the fact that it is less efficient and much heavier. They do not need an external attachment of air compressors due to its high compressor ratio (up to 16:1).Today large industrial turbines not less that 100MW have been able to achieve about 40% LHV and combined cycle efficiencies of up to 60% (LHV) in steam turbines. It should be noted that in power generation applications, simple cycle efficiency does not play a significant role in determining power generation efficiency. A turbine with low cycle efficiency can have a better power plant than a simple high cycle efficiency turbine. This is especially true for heated CHP power plants, which make up the majority of industrial gas turbine power plants.

In petroleum processing and manufacturing industry, gas turbines ranging from 500 kW to 40 MW are utilized for on-site power generation and mechanical propulsion systems. Miniature gas turbines also drive compressors in the gas network. In the oil industry, turbines operate gas compressors to maintain high, stable pressure. In the steel industry, turbines operate air compressors used in furnaces. In process industries such as chemical refining, oil and pulp and paper, as well as large-scale commercial and institutional applications, turbines are used in production mode to generate electricity. (Energy Solutions Center, nd).


**Figure 2.6:** A section of an Industrial gas turbine, within the Wang Noi compressr station in Thailand (SGT-400 industrial gas turbines Courtesy Siemens Power Generation).

### 2.4.4 Micro-turbines

Micro turbines are machine which produce energy by combusting fossil fuels to propel a generator. Today's micro turbine technology is the result of the development of small stationary and vehicle-based gas turbines, auxiliary equipment and turbochargers. Micro turbine sizes and developments range from 30 to 400 kW (kW). Micro turbines have high rotational speed and similar to industrial turbines are limited to power generation or power generation systems. They can work with a variety of fuels, including natural gas and liquid fuels such as gasoline, kerosene and diesel heated/distilled oil. In systems with embedded recycling operations, gases that would be combusted or discharged to the open atmosphere are utilized. These devices are designed to reduce the design and manufacturing costs of turbochargers and the reliability of auxiliary power systems used in commercial aircraft.

Micro turbines can be centrifugal with axial or radial inflow. Cost, efficiency and early emissions will be the three most important design criteria for these devices. For a micro

turbine to be efficient, it must be compact, have low manufacturing costs, be highly efficient, operate quietly, less noise pollution, and have minimal environmental gas emissions. Today's micro turbines use radial turbines and compressors. Reagents are used to increase the overall thermal efficiency of micro turbine designs and can be combined with absorption chillers or other thermal loads to achieve very high efficiencies (Meherwan, 2006).

### 2.5 Ideal Brayton Cycle

The ideal Brayton cycle can be used to understand the thermodynamics of the simple cycle turbine. For the working gas, the ideal Brayton cycle includes the following steps: (1) a reversible adiabatic (no heat exchange) compression process, (2) a constant-pressure heat-addition process to the maximum temperature, (3) an isentropic expansion process through a turbine back to the original pressure, and (4) a constant-pressure cooling process back to the original pressure (exhaust).

The Brayton cycle has the shape seen in Fig. 2.7 on a temperature–entropy diagram. The enthalpy is proportional to the temperature for an ideal working gas of constant specific heat at constant pressure (Cp), hence Fig. 2.7 can also be considered an enthalpy-entropy diagram with suitable ordinate scaling. Isentropic compression is element (1–2). Constant-pressure heating is element 2–3. Isentropic expansion is element (3–4), while constant-pressure cooling is element (4–1). The power required for compression is h2 - h1, where h stands for the enthalpy per unit mass, while the power produced for expansion is h3- h4. The slopes of the constant-pressure elements are proportional to the temperature in Fig. 2.7, which means that element (2–3) has a bigger slope than element (1–4) at a given entropy. As a result, the lines diverge, and the expansion power (h3 - h4) exceeds the needed compressor power (h2-

h1), resulting in a net positive power output per unit mass flow (Encyclopedia of Physical Science and Technology, 2001).

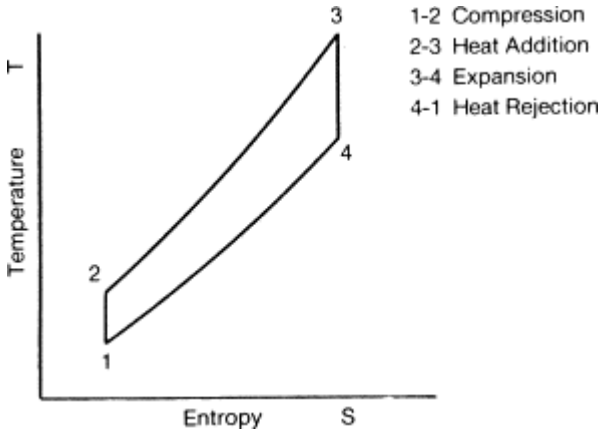



**Figure 2.7.** Ideal Brayton-cycle thermodynamic diagram.

The power output and efficiency for the ideal Brayton cycle can be evaluated from the thermodynamics of the individual elements. For the isentropic compression process,

$$h_2/h_1 = p_2/p_1{}^{\gamma-1/\gamma} \tag{2.1}$$

Where $p$ is the pressure and $\gamma$ is the ratio of the specific heat at constant pressure to the specific heat at constant volume (for air, $\gamma$ equals about 1.4). Thus, the required compressor power (per unit mass flow) is

$$h_2 - h_1 = h_1 p_2/p_1{}^{\gamma-1/\gamma} - 1 \tag{2.2}$$

While for the expansion process the power out is

$$h_3 - h_4 = h_4 p_3/p_4{}^{\gamma-1/\gamma} - 1 \tag{2.3}$$

Because of the nature of the constant-pressure processes, $p_4 = p_1$ and $p_3 = p_2$; therefore the net power is

$$h_3 - h_4 - h_2 - h_1 = h_4 - h_1 p_r^{\gamma - 1/\gamma} - 1 \quad (2.4)$$

$$= h_3 - h_2 1 - p_r^{-\gamma - 1/\gamma} \quad (2.5)$$

Where $p_r$ is the pressure ratio, defined as

$$p_r = p_2/p_1 = p_3/p_4 \quad (2.6)$$

But the energy input rate to the cycle is $(h_3 - h_2)$. Therefore, the efficiency $\eta_B$ of the Brayton cycle is

$$\eta_B = 1 - p_r^{-\gamma - 1/\gamma} \quad (2.7)$$

Since the temperature ratios for both the isentropic compression and expansion elements are given by

$$T_2/T_1 = T_3/T_4 = p_r^{\gamma - 1/\gamma} \quad (2.8)$$

then

$$\eta_B = 1 - T_1/T_2 = 1 - T_4/T_3 \quad (2.9)$$

This differs from, and is lower than, the efficiency $\eta_C$ of the Carnot cycle for the same temperature limits, which would be

$$\eta_C = 1 - T_1/T_3 \quad (2.10)$$

## 2.6 Gas turbine Model Construction Approaches

### 2.6.1 White-box models

The white box model is used when the knowledge of the physics of the system is deficient. In this scenario, the model is produced using mathematical equations for system dynamics. These models usually refer to dynamic equations for coupled and nonlinear systems (Masdi et al, 2015). To simplify these equations and produce a satisfactory model, some assumptions must be made based on the ideal conditions and other methods used to linearize the system.

#### 2.6.1.1 IEEE model

The IEEE models are of two kinds: the model depicting the thermodynamic characteristics of the turbine and the model representing the control loops i.e. the temperature control loop, fuel flow and the air flow control loop. The model tends to ensure low greenhouse gas emission by ensuring a non-high non varying firing temperature keeping the turbine speed constant and regulating the inlet guide vanes. The gas turbine model is applied to a single cycle gas turbine with constant speed and variable blades to maintain a constant combustion temperature for low greenhouse gas emissions. Assessment of Rowen's 1983 paper in contrast to the IEEE model shows that the later has control loops that ensure a constant turbine inlet temperature (TIT). A valuable by effect of the high TIT is the low greenhouse gas emissions such as the release of nitrous oxide ($NO_x$) in minute quantity. The model design assumes a non-dynamic compressor ratio, which cannot be achieved unless the rotational speed is relatively stable. The control schematics is shown in Fig. 2.8, which is analogous to the one developed by Rowen. Moreover, as the guide vanes modulation is modelled (IEEE Transactions on Power System, 1994) there is an inclusion of the air flow control loop.

Figure 2.8 is a Simulink work flow that depicts the controls needed to obtain the mechanical power output of the turbine. It represents the necessary calculations for deriving the mechanical power output of the gas turbine. The signal lines in the diagram are basically from the isentropic equations. The connections in the block diagram are essentially based on the isentropic efficiencies equations (Cohen, Rogers, & Saravanamuttoo, 1996)

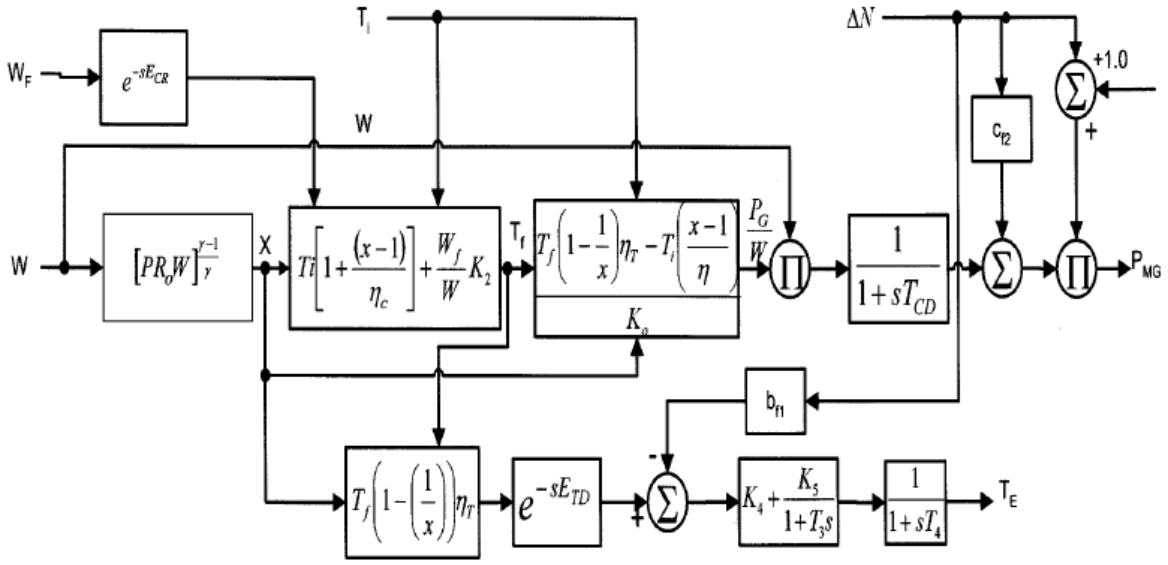



**Figure 2.8**: IEEE model thermodynamic equations.

### 2.6.2 Grey-box models

In addition to white-box and black-box methods, the phrase grey-box may be also used when an empirical model is improved by utilizing a certain available level of insight about the system (Norgaard, Ravn, Poulsen and Hansen, 2000). In this case, experiments can be combined with mathematical model building to improve model accuracy. For many industrial processes there exists some but incomplete knowledge about the structure of the process. The amount of knowledge varies from one process to another, between white box models and black box models there is a grey zone. There are processes about which there is some, but incomplete knowledge about the structure of the process. Models are based on available

physical information. This information is not only incomplete, but also ambiguous. The grey box modelling approach can be efficient even when the process is not linear. In these cases it may be difficult to achieve a satisfactory black box model when the operating point is changing and it may be necessary to use several black box models, which are related to the operating points. It should be pointed out that a grey box model may be more expensive to develop than a black box model, since it may take longer to construct a grey box model than a black box model. Moreover the purpose the model serves would determine the type of model that needs to be developed, and if a grey box model would be profitable (Sohlberg, 1998)

### 2.6.3 Black-box models

A black-box model is used when no or little information is available about the physics of the system. In this case, the aim is to disclose the relations between variables of the system using the obtained operational input and output data from performance of the system. Artificial neural network (ANN) is one of the most significant methods in black-box modelling. ANN is a fast-growing method which has been used in different industries during recent years. The primary goal in developing ANN models which is a subset of artificial intelligence is to portray the model of human brain in order to proffer solutions to complex scientific and industrial problems in diverse areas (Asgari, 2014).

#### 2.6.3.1 Artificial neural network

The science of training machines to find the right solution for complex problems in a more humanlike fashion is known as artificial intelligence (AI). This generally applies to computer machine via an algorithm, which has approximated human brain characteristics. Artificial intelligence has four main branches that include case based reasoning, genetic algorithms, expert systems and neural networks (Hagan, Demuth & Beale, 1996) which can be chosen based on the specific application.

An artificial neural network (ANN) is an information-processing paradigm that is inspired by the circuit of biological neurons with the aid of learning to recognize patterns from a certain data. At the heart of this paradigm is a new method of analysing data with reasonably accurate resolution when the underlying data relationships are unknown. (Kumar, 2007). Once the NN is trained on collection of data, it can predict the outcome by detecting similar patterns of the input. Thus using ANN for analysing the data stream can solve the problem of detecting and predicting possible faults even if the data are imprecise and noisy (Vatani, 2013). Instead of being pre-built to specifications, neural and adaptive systems use external data to automatically set parameters. ANNs can be used to solve a variety of problems, including classification, regression, and general evaluation problems. The two main paradigms are:

• **Supervised learning**: in which both inputs and desired outputs are known. This means that the network can measure its predictive performance for given inputs.

• **Unsupervised learning**: in which the targets are unknown. The ANN has to find the underlying relationships within the data set by itself, and build clusters of data (Darel, 2001).

Supervised learning is used for tasks of classification and regression, whereas unsupervised learning is more suitable for data clustering, compression and filtering tasks. There is a fundamental difference between clustering and classification. Clustering represents the

process of grouping input samples that are spatial neighbours, whereas classification involves the labelling of input samples via some external criterion. Clustering is an unsupervised process of grouping, while classification is supervised. These differences are illustrated in Figure 2.10 (Banzhaf, 1998).

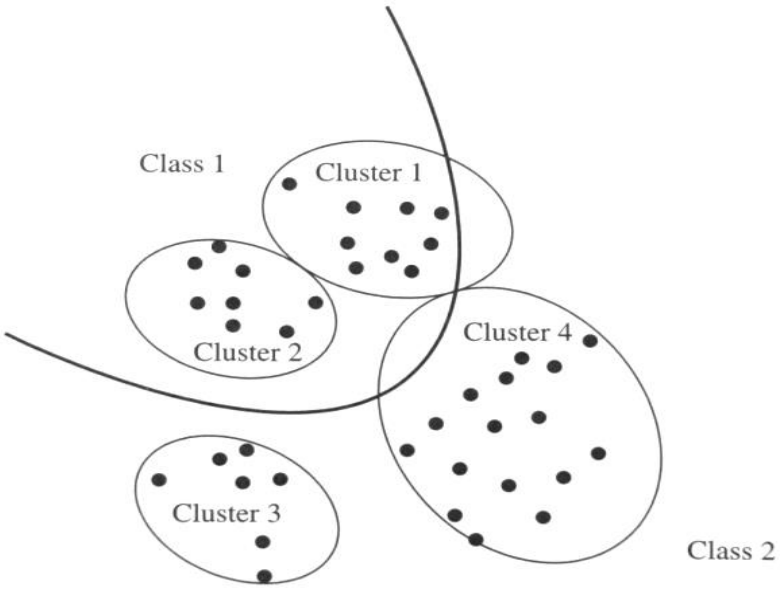



**Figure 2.9:** The differences between clustering and classification (Banzhaf, 1998).

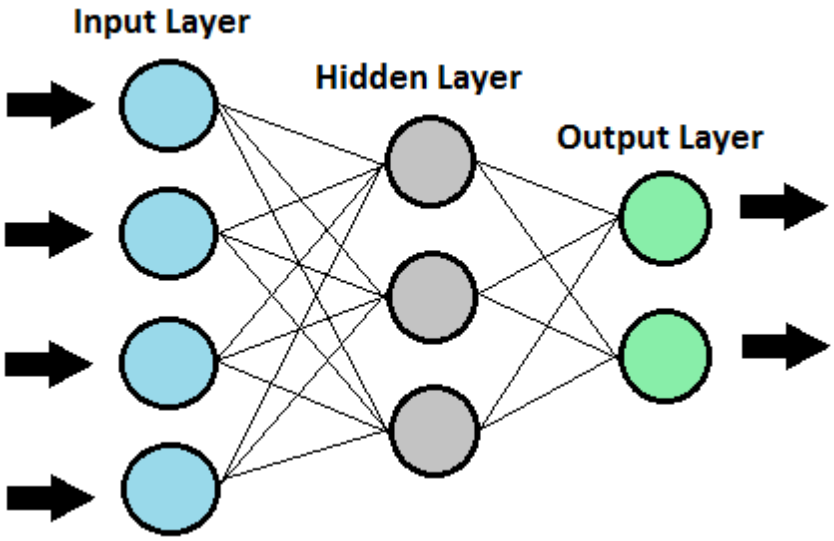



**Figure 2.10:** A simple structure of an ANN with input, hidden and output layers.

The main idea behind creating artificial neural networks is that they mimic the human brain and are utilized as a technique to solve complex problems in various scientific fields, including technology, psychology, linguistics, philosophy, economics, and neurobiology. ANN consist of system of simple, interconnected groups of process elements (neurons) with linear or nonlinear transfer functions. These elements process information through dynamic state responses to external inputs (Caudil, 1989). Neurons exist in various layers, including input layers, hidden layers, and output layers. The number of neurons and layers in an ANN model depends on level of complexity of the system dynamics. ANNs observes interlink between system inputs and outputs through an iterative process called learning. Each input to a neuron has its own weight. Weight is an adjustable value that is determined during training. Choosing suitable input and output parameters for an ANN is very important to generate accurate and reliable models. Availability of data for selected parameters, system information to identify relationships between various parameters, and model creation goals are the most important factors in selecting appropriate inputs and outputs. The accuracy of the selected output parameters can be checked using sensitivity analysis.

#### 2.6.3.2 The model of artificial neuron

Artificial neurons are the most important and fundamental building blocks of any artificial neural network structure. Figure. 2.12 shows a simple single-input neuron with components including inputs, outputs, summation and function blocks (Hagan, Demuth & Beale, 2002). *p*, *w*, *b*, *f*, and *a*, are scalar input, scalar weight, bias, transfer (activation) function, and scalar output, respectively. The output of the neuron is calculated using Equation (2.1). The parameters w and b can be established by training rules such that the relationship between input and output is consistent with the expected goal (Hagan, Demuth & JESÚS, 2002). Bias is a weight independent of other nodes, and the input is always set to 1. The purpose of the

bias is to compensate for the origin of the transfer function for faster convergence. So the offset allows the node to have an output even if the input is zero.

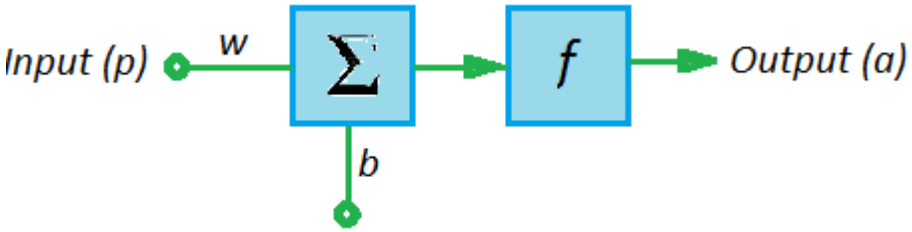



**Figure 2.11:** Single-input neuron structure.

$$a = (w*p+b) \tag{2.11}$$

A neuron usually has one or more inputs. Figure 2.10 show multiple-input neuron structures with one and multiple neurons in the hidden layer respectively (Hagan & Demuth, 1999). R and S represent the number of elements in the input vector and the number of neurons in the layer. In this case the input *P*, weight *w* and output *a* will be vectors and equation (2.1) will have matrix sign as shown in figure 2.10. A neural network can have multiple layers operating in parallel. Each level has its own inputs, outputs and components.

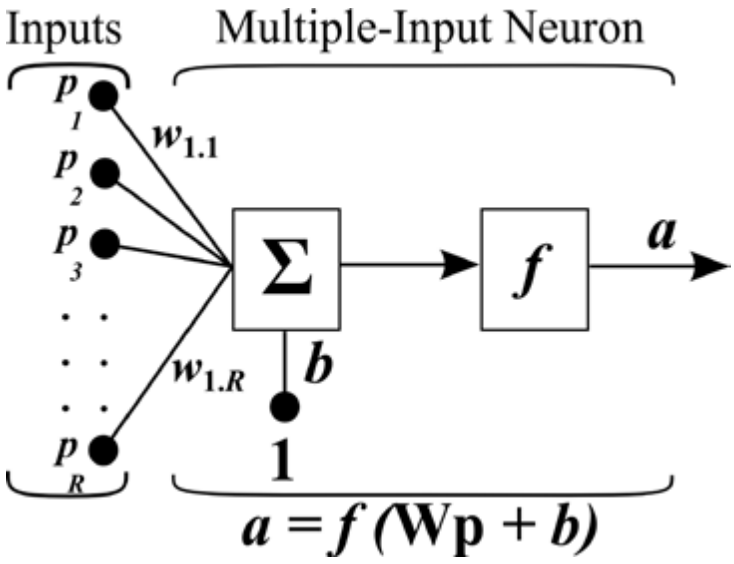



**Figure 2.12:** Multiple-input neuron structure (Hagan, Demuth & Jesús, 2002).

#### 2.6.3.3 Feed-forward back propagation network

A feed-forward neural network (FFNN) consists of three or more layers of neurons: an input layer, one or more intermediate hidden layers, and an output layer. In most cases, neurons are coupled in a feed-forward fashion, with input units fully connected to hidden layer neurons and hidden neurons fully connected to output layer neurons. Back propagation is a traditional FF NN training method in which neurons adjust their weight to gain new knowledge. It stands for "backward propagation of errors" in abbreviated form. It's a technique for fine-tuning the weights of a neural network using the error rate from the previous epoch (i.e., iteration). By fine-tuning the weights, you may lower error rates and improve the model's generalization, making it more accurate (www.guru99.com, n.d).

The feed-forward back propagation works as follows:

• Input X arrives through the preconfigured path.

• Real weights W are used to model the input. The weights are normally chosen at random

• Calculate output error
  Error = actual output - desired output.

•Revert from the output layer to the hidden layer to change the weights in order to reduce the error

• Repeat the process over and over until you get the results you want.

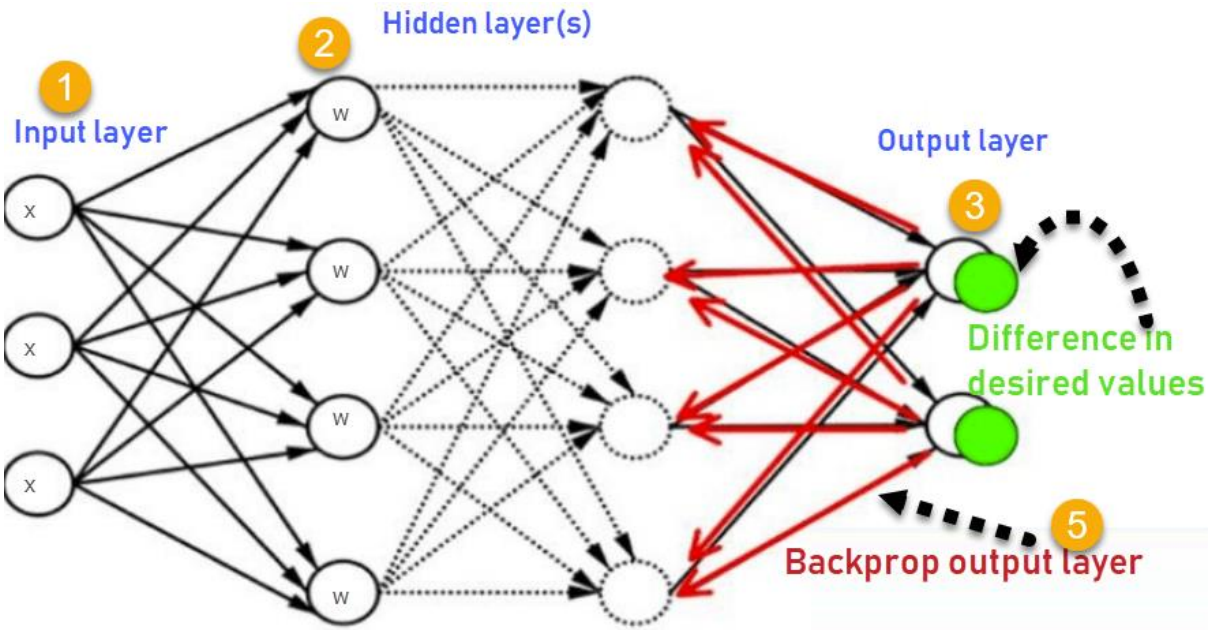



**Figure 2.13**: Feed-forward back propagation network (www.guru99.com).

During the training phase of FFNN with back propagation, each input pattern from the training set is applied to the input layer and then propagates forward. To calculate an error signal, the pattern of activation arriving at the output layer is compared to the correct (related) output pattern. The error signal for each of these desired output patterns is then returned from the output layer to the input neurons, allowing the weights in each layer of the network to be adjusted. After the training phase, in which the NN learns how to classify a set of inputs correctly, it can be tested on a second (test) set of samples to determine how well it classifies new patterns. As a result, how well the network generalizes is a crucial factor when using back propagation learning (www.cs.csi.cuny.edu, n.d).

#### 2.6.3.4 Recurrent neural networks (RNN)

The recurrent neural network is a special type of neural network (RNN). There can be one or more hidden layers of neurons in the network. RNNs and feed forward neural networks are distinguished by the presence of one or more feedback loops. Any two neurons or layers can establish a feedback loop in a variety of ways. Because they include a large number of feed forward and feedback connections, recurrent neural networks have complex dynamics (Chow and Cho, 2007). These connections give them an edge over feed forward ANNs when it comes to dealing with time-series and dynamical difficulties. A lower network size recurrent network may be equivalent to a large or intricate feed forward NN architecture. Recurrent networks have been widely used in the fields of intelligent control, system identification, and dynamical systems applications. Theoretical studies of network stability, convergence, and functional approximation skills are vital in these applications. Kim, Lewis, and Abdallah (1997) present an adaptive observer based on a generalized recurrent neural network with on-line learning. It is demonstrated that the whole adaptive observer scheme is consistently ultimately bounded. To train the RNN, a generalized real-time iterative learning algorithm is created and used. The study demonstrates that an RNN utilizing the real-time iterative learning approach can accurately imitate any trajectory tracking. In nonlinear dynamical systems, the use of recurrent neural networks as predictors and identifiers has gotten a lot of attention (Mandic & Chambers, 2001). Adders, delays, multiplies, and zero-memory nonlinearities are the fundamental building pieces of a discrete time predictor. Zero-memory nonlinearities such as threshold, piecewise-linear, and logistic are necessary to employ neural networks as nonlinear predictors. The neurons in the NN design are made up of these basic components. The neuron output $y_k$ is assumed to be the delayed version of the inputs. The nature of the network's inputs must capture some information about the discrete time signal in order to solve the prediction challenge.

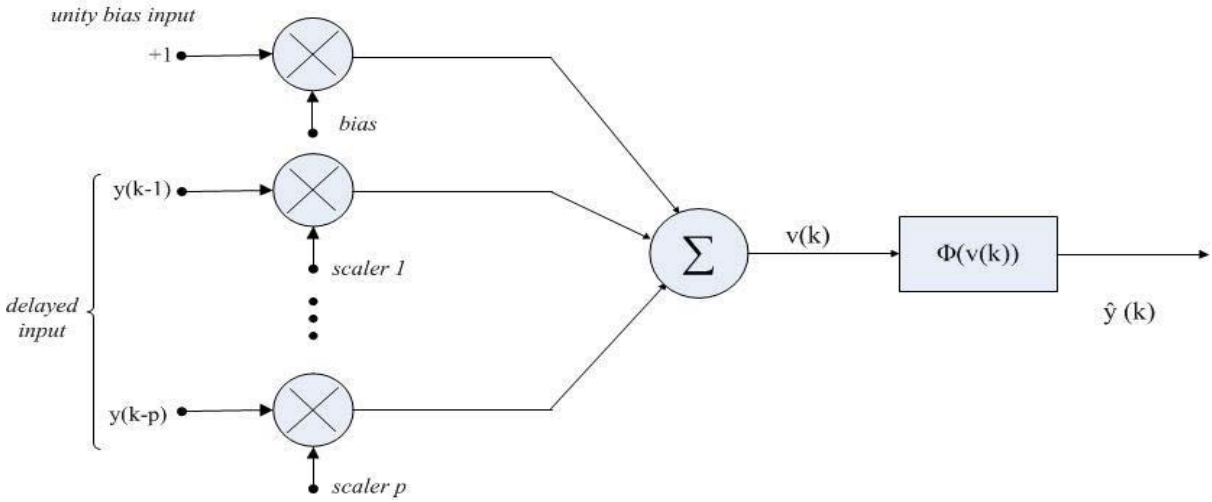



**Figure 2.14:** Structure of a neuron for prediction (Mandic and Chambers, 2001).

The simplest case is for the inputs to be the time-delayed form of the output signal, i.e. $y(k-i)$, $i = 1, 2, ..., p$, which is referred as a tapped delay line. This type of network provides a short-term memory of the signal. The overall predictor can be represented as:

$$y(k) = \varphi(y(k-1), y(k-2), ..., y(k-p)) \quad (2.12)$$

Where φ is the non-linear mapping of the neural network. A typical recurrent neural network is shown in Figure 2.15. If we consider linking a delayed variant of the network output $y_{(k)}$ to the input, along with the delays, we can introduce memory into the network, making it appropriate for prediction. Information on the network's stability can be found in (Mandic & Chambers, 2001).The feedback within the network might be local or global, as shown in the diagram. The global feedback is obtained by linking the network output to the network input, whereas the local feedback is achieved by adding feedback to the hidden layer. When used as a predictor, the RNN output can be expressed as global feedback.

$$y(k) = \varphi(y(k-1), y(k-2), ..., y(k-p), \hat{e}(k-1), ..., \hat{e}(k-q)) \quad (2.13)$$

Where $e(k-j) = y(k-j) - \hat{y}(k-j)$. In other words, by adding the feedback and tapped-delay line to a static network architecture, we are adding memory to the network and then it becomes capable of prediction.

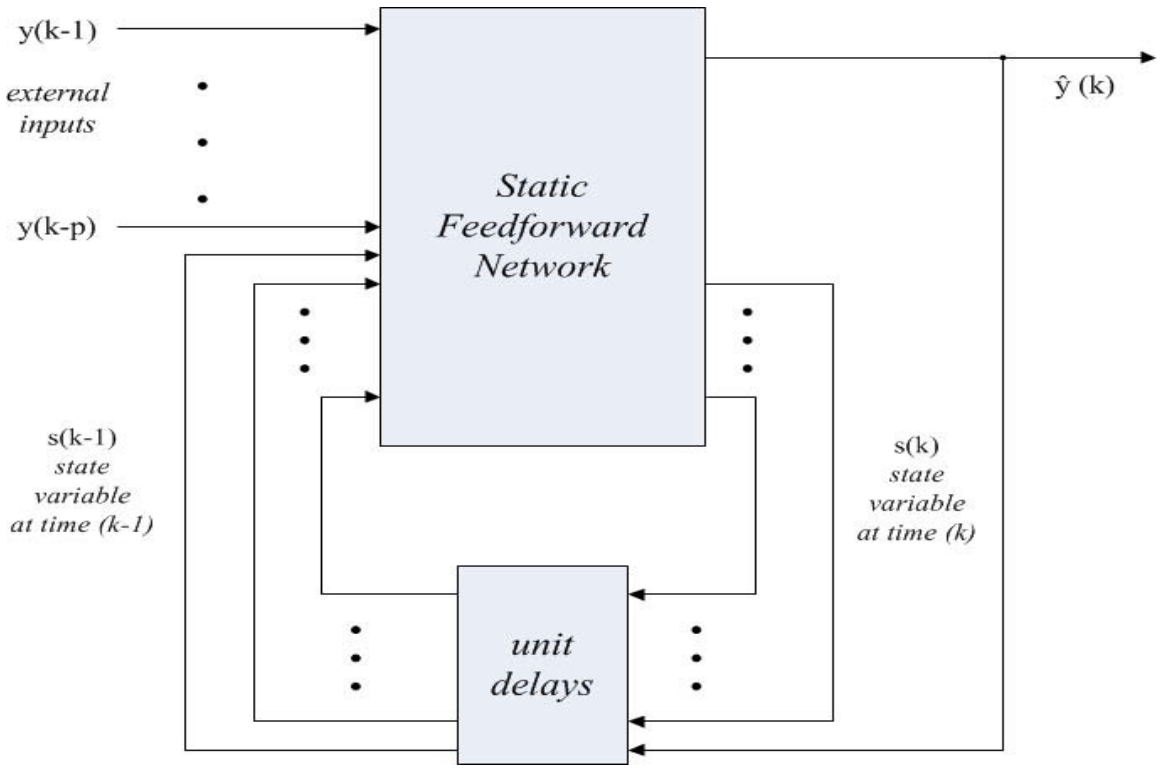



**Figure 2.15**: Canonical form of a recurrent neural network for prediction (Mandic and Chambers, 2001).

To introduce nonlinearity to the network, we use nonlinear activation functions. For gradient-descent learning algorithm, the function $\sigma$ (.) should be differentiable and belong to the class of sigmoid function. Surveys of neural transfer functions can be found in (Duch and Jankowski, 1999). According to Cybenco (Cybenko, 1989) a neural network with a single hidden layer of neurons with sigmoidal functions and enough neurons can approximate an arbitrary continuous function

#### 2.6.3.5 Nonlinear autoregressive neural network (NARNN)

This model is a hybrid of recurrent and dynamic networks as it has both a feedback path and dynamical neurons or the tapped-delay line. NARNN is a dynamic recurrent neural network that can model time series with long-term dependencies efficiently. Different layers of the network may be wrapped by feedback links. Figure 2.14 illustrates the overall architecture. The network is known as a nonlinear autoregressive neural network with exogenous input when the external input is taken into account (NARX). When an external input is used, the following network is generated:

$$y(k+1) = f[y(k), ..., y(k-dy+1); u(k), u(k-1), ..., u(k-du+1)]$$

$$= f[\mathbf{y}(k); \mathbf{u}(k)] \quad (2.14)$$

At time *k*, the next value of the dependent output signal y(k) is regressed against previous values of the output signal as well as previous values of an independent (exogenous) input signal u(k). Memory delays are represented by the parameters du and dy. The linear ARX model, which is extensively used in time-series modeling, is the foundation of the NARX model. This sort of neural network has two main architectures: parallel and series-parallel architecture.

The output of the NARNN network should be fed back to the input of the feed forward neural network as part of the conventional NARNN architecture if the output is an estimate of the output of some nonlinear dynamical system that we are trying to describe. The output regressor y(n) for the parallel configuration is as follows:

$$y_p(k) = [\hat{x}(k), ..., \hat{x}(k-dy+1)] \quad (2.15)$$

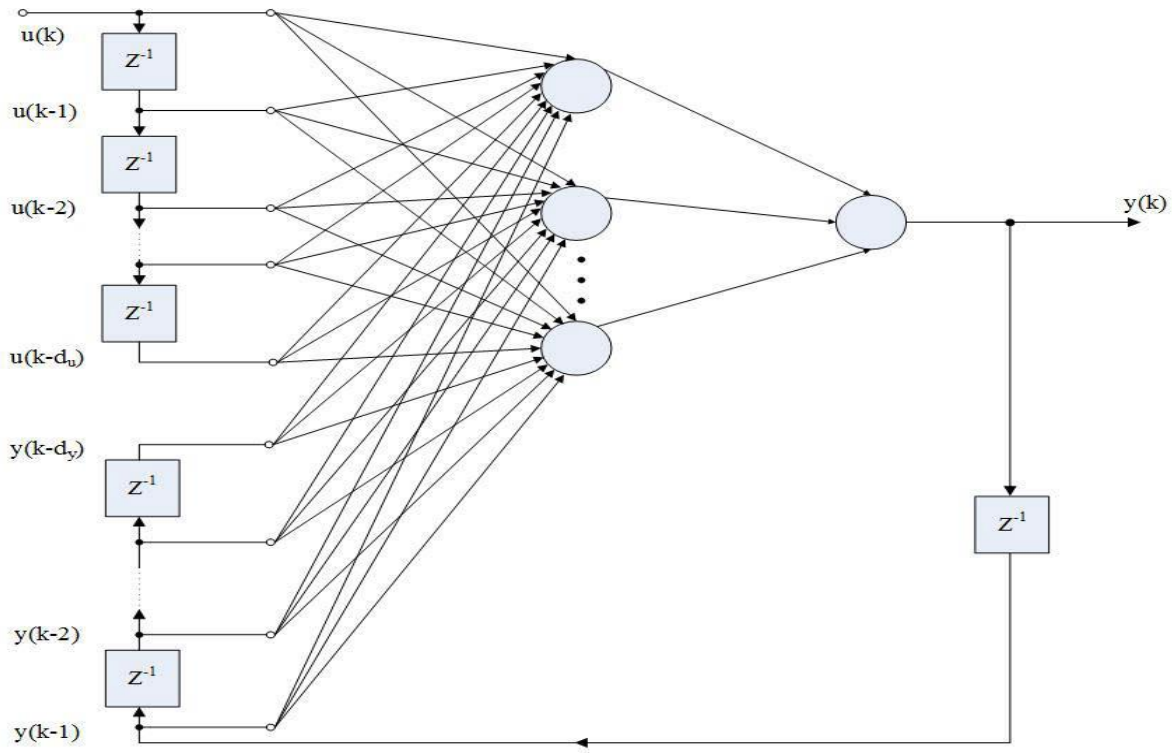



**Figure 2.16:** Nonlinear autoregressive neural network architecture (NARNN) ( Menezes & Barreto, 2006).

The P-mode comprises the estimated time series' previous past values. The second arrangement is known as series-parallel architecture, in which the actual output is used rather than the predicted output being fed back. In the SP-mode, the output regressor y(n) is obtained similarly to the P-mode:

$$y_p(k) = [x(k), \ldots, x(k - dy + 1)] \quad (2.16)$$

where the SP-mode holds the real time series' previous past values. There are two advantages to this architecture. To begin with, the feed forward network's input is more accurate since it takes historical values from the real time series rather than estimated values. Secondly, the resulting network has a completely feed forward design, and it can be trained via static back propagation. Compared to a fully recurrent network, the NARNN model converges faster and requires less training cycles. The NARNN model describes the system as a nonlinear function of delayed input, output, and prediction error, and so may be used to mimic any nonlinear

dynamical system (Siegelmann, Horne, and Giles, 2008). As a result, they've seen a lot of use in areas including prediction, time-series modeling, chaotic time-series prediction, nonlinear filtering, control and modeling, and so on. When compared to other methods, this method reduces network training time, improves accuracy, and yields good control performance under a variety of operating conditions (Gang & Yu, 2005).

$$y(k) = \Psi\{u(k), u(k-1)..., u(k-du), y(k-1), ..., y(k-dy)\} \quad (2.17)$$

Where Ψ denotes the mapping performed by the network's mult-layer preceptron.

The following states have been updated:

$$x_i(k+1) = \begin{cases} u(k) & i = d_u \\ y(k) & i = d_u + d_y \\ x_{i+1}(k) & i < i < d_u \text{and } d_u < i < d_u + d_y \end{cases}$$

so that at time *k* the taps correspond to the values

$$x(k) = [u(k-1), ..., u(k-du), y(k-1), ..., y(k-dy)] \quad (2.18)$$

## 2.7 Review of Related Works

A single shaft low power gas turbine was modelled and simulated by Asgari (2014a) . Based on thermodynamic and energy balance equations, the author developed a Simulink model of a low-power gas turbine in his research. A re-simulated nonlinear dynamic model of a low-power single-shaft gas turbine was used to obtain the data for this investigation. The main goal of their research was to increase the engine's dynamic response by implementing a custom nonlinear controller. Based on engineering principles, gas turbine dynamics, constitutive algebraic equations, and operational data, the model was created and simulated in Simulink MATLAB. The method offered a detailed information about the performance of over 18720 ANN models for single-shaft gas turbine system identification. A comprehensive computer code for a two-layer MLP network consisting of various back propagation training functions, transfer functions, and different number of neurons was generated and run in MATLAB to obtain an accurate network structure and ensure good generalization characteristics of the LPGT turbine model. The results show a very good fit for each of the training, validation, and test data sets. The author demonstrated that a neural network-based model could accurately predict the system's response to changes in input parameters. It reliably predicts the LPGT's output parameters based on changes in the system's inputs. However, he failed to conduct a comparative analysis to identify which sort of neural network is the most effective for modelling the LPGT. The researcher may have created a model of the LPGT operating at full speed or in steady state.

Baheta et al (2016) developed a twin shaft gas turbine engine model for performance simulation using GASTurb simulation software and validated the model with operational data collected from a working gas turbine. The researchers made use of genetic optimization algorithm in order to obtain unknown parameters in the design of compressor and turbine

maps in addition to the use of the GASTURB software as the marginal errors between the real and simulated values were way too abysmal**.**

Chaibakhsh and Amirkhani (2018) in their study built a simulation model for analyzing the transient behavior of a V94.2 industrial gas turbine. The mathematical models for heat transfer were created using mass and energy balances, thermodynamic state conversion, and semi-empirical correlations. The variables of the models were altered by applying a genetic algorithm (GA)-based optimization approach using data from online system performance. Although, the acquired results demonstrated the proposed model's remarkable performance for long-term simulation; but even so, they limited the study to combined cycle turbines, while simple cycle turbines are the most common turbines used in plants throughout the world.

The transient behavior of an industrial power gas turbine was studied and simulated by Asgari (2014b). They were created to mimic an IPGT's very low-power operating region. Using Simulink and NARX models, the researcher was able to recreate gas turbine transient behavior during startup and near full speed operation. At the time, according to Asgari (2014b), no model for representing IPGT transient behavior had been published.

The parameters which were measured directly from the IPGT include: rotational speed ($N$), alternator power ($Wload$), ambient pressure ($P00$ ), ambient temperature ($T00$ ), compressor inlet stagnation pressure ($P01$ ), compressor inlet temperature ($T01$ ), compressor outlet stagnation pressure ($P02$), compressor outlet temperature ($T02$), turbine outlet temperature ($T04$), and fuel flow rate ($\dot{m}f$). During several start-up maneuvers, the data sets utilized for model setup and verification were collected experimentally. Asgari (2014b) used four separate maneuvers, labeled M1, M2, M3, and M4. Model tuning was done with the M1 and M2 time series data sets, while model verification was done with M3 and M4. In the

Simulink-MATLAB environment, the gas turbine was modelled using thermodynamic and energy balance equations. The NARX model for the IPGT was built using the same time-series data sources. For all manoeuvres, the RMSE of rotational speed, pressure ratio, compressor outlet temperature, and turbine outlet temperature was equal to or less than 0.8 percent, 4 percent, 1.6 percent, and 3.1 percent, respectively. Simulink and NARX models had maximum errors of 4.3 percent and 2.8 percent, respectively. The findings of this study showed that the NARX approach modelled gas turbine behaviour more accurately than the Simulink approach. It was demonstrated that ANN may be used to identify system dynamics in a reliable and powerful manner. The author could have compared the IPGT's deterioration to the baseline value by using fouling and erosion as a case study. In showing the dynamic model of the IPGT, the researcher could have also compared other types of neural networks that outperform the NARX network with a lower RMSE. Furthermore, the researcher may have used SIMULINK and NARX models to construct a steady state model of the IPGT.

Asgari (2014c) simulated the model of an industrial power gas turbines with the NARX model. Modelling and simulations were performed on experimental time series data sets. The NARX IPGT model was constructed using three measured time series data sets. The NARX model was tested against three other experimental datasets that could be used to validate the model. Towards this end, he compared the corresponding values in the measured dataset with the four important model outputs: compressor temperature (T02) and turbine outlet temperature (T03), compressor compression ratio (PRC), and rotor speed (N) versus compressor inlet temperature, atmospheric pressure (P01) and mass fuel flow rate (MF). The time step of data collection is 1000 milliseconds. Because the compressor inlet temperature and pressure were taken at different times of the year (August, October and December), the amount of training data is different. This choice was specifically made to improve the generalization properties of the NARX model. The RMSE values for T02, T04, PRC and N

were 0.7%-4.1%, 0.8%-2.6%, 4.6%-5.6%, and 3.0%, respectively. Given that the NARX model was trained only with the three input variables of experimental data. The author attributed the slight variation between numerical values generated by the network and the experimental values was as result of in accuracy of the measurement systems (examples: sensors and transmitters) without treating this effect in isolation to understand the extent of its error or damage. The researcher could have studied other types of neural networks that could outperform the NARX network with a lower RMSE in depicting the dynamic model of the IPGT. He could have also performed the deterioration of the IPGT using fouling and erosion as a case study and comparing it with the basal value or developed a steady state model of the IPGT using SIMULINK and NARX models respectively.

Masdi et al (2015) investigated a frame work for short term performance prediction of gas turbine which he did by developing an intelligent prognostic model based on the ELMAN recurrent neural network (ERNN). The data was gotten by taking a daily average of four month operational data which was assumed to represent the steady state operation of the plant. The purpose of the network is to forecast four-step-ahead by using three series of past historical data. Obtained information was screened to remove any transient data during start up or shutdown of turbine operation which would otherwise reduce the accuracy of prediction of the final model. The RNN predictors are trained using past machine measurements that have been collected. Calculation of available parameters was impossible in the study due to a lack of available measurements. As a result, the data fusion method was employed to propose a new efficiency indicator (EI) parameter as a prognosis criterion. The authors assert that when the prediction interval grows, the model's accuracy falls. The average normalized error (ANE) was used to determine the accuracy of the forecast. In other words, the average ANE error in the gas turbine application utilizing this proposed model to predict the performance of a gas turbine unit 10 days in advance is 3.24 percent, which is considered acceptable.

Furthermore, these findings show that this approach is better suited to short-term forecasting. The researchers could have used the Jordan Multi step ahead recurrent neural network, as well as compared other multi step networks to the RNN to discover which is more practical for short and long-term prediction.

Yahao et al (2018) studied static and dynamic performance modelling and simulation of a micro turbine. The micro turbine T100 from Turbec AB is a combined heat and power (CHP) generation system which could produce 100KW of electric power. The architecture of the model was divided into compressor, burner, turbine, recuperator, exhaust diffuser and a power generator with a single shaft connecting the turbine, compressor and power generator. Component models were built according to turbine performance maps. Then the operating points on the maps were solved using the Newton Raphson method. The models were built in Simulink and with components from the common turbo machinery block of T-MATS. The map data was adopted from the GASTURB software. The dynamic mathematical model of the recuperator was reduced from a set of partial differential equations to a set of ordinary differential equations. The parameters used were: pressure ratio, corrected shaft speed, and corrected mass flow rate, gas mass flow through the compressor, the compressor pressure ratio and the compressor efficiency. This was calculated from the map and the thermodynamic parameters of the outlet gas are further calculated by the data obtained from the map. Steady state data were verified with acceptable results. From the results of the static verification, the largest error was in the exhaust gas temperature (0.8%). It also showed that the dynamic validation in the part load experiments had a good match with the measured data. From the results it showed 2-4% error deviation when the complete micro turbine system was analysed. The shaft speed had a relative error 6.82% with operation point of 50%. The researcher claims that the error was as a result of scaling curves of the generic compressor and the turbine map. The researchers did a fantastic job by combining generic

algorithm with the turbine maps to make up for unknown portion of the design; because manufacturers conspicuously keep it as a proprietary material. However, they could have gone through the route of neural networks supervised learning, which makes use of the input, and output variables, as the researcher stated in the work that the large error in the shaft speed was as result of scaling curves of the generic compressor and turbine map.

Thiago et al (2018) investigated predictive control of nonlinear models applied to transient control of gas turbines. The paper tried to examine the application of a nonlinear model-based predictive control (MPC) to a gas turbine. The work tried to present a comprehensive turbine control strategy as a way to prevent unsafe or improper operation. A load release test was performed during gas turbine operation to evaluate speed control in the event of a sudden load collapse without the turbine shutting down. The acceptance criterion for these tests is that the speed should not exceed 10% of the nominal value. In this strategy, a nonlinear dynamic representation of the gas turbine was performed using the DESTUR process simulator. In this work, MPC was implemented in MATLAB integrated with the DESTUR process simulator. The objective function that needed to be minimized here is the error between the predicted system output and the desired value for a given future period (range of control). So the control input (u) is the fuel consumption and the output (y) is the compressor speed. Results were obtained for three simulations of gas turbine control problems which include: fuel consumption/reduction, load removal/addition, and load rejection. Process simulators solved state and conservation equations for mass, energy, and momentum. The MPC strategy was used to control the fuel flow (mf) to the combustion chamber to regulate the compressor speed (N) when the load on the gas turbine changes. During operation with a constant power of 107 MW, after 10 seconds the fuel consumption was gradually optimized from 6.90 kg / s to 6.88 kg / s and the speed was adjusted to a speed of 3600 rpm. The MPC successfully selected an operating point with low fuel consumption while maintaining the

desired rotation. The MPC thus promptly acts, increasing fuel flow until the rotation set point of 3,600 rpm was achieved. The MPC controller was able to select an operating point with reduced fuel consumption while maintaining the desired rotation. For the second case, step disturbances in the load of the gas turbine were applied. At first, a sudden load fall of 10% occurred, reducing it from 107 MW to 96.3 MW, after load removal, the turbine operation stabilized. Consequently, the fuel flow was automatically operated on by the MPC. After stabilization, load is added back, decreasing rotation. The MPC thus promptly acts, increasing fuel flow until the rotation set point of 3,600 rpm was achieved. The third and last scenario was a load rejection, (a sudden load fall) related to power failure due to a lightning strike or mechanical failure. In this case, the power output of 107 MW to the grid is unexpectedly lost at 10 seconds. As a consequence, as indicated, the compressor speed achieves 3,832 rpm, within 10% of its nominal value. Fuel flow was rapidly manipulated downwards, such that the speed went back to 3,600 rpm in the course of the next 11 seconds.

Akilu et al (2016) developed twin shaft gas turbine engine model for performance simulation and validated the model with operating data collected from running gas turbines. Due to limited measurement data, a complete validation of the various components and parameters has not been performed. However, sufficient data were collected to demonstrate the process of validation. A change in the gas turbine operation was done by varying the turbine power output, which in actual operation of the gas turbine causes a reduction in the fuel flow rate to the combustor. Hence, the power output and ambient temperature are used as input for simulation. The gas generator turbine was modelled by employing component characteristic map similar to compressor modelling. The compressor pressure ratio, speed, and flow rate were known, the missing design parameters namely, turbines inlet temperatures and pressure ratios were predicted using GasTurb simulation software. The predicted parameters were

used to determine the exit temperature and pressure of the compressor and then the compressor work was calculated. The compressor exit pressure was observed to be proportional with the power output. The mean percentage error between the simulation and operational data is around 3.8%. The compressor exit temperature increases with the power output. The comparison between the simulation and operational data showed good agreement. The average error between simulated and operational data was about 5%. The efficiency of a gas turbine was found to increases in proportion to the power of the turbine. This was expected as the maximum efficiency was about 29.2%. Specific fuel consumption (SFC) is also found to be inversely proportional to efficiency. The Researchers could have adopted optimization algorithms when designing compressor and turbine maps rather than the use of GASTURB software to obtain unknown parameters. This is because the marginal error between the real and simulated values is too severe.

Chibakhsh and Amirkhani (2018), developed a simulation model for transient behavioural analysis of V94.2 industrial gas turbine. A mathematical model was developed based on semi-empirical correlations for mass and energy balance, heat transfer, and thermodynamic state transformation. The model parameters were adjusted by applying a method of optimizing a genetic algorithm (GA) to data obtained from real system functions. For this purpose, the objective function was defined as the mean of the squared difference between the set point and the actual value of the compressor compression ratio, the compressor outlet temperature, gas fuel consumption, net unit thermal efficiency, corrected outlet temperature and power generation. Finally, the function of the developed model was evaluated by comparing the response of the developed model with that of the real system under static and transient conditions. The GA learning process was repeated until the target function fell below the desired threshold. The optimization process was performed with MATLAB optimization toolbox. Actual system functions, including transient and constant static

conditions, were acquired with a sampling time of 1 second and recorded as three different sets of data. Pre-processing of the recorded data was performed to estimate the time delay for the input variables. At that stage, the initial estimate was determined according to the time constant. The delay parameters were adjusted based on the observed time offset differences for the recorded and simulated signals. Through optimization, unknown parameters related to different parts of the gas turbine were obtained. The accuracy of the developed model was evaluated by comparing the model solution with the real plant solution. To this end, the author calculated the deviation, including mean absolute percent error (MAPE) and mean absolute error (MAE), derived from the training and test datasets. The author further claimed that compressor performance depends on CPR deviation from its nominal value had a prediction error of less than 0.5%. The estimated value for OTC (Outlet Temperature Corrected) was compared with its actual value. The obtained results indicate the accuracy of the developed model as the OTC value agrees with the real system data. The maximum deviation was less than 3.5 K. The air mass flow rate and the overall generated power by the gas turbine while the unit load ramped up from 35% to full-load conditions were presented. The researchers could have extended their research to simple cycle gas turbines as the turbine chosen for the work was a combined cycle turbine; because turbine exhaust temperature was not allowed to vary due to changes in the disturbance of the system and was pegged at 850 kelvin while the net power was allowed to vary, even at steady state. Moreover; they could have carried out a comparative study by juxtaposing the simulation from GasTurb software and the result of simulation from the research work.

Ren et al (2018) worked on static and transient performance modelling and simulation of a micro turbine. The experimental data was used to match the operation of the model at part load and the dynamic model was supported by using genetic algorithm to optimize map

characteristics. Performance maps are exclusively proprietary of OEM therefore they were restricted by limited access to qualitative inputs and outputs data and was constrained to generic turbine and compressor maps which adversely affected the result.

## 2.8 Summary of Literature Review

Gas turbine modelling and simulation has recently received a lot of attention from researchers, engineers and OEM manufacturers. With the increasing demand for gas turbines in modern power generation, considerable research has been done over the years, especially in the modelling and simulation of gas turbines. Although, works on transient modelling of turbines have been done by Asgari 2014a, Chibakhsh and Amirkhani 2018, Thiago et al 2018. Further research still needs to be done by exploring other ANN's for better performance with the end goal of increasing reliability. Artificial neural network models can be broadly investigated using real life operational data from gas turbine plant. ANN models can be extensively explored by using real-world operational data from gas turbine plants. By examining various ANNs ((layer recurrent, feed forward back propagation and nonlinear auto regressive network with exogenous input) for dynamic gas turbine modelling in-order to create a model which can be able to predict dynamic start-up behaviour of the system as precise as possible which model can also be used as a reliable tool for maintenance, trouble shooting and condition monitoring of the gas turbine. As can be observed from the works of literature, none of the authors were able to carry out comparative studies using different artificial neural network types.

# CHAPTER THREE

# MATERIALS AND MEHTODS

## 3.1 Model Description

The turbine chosen for this experiment is the GT13E2 single shaft power gas turbine. The model specification by design of the General Electric (formerly Alston) GT13E2 power gas turbine is:

**Table 3.1:** Model specification of GT13E2 gas turbine (Afam Power Plant, 2019)

| Specification | Minimum Value | Maximum Value |
|---|---|---|
| Efficiency | - | 38% |
| Turbine outlet temperature | $0^{O}C$ | $700^{O}C$ |
| Compressor outlet temperature | $0^{O}C$ | $600^{O}C$ |
| Load generator active power | -18MW | 198MW |
| Flow rate | 0 Kg/s | 15 Kg/s |
| Rotational speed | 0 RPM | 3000 RPM |
| Compressor inlet temperature | $-40^{O}C$ | $80^{O}C$ |
| Compressor inlet pressure | 60 KPA | 110 KPA |
| Compressor outlet pressure | 0 KPA | 2000 KPA |

## 3.2 Start-up Procedure

Sequel to the collection of the start-up result, the GT13E2 gas turbine undergoes the following start up processes:

**Starter On**: Starting of the GT13E2 gas turbine engine often requires an external source to overcome the initial starting torque required for self-sustenance. If a generator is selected as a starting means, then it behaves like a motor and draws current from the grid to perform the start-up operation. In accordance with a preferred embodiment, a generator circuit breaker, step up transformer, and a grid high voltage transformer are arranged between the static frequency converter and the grid in order to output power to the grid, and starting transformer and start-up switch are arranged parallel to the generator circuit breaker for power consumption. Sequential checks are also carried out if the system indicator gives permission

after these checks, then the SFC takes alternating current via the electrifier module and converts it to direct current. The DC is sent directly to the generator. The generator takes the DC on the generator phase via the commutator supplement and the carbon brush connected to the slip rings which creates a magnetic phase. The presence of a magnetic phase on the generator rotor creates electrical torque that rotates the shaft of the generator

**Purge**: Purging is a safety measure that removes any combustible gas/fuel from the engine system before it comes in contact with a hot source such as ignition. In a typical engine the starting sequence is marked as the initiation of purging, starting of lubrication oil pump, and turning of the cooling fan if necessary. Purging is required to flush out and choke out liquid fuel nozzles which may have traces of fuel and residues which are left out in the combustion chamber after the shutdown operation of the turbine.

**Crank**: This is a process whereby the turbine is made to run at required speed during start up and shutdown in order to prevent the turbine rotor from hogging and sagging. The starting mean system is used to start the gas turbine rolling, crank it to firing speed and assist the fired turbine to self-sustaining speed

**Firing**: When admission of fuel in the combustion chamber takes place firing occurs. During this period the gas turbine will have firing status. Prior to this, the control system carries out protection checks on the lube oil system, fuel gas valves opens if all checks has been passed and the fuel flows to the combustion chamber for firing or flame on. The thermal heat of combustion is converted to mechanical energy that drives the turbine shaft.

**Full speed part load/Full speed full load**: As the electrical energy is carrying out its function to drive the generator shaft the thermal energy generated from the combustible fuel is also driving the turbine shaft waiting for 95% nominal speed, so that the SFC can disconnect automatically; because at that moment both systems have succeeded in driving

the turbine to self-sustaining speed. So at that point SFC will automatically switch off from receiving current from the grid, while the turbine is being powered by propane combustion processes and at that point the turbine has attained synchronous speed i.e. it has attained the ability to run on its own. It does not need the external support via the electrical system. As the unit attains self-sustaining speed the load controller gets activated to load up the unit to its base/maximum load or to any load the national control centre instructs the control room to maintain in order to prevent load overflow from the transmission network.

### 3.3 Data Acquisition and Preparation

The data sets used for model set-up and verification was taken experimentally during several start-up manoeuvres. Therefore, these data are representative of the operating conditions during start-up and account for all the conditions related to this type of transient operation (e.g. bleed valve opening, IGV control, etc.). In general, the data sets during start-up can be categorized as:

• Cold start-up: The gas turbine was shut down some days before start-up.

• Warm start-up: The gas turbine was shut down some hours before start-up.

• Hot start-up: The gas turbine was shut down just few hours or less before start-up.

The data sets to be used in this study refer to hot start-up.

### 3.4 Measured Parameters

The parameters measured directly from the IPGT include:

1. Rotational speed (RS),
2. Load/alternator power (MW),
3. compressor inlet stagnation pressure(CIP KPA) ,

4. compressor inlet temperature (CIT K) ,
5. compressor outlet stagnation pressure (COP KPA) ,
6. compressor outlet temperature (COT K) ,
7. turbine outlet temperature (TOT K) , and
8. fuel flow rate ( FR).

### 3.5 Data acquisition protocol

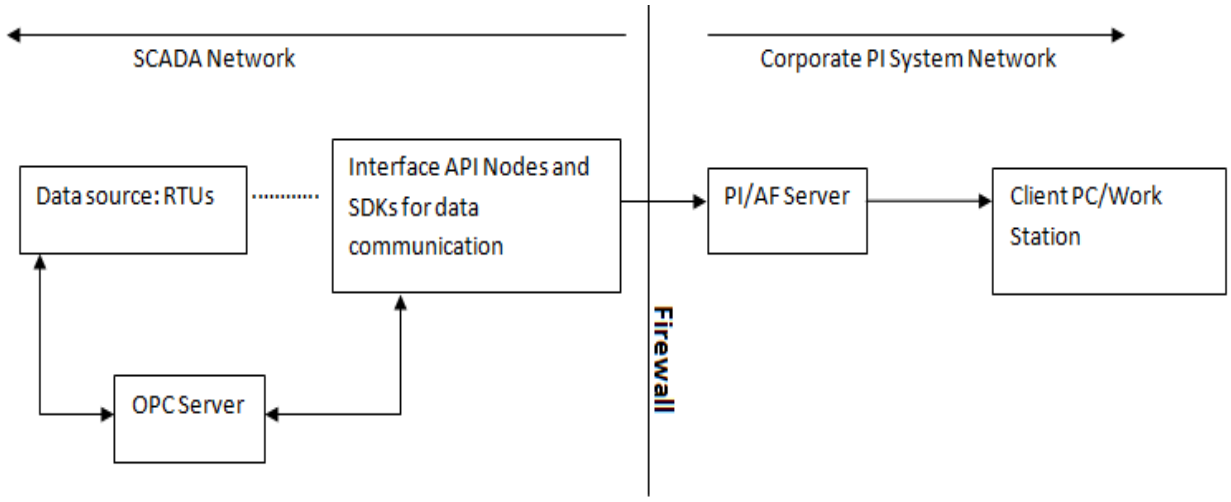



**Figure 3.1:** Data acquisition flow sheet from Afam power plant.

SCADA applications in the plant are comprised of two elements, namely, the turbine machinery system that needs to be controlled and an interconnection of devices which form an interface with the process under consideration via sensors and control outputs. Data acquisition begins at the RTU or PLC level and includes meter readings and equipment status reports that are communicated to SCADA as required. RTUs are connected to SCADA systems by means of wireless Area Network (WAN). It relies on microcomputers called programmable logic controllers (PLCs) and remote terminal units (RTUs) at its' core in the turbine plant.

An open source OPC server bridges the communication gap between the data source from the RTUs and the interface node of the SCADA network. The SCADA systems have application program interface (API) and software development kit (SDK) to aid in the ease of transfer of data from its data source to the process information (PI) system server. In order to collect data into the data historian an interface node is introduced. The interface node will run some PI software and it's able to communicate with a proprietary system. The PI API node interface collects the information from the data source (RTU) and puts it into the snapshot table. The process information system has a real time data historian which sends digitized information in real time, also it automatically compiles backlogs of all collected data to allow for trending and other analytical auditing. The snapshot receives the current value and its corresponding time stamp. No specific tag limit exists for the interface. Engineer at the plant workstation configures the interface to copy exception (snapshot) or archive data. The snapshot table harbours as many tags in the PI server it could range from 500-3000000 tags (values). As data come into the snapshot table, the PI server compresses the data and stores the record for each tag in the archive records. When the system integration has been done, the interface node will put the data into the data historian through the asset framework (AF) and the client PC/ Work station can access the data.


**Figure 3.2**: tags and values transfer from snapshot table to archive

### 3.6 ANN Methodology

Matrix laboratory (MATLAB 2019a) tool is adopted for building and simulating the ANN models due to the plethora of literature in ANN available with using this tool and its ease of use makes it the tool of choice in numerical computing. The networks is built in "NNstart" graphical user interface (GUI) after which the GUI was used to generate command line scripts for other MISO and MIMO models.

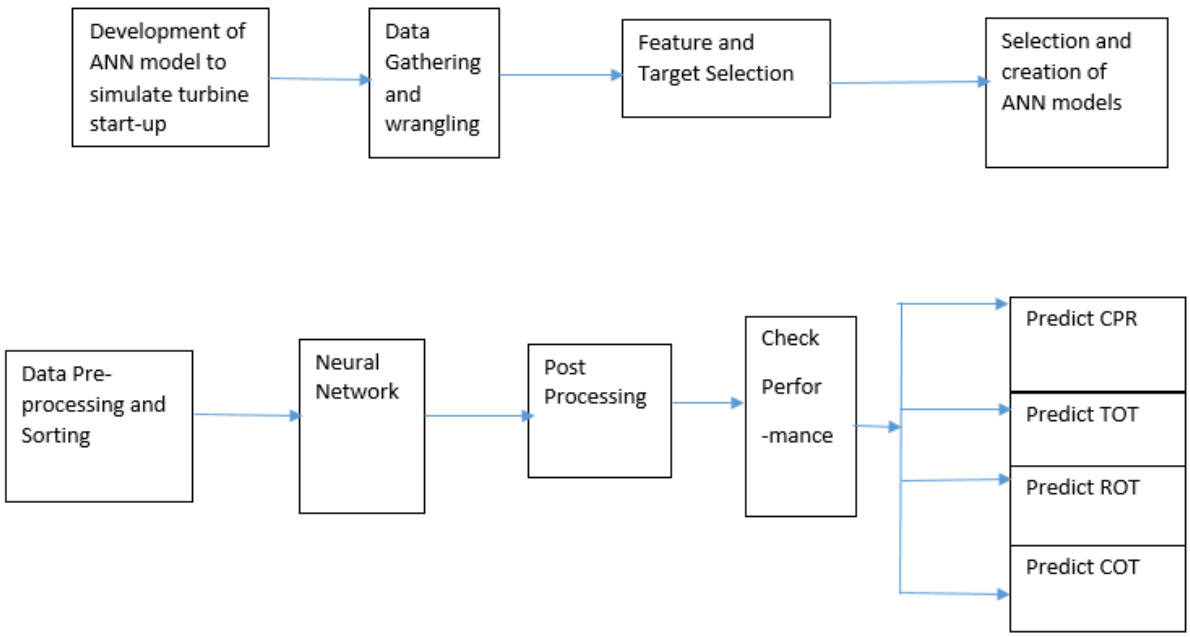



**Figure 3.3**: Flow chart of ANN model development.

**Development of ANN model to simulate turbine start-up**: MIMO and MISO models is developed for turbine start-up. However; the performance of the single output and multi output models was obtained from which decision was made to determine the model structure that best suits turbine transient operation.

**Data gathering and wrangling**: The obtained data was queried from FIPL/Afam plant data base and data of 1048573 seconds in step size was obtained equivalent to 12.14 days of operational data. Five start up data was identified in the queried data. Out of the five start up data 1 is selected for MIMO and MISO modelling because it covers the five different regimes of transient turbine operation which aids in generalization of the ANN model. However; the fifth data was used for simulation with the created neural network MIMO and MISO models to test how well it was able to generalize. The start-up data has 1067 data points while the test data has 480 data points. It was observed that the queried data lacked unit dimensional consistency. Therefore; temperature was changed from "$^{o}$C" to Kelvin while pressure was converted from mbar and bar to "kPa" for unit dimensional consistency.

**Feature and target selection:** Four dependent variables which includes compressor outlet temperature in Kelvin (COT), turbine outlet temperature in Kelvin (TOT), rotational speed in revolutions per minute (ROT) and compressor pressure ratio (CPR) were selected as targets or dependent variables while compressor inlet temperature in Kelvin (CIT), fuel flow rate in kg/s (FR) compressor inlet pressure in kPa (CIP) and load in MW were selected as features or independent variables.

**Selection and creation of ANN models**: Feed forward back propagation network, non linear autoregressive network with exogenous input and layer recurrent networks was selected because they are designed to work with time series data for prediction and modelling. Sensitivity analysis was done by Asgari (2014) with similar features and targets. In the study, the author suggest the use of one hidden layer, 12 hidden neurons, tapped delay line of 1-2 seconds at the input. In addition a regressed outputs of 2 seconds backwards at the point t-1 and t-2 is suggested for good performance. "Trainlm" was used as the training function because it updates weights and bias values according to Levernberg- Maarquardt optimization. The training function is the fastest back propagation algorithm and often recommended for supervised learning task (Mathworks.com, n.d).

**Data pre-processing, sorting and post processing**: Due to the extreme range of temperature values and rotational speed relative to flow rate and compressor pressure ratio normalization and scaling was inevitable so as not to hinder the performance of our model. To ensure that the training process is optimized some pre-processing stages was performed on the inputs and targets. Normalization was carried out to prevent over saturation of the hyperbolic tangent sigmoid transfer (tansig){Citation} function (which will be utilized).

Mapminmax MATLAB function was called in "MATLAB" so that inputs and targets will fall in -1 and +1 range. The min-max normalization is given by the statement:

$$x^1 = ((x - xmin)/(xmax - xmin)) * (u - l) + l \quad 3.1$$

Given that: x is the original feature or target data set and $x^1$ is the normalized data set

xmin and xmax are the minimum and maximum values of the actual vector/data set,

u, l are the lower and upper values of the new span of normalized data, for mapminmax normalization u = 1, l = -1

Furthermore “removeconstantrows” MATLAB function was adopted to removed rows in input and target with constant values.

**Neural network:** Feed forward back propagation networks (FF BPP), layer recurrent (LayRecnet) and non-linear autoregressive network with exogenous input (NARX) was selected. Because the ANN technique is a black box approach to modelling nonlinear systems details of the weights and bias value are unknown. MATLAB was used to visualize the NARX and LayRecnet networks but the FF BPP wasvisualized in R studio software because it displays the relationship between the weights and biases along with the features and hidden neurons. The data was imported from Microsoft excel and was converted to matrix column vectors using xlsread to read excel data and storing them in assigned variables. Furthermore; feature and target matrices were randomly divided into 70:15:15 percent ratios. 70 percent is used for model training, 15% was used to validate generalization and to ensure training stops before overfiting while the remaining 15% was used as a wholly nondependent test of network generalization. A sigmoid transfer function was set in the hidden layer while a liner transfer function is present in the output layer to determine layers output from net input.

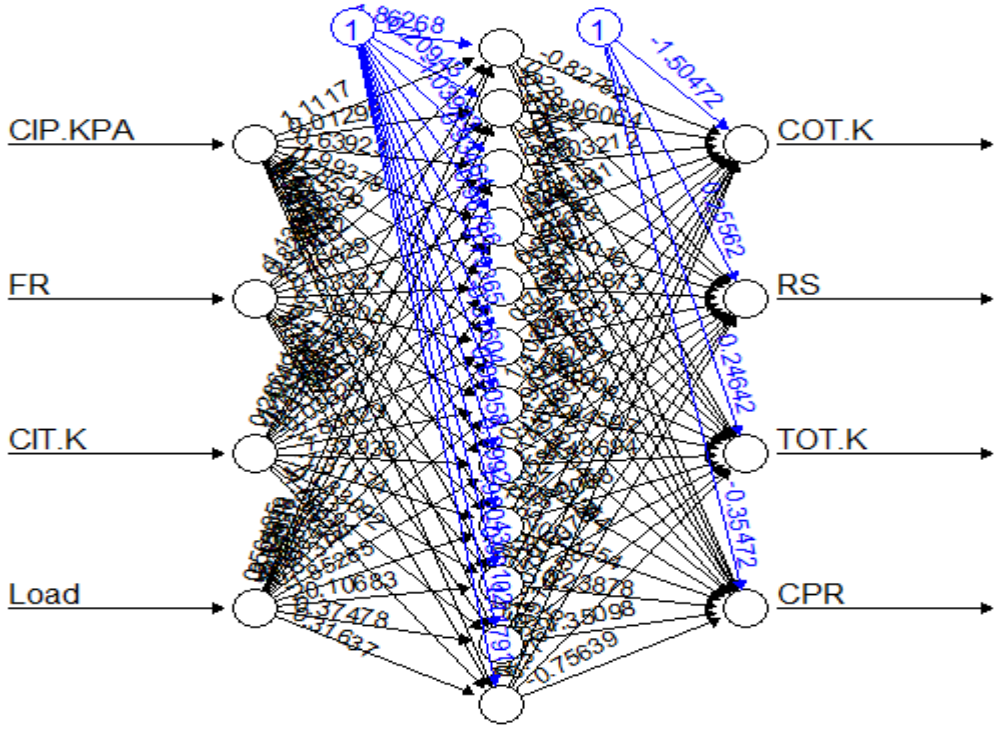



**Figure 3.4:** Trained FF BPP MIMO network in Rstudio showing weights and bias connections.

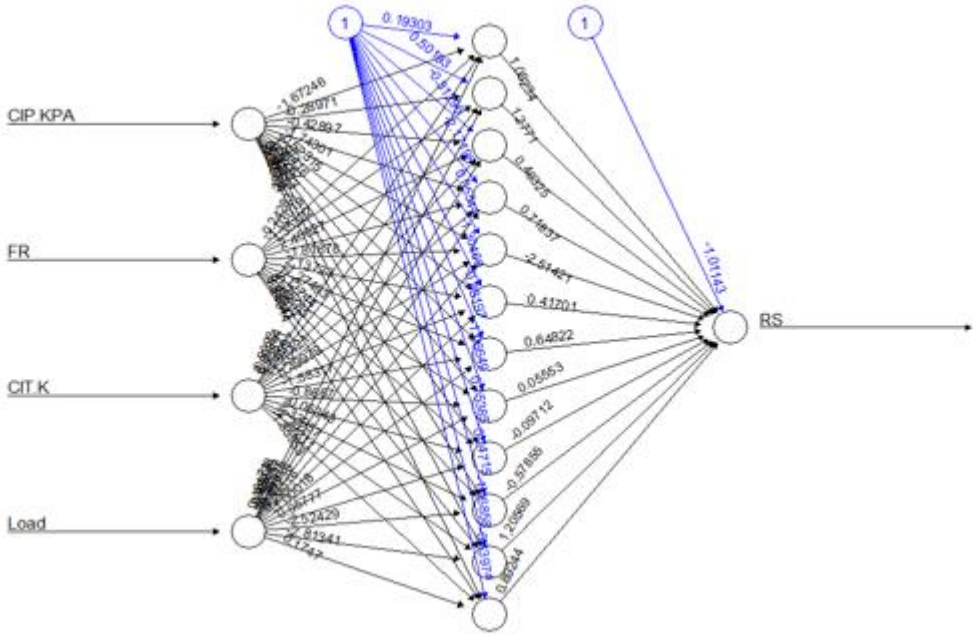



**Figure 3.5:** Trained FF BPP MISO network in Rstudio showing weights and bias connections.

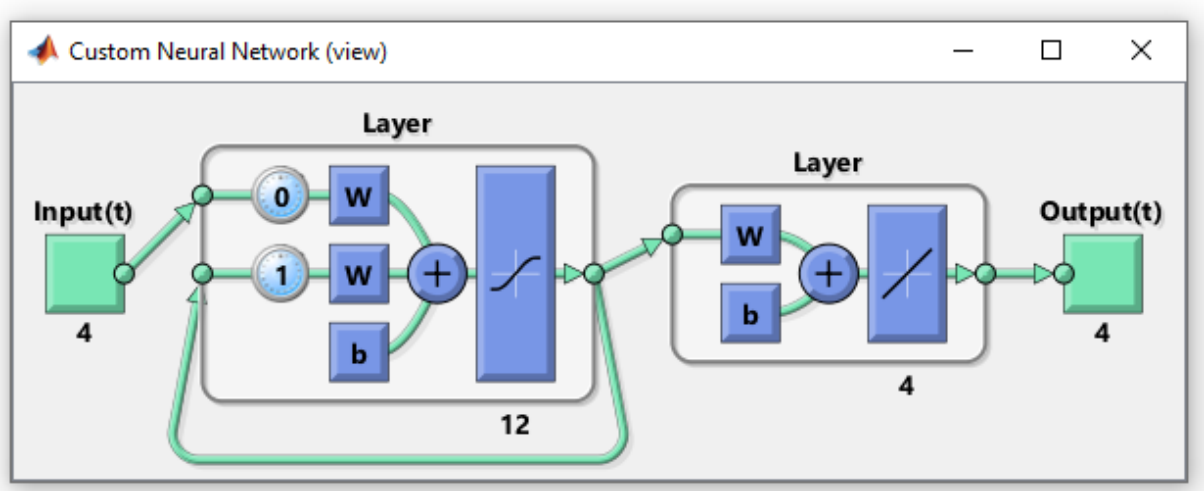



**Figure 3.6:** MIMO layrecnet with tapped delay at the hidden layer.

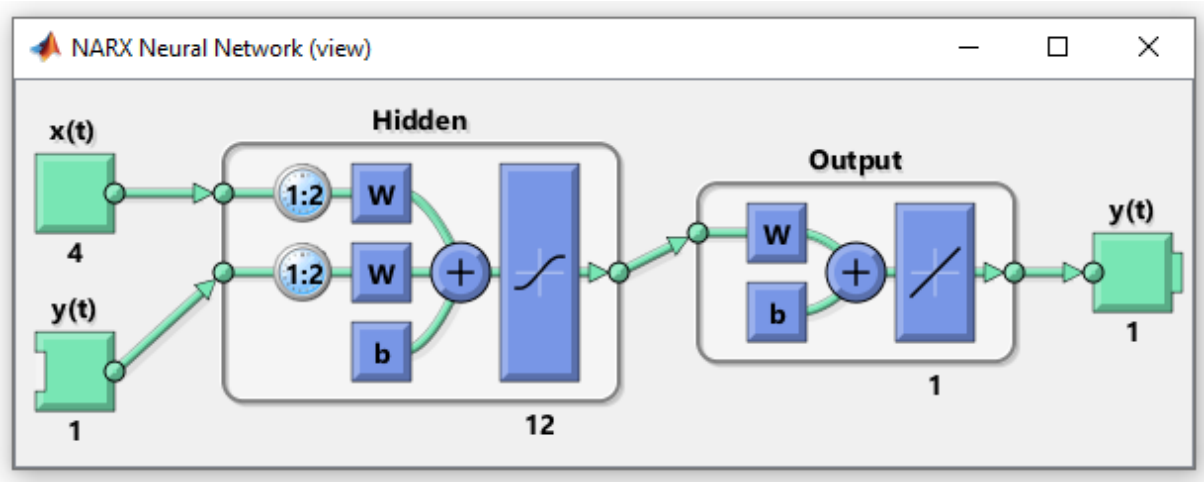



**Figure 3.7:** An open loop NARX MISO network in MATLAB with input and output delay.

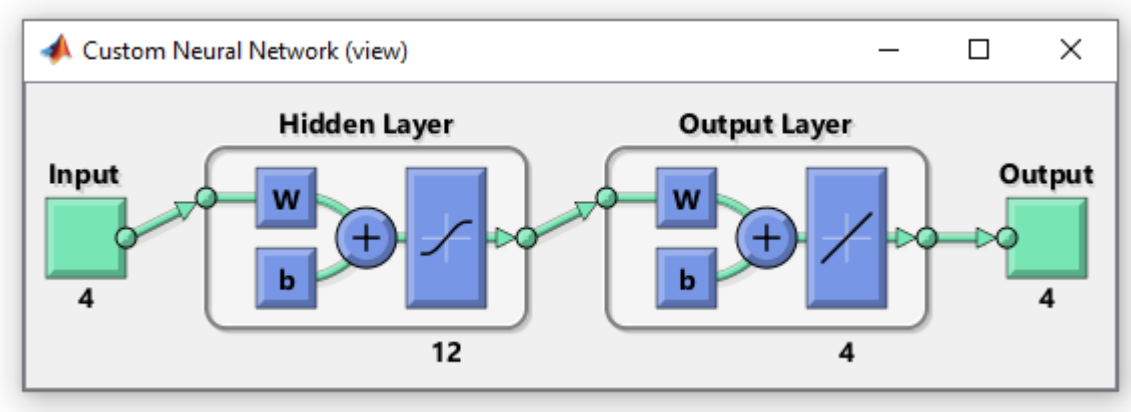



**Figure 3.8**: FF BPP MIMO network in MATLAB.

**Determine performance:** The performance of the neural network model was measured with mean of squared error (MSE) and mean of absolute percentage error (MAPE). The mean squared error is calculated by the average of the squared predicted error values. Squaring the predicted error values forces it to be positive however the MAPE calculates forecasted error as a percentage and measures absolute percent error for every time step subtracted from the targets and divided by the targets.

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{\text{Actual}_i - \text{Forecast}_i}{\text{Actual}_i} \right| * 100 \quad 3.2$$

$$\mathbf{MSE} = \frac{\mathbf{1}}{\boldsymbol{n}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} (\boldsymbol{y_i} - \hat{\boldsymbol{y}}_{\boldsymbol{i}})^{\mathbf{2}} \quad 3.3$$

### 3.7 Simulation with Test Data

The model with the best performance in MSE and MAPE was used for simulation with the test data that has not been seen or used by the three networks.

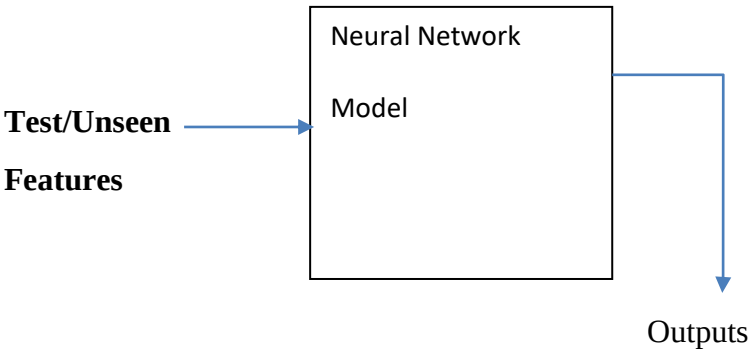



**Figure 3.9:** Flowchart of model simulation with unseen data.

# CHAPTER FOUR

# RESULTS AND DISCUSSION

## 4.1 Result for Starting Manouvers

The queried data is an accumulation of about 13 days of turbine operation. The data before being put to use, some cleaning processes were carried out on it such as making the variables unit dimensionally consistent, merge, and splitting operations

.

**Table 4.1:** Top (10) heads of raw operational data

| Time | EXHAUST TEMPERATURE | ACTIVE POWER | COMPRESSOR OUTLET TEMPERATURE | MASSFLOW FUEL GAS | TURBINE SPEED | ATMOSPHERE TEMP | ATMOSPHERE PRESSURE | COMPRESSOR OUTLET PRESSURE |
|---|---|---|---|---|---|---|---|---|
| Hr/min/sec | 0 | -18 | 0 | 0 | 0 | -40 | 600 | 0 |
| | 700 | 198 | 600 | 15 | 3000 | 80 | 1100 | 20 |
| | degC | MW | degC | kg/s | RPM | degC | mbar | bar |
| 17:29:23 | 476.198 | 89.759 | 377.971 | 6.229 | 3023.492 | 28.941 | 1007.617 | 10.283 |
| 17:29:24 | 476.19 | 89.751 | 377.963 | 6.229 | 3023.459 | 28.939 | 1007.613 | 10.283 |
| 17:29:25 | 476.182 | 89.743 | 377.955 | 6.229 | 3023.426 | 28.937 | 1007.609 | 10.283 |
| 17:29:26 | 476.174 | 89.735 | 377.947 | 6.228 | 3023.392 | 28.935 | 1007.606 | 10.283 |
| 17:29:27 | 476.165 | 89.726 | 377.939 | 6.228 | 3023.358 | 28.934 | 1007.602 | 10.283 |
| 17:29:28 | 476.157 | 89.718 | 377.93 | 6.227 | 3023.325 | 28.932 | 1007.598 | 10.283 |
| 17:29:29 | 476.149 | 89.71 | 377.922 | 6.227 | 3023.292 | 28.93 | 1007.595 | 10.282 |
| 17:29:30 | 476.141 | 89.702 | 377.914 | 6.227 | 3023.258 | 28.928 | 1007.591 | 10.282 |
| 17:29:31 | 476.133 | 89.694 | 377.906 | 6.226 | 3023.225 | 28.927 | 1007.587 | 10.282 |
| 17:29:32 | 476.125 | 89.686 | 377.898 | 6.226 | 3023.191 | 28.925 | 1007.583 | 10.282 |
| 17:29:33 | 476.117 | 89.677 | 377.89 | 6.225 | 3023.158 | 28.923 | 1007.58 | 10.282 |

**Table 4.2:** Tail (10) values of raw operational data

| Time | EXHAUST TEMPERATURE | ACTIVE POWER | COMPRESSOR OUTLET TEMPERATURE | MASSFLOW FUEL GAS | TURBINE SPEED | ATMOSPHERE TEMP | ATMOSPHERE PRESSURE | COMPRESSOR OUTLET PRESSURE |
|---|---|---|---|---|---|---|---|---|
| Hr/min/sec | 0 | -18 | 0 | 0 | 0 | -40 | 600 | 0 |
| | 700 | 198 | 600 | 15 | 3000 | 80 | 1100 | 20 |
| | degC | MW | degC | kg/s | RPM | degC | mbar | bar |
| 21:02:52 | 518.872 | 150.372 | 406.303 | 8.887 | 3017.405 | 24.983 | 1009.966 | 13.541 |
| 21:02:53 | 518.861 | 150.36 | 406.298 | 8.884 | 3017.398 | 24.98 | 1009.963 | 13.54 |
| 21:02:54 | 518.85 | 150.349 | 406.294 | 8.882 | 3017.39 | 24.978 | 1009.959 | 13.54 |
| 21:02:55 | 518.839 | 150.337 | 406.29 | 8.88 | 3017.383 | 24.976 | 1009.956 | 13.54 |
| 21:02:56 | 518.829 | 150.326 | 406.286 | 8.878 | 3017.375 | 24.973 | 1009.953 | 13.539 |
| 21:02:57 | 518.818 | 150.314 | 406.281 | 8.875 | 3017.367 | 24.971 | 1009.949 | 13.539 |
| 21:02:58 | 518.807 | 150.302 | 406.277 | 8.873 | 3017.36 | 24.969 | 1009.946 | 13.538 |
| 21:02:59 | 518.796 | 150.291 | 406.273 | 8.871 | 3017.352 | 24.966 | 1009.942 | 13.538 |
| 21:03:00 | 518.785 | 150.279 | 406.269 | 8.869 | 3017.345 | 24.964 | 1009.939 | 13.537 |
| 21:03:01 | 519.457 | 150.994 | 406.58 | 9.027 | 3018.266 | 25.028 | 1010.092 | 13.559 |
| 21:03:02 | 519.448 | 150.986 | 406.573 | 9.021 | 3018.239 | 25.026 | 1010.089 | 13.559 |

Results for the gas turbine consist of  time series data which is a representation of the starting manoeuver

Data 1

No of samples: 1067

Step time: 1s

**Table 4.3:** Start – up description

| Start – up/dynamic state | Status |
|---|---|
| Grid Power Source | Yes |
| Purging | No |
| Cranking | No |
| Firing/Fuel Valve Open | Yes |
| Full Speed No Load | Yes |
| Full Speed Part Load | Yes |

**Table 4.4:** Top (18) heads of cleaned sample 1 data

| TIME SERIES DATA FOR SAMPLE 1 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| S/N | CIP KPA | FR | CIT K | Load | RS | COT K | TOT K | CPR |
| 1 | 101.151 | 0.316 | 299.762 | -0.194 | 678.271 | 435.845 | 460.235 | 0.17993 |
| 2 | 101.151 | 0.322 | 299.764 | -0.194 | 681.973 | 435.467 | 462.201 | 0.18388 |
| 3 | 101.151 | 0.328 | 299.765 | -0.195 | 685.675 | 435.09 | 464.167 | 0.18685 |
| 4 | 101.151 | 0.334 | 299.766 | -0.195 | 689.378 | 434.712 | 466.133 | 0.19081 |
| 5 | 101.15 | 0.34 | 299.768 | -0.195 | 693.08 | 434.334 | 468.099 | 0.19476 |
| 6 | 101.15 | 0.347 | 299.769 | -0.196 | 696.783 | 433.956 | 470.064 | 0.19773 |
| 7 | 101.15 | 0.353 | 299.771 | -0.196 | 700.485 | 433.579 | 472.03 | 0.20168 |
| 8 | 101.15 | 0.359 | 299.772 | -0.197 | 704.187 | 433.201 | 473.996 | 0.20465 |
| 9 | 101.15 | 0.365 | 299.773 | -0.197 | 707.89 | 432.823 | 475.962 | 0.2086 |
| 10 | 101.15 | 0.371 | 299.775 | -0.197 | 711.592 | 432.445 | 477.928 | 0.21256 |
| 11 | 101.149 | 0.378 | 299.776 | -0.198 | 715.294 | 432.067 | 479.894 | 0.21552 |
| 12 | 101.149 | 0.384 | 299.777 | -0.198 | 718.997 | 431.69 | 481.859 | 0.21948 |
| 13 | 101.149 | 0.39 | 299.779 | -0.199 | 722.699 | 431.312 | 483.825 | 0.22244 |
| 14 | 101.149 | 0.396 | 299.78 | -0.199 | 726.402 | 430.934 | 485.791 | 0.2264 |
| 15 | 101.149 | 0.402 | 299.782 | -0.199 | 730.104 | 430.556 | 487.757 | 0.23035 |
| 16 | 101.149 | 0.409 | 299.783 | -0.2 | 733.806 | 430.178 | 489.723 | 0.23332 |
| 17 | 101.148 | 0.415 | 299.784 | -0.2 | 737.509 | 429.801 | 491.688 | 0.23728 |
| 18 | 101.148 | 0.421 | 299.786 | -0.201 | 741.211 | 429.423 | 493.654 | 0.24024 |

**Table 4.5:** Tail (18) values of cleaned sample 1 data

| S/N | CIP KPA | FR | CIT K | Load | RS | COT K | TOT K | CPR |
|---|---|---|---|---|---|---|---|---|
| 1049 | 101.165 | 4.745 | 299.592 | 40.771 | 3023.41 | 628.366 | 639.557 | 9.20868 |
| 1050 | 101.165 | 4.745 | 299.592 | 40.761 | 3023.28 | 628.359 | 639.553 | 9.2077 |
| 1051 | 101.165 | 4.744 | 299.592 | 40.75 | 3023.16 | 628.353 | 639.548 | 9.20772 |
| 1052 | 101.165 | 4.744 | 299.591 | 40.74 | 3023.04 | 628.346 | 639.544 | 9.20773 |
| 1053 | 101.165 | 4.744 | 299.591 | 40.73 | 3022.92 | 628.339 | 639.539 | 9.20676 |
| 1054 | 101.165 | 4.743 | 299.591 | 40.72 | 3022.8 | 628.333 | 639.534 | 9.20677 |
| 1055 | 101.165 | 4.743 | 299.591 | 40.71 | 3022.68 | 628.326 | 639.53 | 9.20579 |
| 1056 | 101.164 | 4.742 | 299.591 | 40.699 | 3022.56 | 628.32 | 639.525 | 9.20581 |
| 1057 | 101.164 | 4.742 | 299.591 | 40.689 | 3022.44 | 628.313 | 639.521 | 9.20483 |
| 1058 | 101.164 | 4.741 | 299.59 | 40.679 | 3022.31 | 628.306 | 639.516 | 9.20485 |
| 1059 | 101.164 | 4.741 | 299.59 | 40.669 | 3022.19 | 628.3 | 639.512 | 9.20486 |
| 1060 | 101.164 | 4.741 | 299.59 | 40.658 | 3022.07 | 628.293 | 639.507 | 9.20389 |
| 1061 | 101.164 | 4.74 | 299.59 | 40.648 | 3021.95 | 628.287 | 639.502 | 9.20389 |
| 1062 | 101.164 | 4.74 | 299.59 | 40.638 | 3021.83 | 628.28 | 639.498 | 9.20292 |
| 1063 | 101.163 | 4.739 | 299.59 | 40.628 | 3021.71 | 628.274 | 639.493 | 9.20293 |
| 1064 | 101.163 | 4.739 | 299.589 | 40.618 | 3021.59 | 628.267 | 639.489 | 9.20195 |
| 1065 | 101.163 | 4.739 | 299.589 | 40.607 | 3021.47 | 628.26 | 639.484 | 9.20197 |
| 1066 | 101.163 | 4.738 | 299.589 | 40.597 | 3021.34 | 628.254 | 639.48 | 9.20198 |
| 1067 | 101.163 | 4.738 | 299.589 | 40.587 | 3021.22 | 628.247 | 639.475 | 9.20101 |

Test Data

No of samples: 480

Step time: 1s

**Table 4.6:** Start – up description

| Start – up/dynamic state | Status |
|---|---|
| Grid Power Source | Yes |
| Purging | No |
| Cranking | No |
| Firing | Yes |
| Full Speed No Load | Yes |
| Full Speed Part Load | No |

**Table 4.7:** Top (18) heads of cleaned sample Test data

| TIME SERIES DATA FOR SAMPLE TEST | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| S/N | CIP KPA | FR | CIT K | Load | RS | COT K | TOT K | CPR |
| 1 | 101.1526 | 0.254 | 299.748 | -0.19 | 641.247 | 439.623 | 440.577 | 0.144336 |
| 2 | 101.1524 | 0.26 | 299.75 | -0.19 | 644.949 | 439.245 | 442.543 | 0.148291 |
| 3 | 101.1523 | 0.266 | 299.751 | -0.19 | 648.652 | 438.868 | 444.509 | 0.152246 |
| 4 | 101.1521 | 0.272 | 299.753 | -0.191 | 652.354 | 438.49 | 446.475 | 0.155212 |
| 5 | 101.152 | 0.278 | 299.754 | -0.191 | 656.056 | 438.112 | 448.441 | 0.159166 |
| 6 | 101.1518 | 0.285 | 299.755 | -0.192 | 659.759 | 437.734 | 450.406 | 0.162133 |
| 7 | 101.1516 | 0.291 | 299.757 | -0.192 | 663.461 | 437.356 | 452.372 | 0.166087 |
| 8 | 101.1515 | 0.297 | 299.758 | -0.192 | 667.163 | 436.979 | 454.338 | 0.169053 |
| 9 | 101.1513 | 0.303 | 299.76 | -0.193 | 670.866 | 436.601 | 456.304 | 0.173008 |
| 10 | 101.1511 | 0.309 | 299.761 | -0.193 | 674.568 | 436.223 | 458.27 | 0.176963 |
| 11 | 101.151 | 0.316 | 299.762 | -0.194 | 678.271 | 435.845 | 460.235 | 0.179929 |
| 12 | 101.1508 | 0.322 | 299.764 | -0.194 | 681.973 | 435.467 | 462.201 | 0.183884 |
| 13 | 101.1507 | 0.328 | 299.765 | -0.195 | 685.675 | 435.09 | 464.167 | 0.18685 |
| 14 | 101.1505 | 0.334 | 299.766 | -0.195 | 689.378 | 434.712 | 466.133 | 0.190805 |
| 15 | 101.1503 | 0.34 | 299.768 | -0.195 | 693.08 | 434.334 | 468.099 | 0.19476 |
| 16 | 101.1502 | 0.347 | 299.769 | -0.196 | 696.783 | 433.956 | 470.064 | 0.197726 |
| 17 | 101.15 | 0.353 | 299.771 | -0.196 | 700.485 | 433.579 | 472.03 | 0.201681 |
| 18 | 101.1499 | 0.359 | 299.772 | -0.197 | 704.187 | 433.201 | 473.996 | 0.204647 |

**Table 4.8:** Tail (18) values of cleaned Test data

| S/N | CIP KPA | FR | CIT K | Load | RS | COT K | TOT K | CPR |
|---|---|---|---|---|---|---|---|---|
| 460 | 101.1642 | 2.993 | 299.543 | -0.848 | 3023.613 | 601.649 | 545.264 | 8.263793 |
| 461 | 101.1641 | 2.992 | 299.541 | -0.865 | 3023.768 | 601.702 | 545.254 | 8.263801 |
| 462 | 101.164 | 2.992 | 299.539 | -0.882 | 3023.923 | 601.756 | 545.244 | 8.264798 |
| 463 | 101.1639 | 2.991 | 299.536 | -0.9 | 3024.079 | 601.809 | 545.234 | 8.264806 |
| 464 | 101.1638 | 2.99 | 299.534 | -0.917 | 3024.234 | 601.863 | 545.224 | 8.264814 |
| 465 | 101.1637 | 2.989 | 299.532 | -0.934 | 3024.389 | 601.916 | 545.214 | 8.265811 |
| 466 | 101.1636 | 2.988 | 299.53 | -0.951 | 3024.545 | 601.97 | 545.204 | 8.265819 |
| 467 | 101.1635 | 2.987 | 299.527 | -0.969 | 3024.7 | 602.023 | 545.194 | 8.265827 |
| 468 | 101.1635 | 2.986 | 299.525 | -0.986 | 3024.855 | 602.076 | 545.184 | 8.265827 |
| 469 | 101.1634 | 2.986 | 299.523 | -1.003 | 3025.01 | 602.13 | 545.174 | 8.266824 |
| 470 | 101.1633 | 2.985 | 299.521 | -1.02 | 3025.166 | 602.183 | 545.164 | 8.266832 |
| 471 | 101.1632 | 2.984 | 299.519 | -1.037 | 3025.321 | 602.237 | 545.153 | 8.26684 |
| 472 | 101.1631 | 2.983 | 299.516 | -1.055 | 3025.476 | 602.29 | 545.143 | 8.267837 |
| 473 | 101.163 | 2.982 | 299.514 | -1.072 | 3025.632 | 602.344 | 545.133 | 8.267845 |
| 474 | 101.1629 | 2.981 | 299.512 | -1.089 | 3025.787 | 602.397 | 545.123 | 8.267853 |
| 475 | 101.1628 | 2.981 | 299.51 | -1.106 | 3025.942 | 602.451 | 545.113 | 8.267861 |
| 476 | 101.1627 | 2.98 | 299.507 | -1.124 | 3026.098 | 602.504 | 545.103 | 8.268858 |
| 477 | 101.1627 | 2.979 | 299.505 | -1.141 | 3026.253 | 602.557 | 545.093 | 8.268858 |
| 478 | 101.1626 | 2.978 | 299.503 | -1.158 | 3026.408 | 602.611 | 545.083 | 8.268866 |
| 479 | 101.1625 | 2.977 | 299.501 | -1.175 | 3026.563 | 602.664 | 545.073 | 8.269863 |
| 480 | 101.1624 | 2.976 | 299.499 | -1.193 | 3026.719 | 602.718 | 545.063 | 8.269871 |

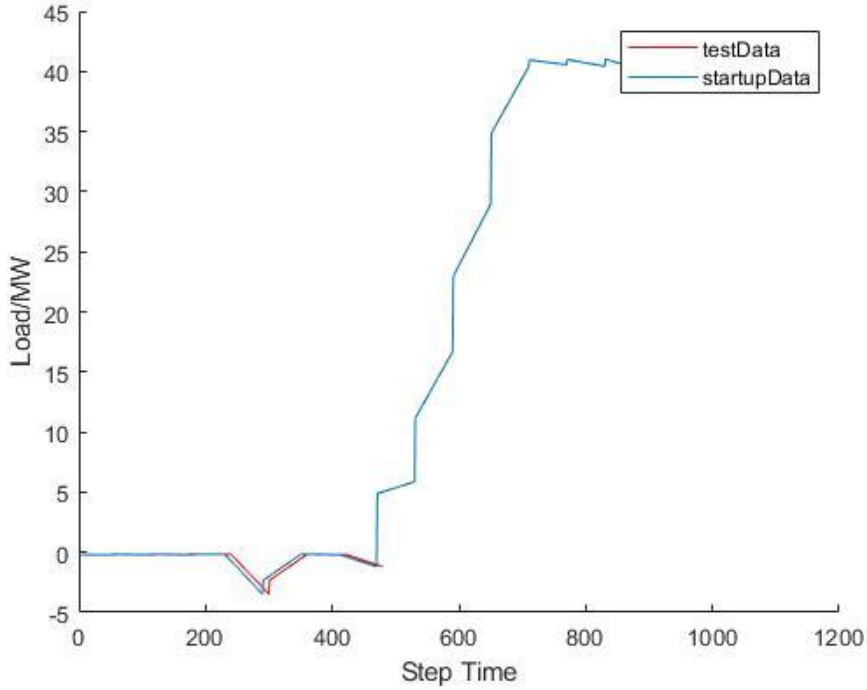



**Figure 4.1:** Variations of load for different data samples/manouvers.

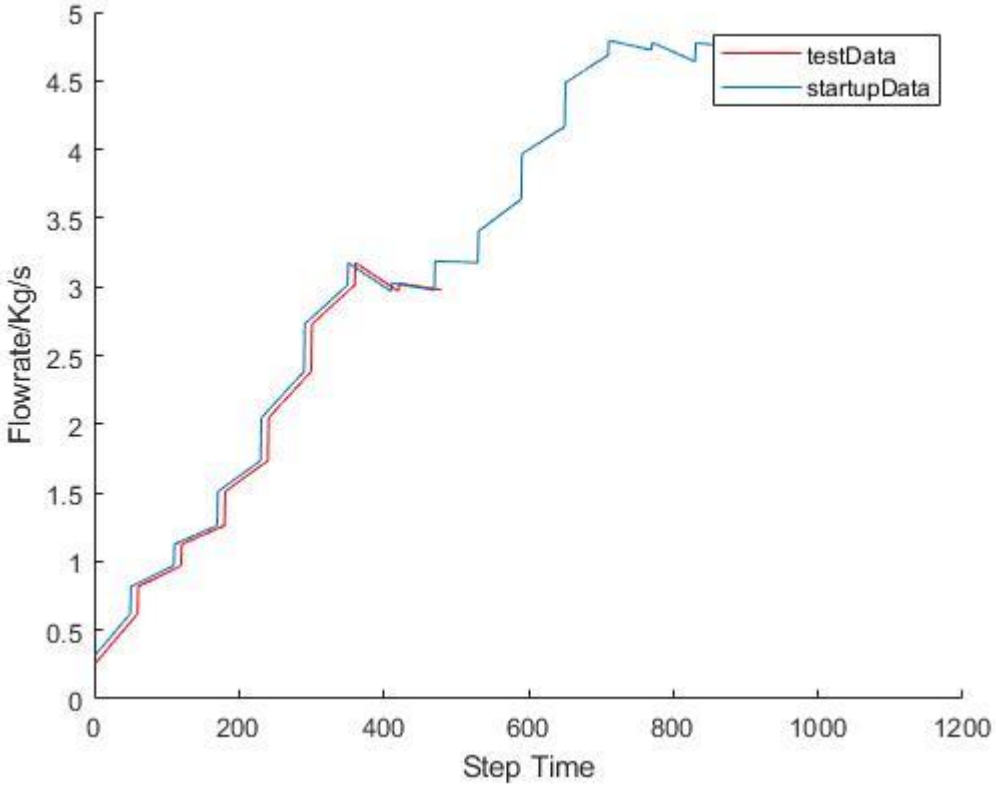



**Figure 4.2:** Variations of flow rate for different data samples/manouvers.

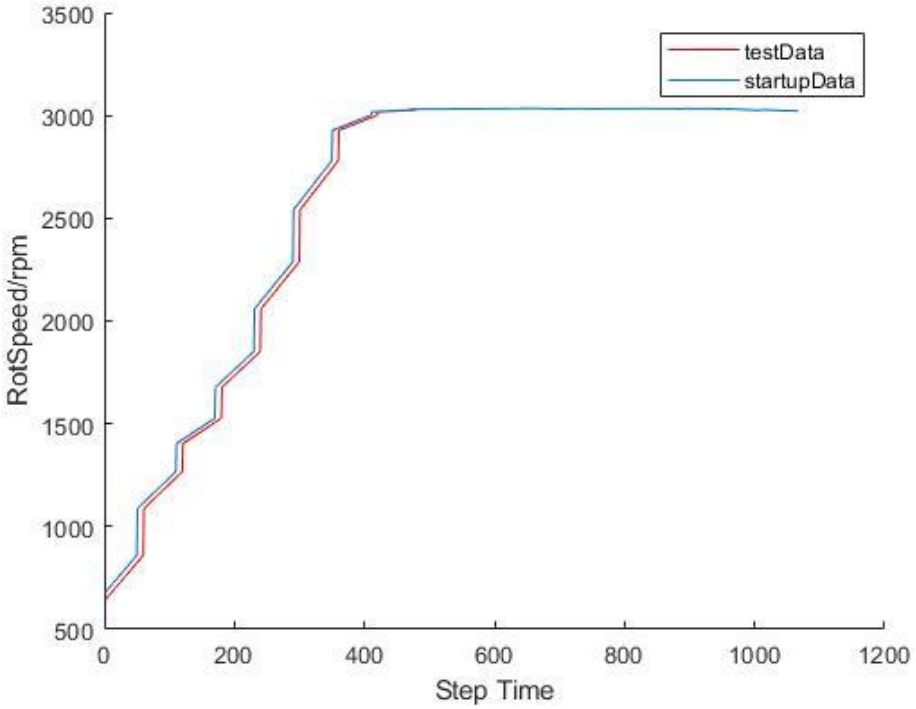



**Figure 4.3:** Variations of rotational speed for different samples/ manouvers.

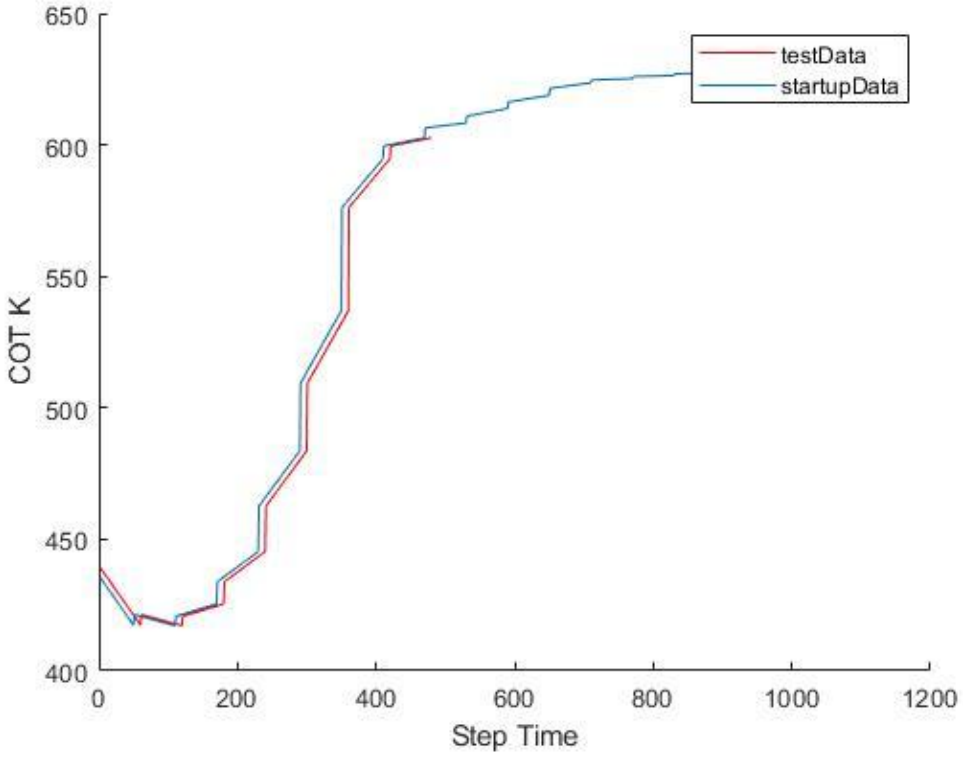



**Figure 4.4:** Variations of COT for different manouvers/samples.

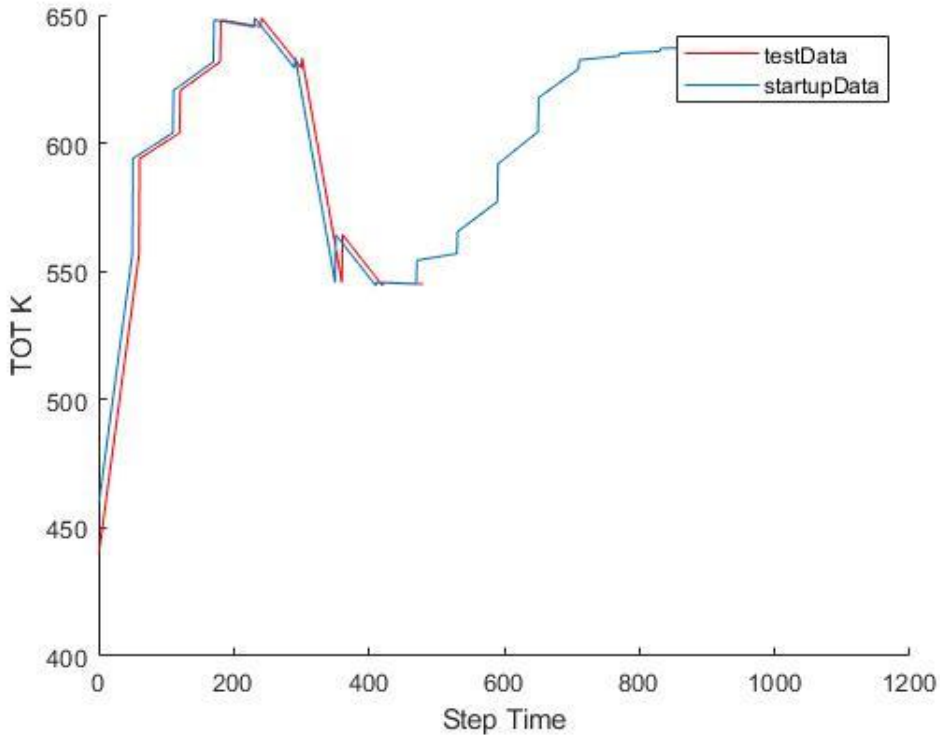



**Figure 4.5:** Variations of TOT for different samples/ manouvers.

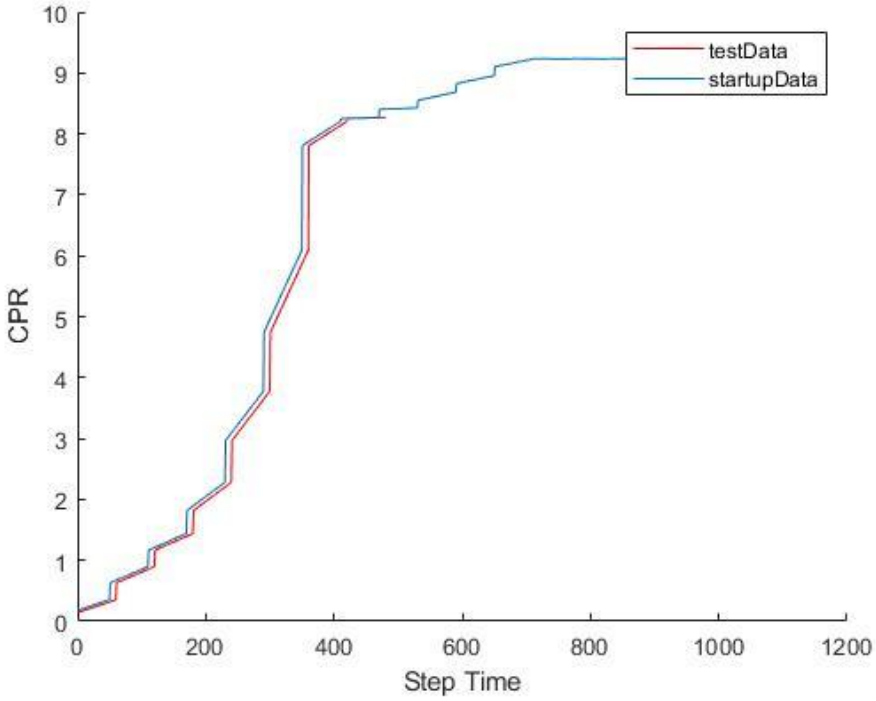



**Figure 4.6:** Variations of CPR for different manouvers.

### 4.1.1 MIMO model result

The MIMO system consists of 4 inputs and 4 outputs and represent the different start-up manouver

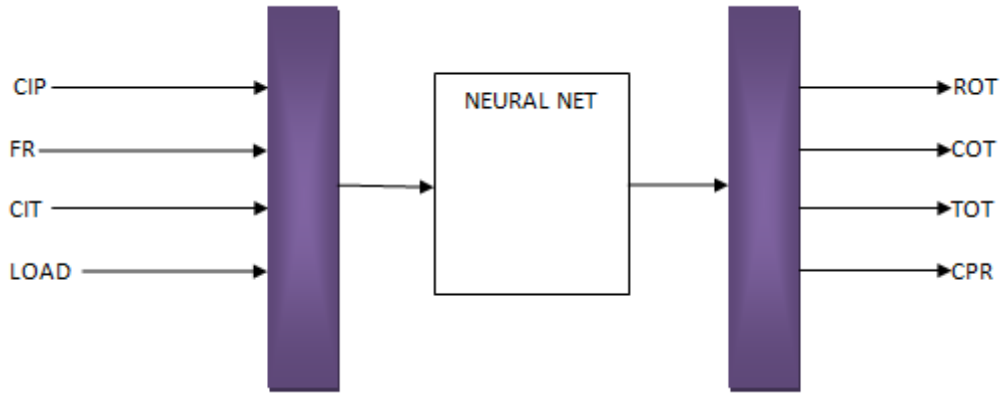



**Figure 4.7:** Block diagram of complete MIMO neural network model.

**Table 4.9:** Top (15) heads of FF_ BPP  MIMO model result

| MIMO Model Comparison for Feed Forward Back Propagation for Data 1 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| S/N | RS | COT K | TOT K | CPR | RS' | COT(K)' | TOT(K)' | CPR |
| 1 | 678.271 | 435.845 | 460.235 | 0.179929 | 683.9169 | 437.1112 | 465.7369 | 0.364692 |
| 2 | 681.973 | 435.467 | 462.201 | 0.183884 | 685.8417 | 436.475 | 466.6243 | 0.375616 |
| 3 | 685.675 | 435.09 | 464.167 | 0.18685 | 687.1968 | 436.1548 | 467.1737 | 0.382854 |
| 4 | 689.378 | 434.712 | 466.133 | 0.190805 | 689.5323 | 435.6941 | 468.0547 | 0.392093 |
| 5 | 693.08 | 434.334 | 468.099 | 0.19476 | 692.6759 | 434.995 | 469.321 | 0.402059 |
| 6 | 696.783 | 433.956 | 470.064 | 0.197726 | 694.8978 | 434.6483 | 470.1128 | 0.409382 |
| 7 | 700.485 | 433.579 | 472.03 | 0.201681 | 698.8153 | 433.9178 | 471.6814 | 0.418148 |
| 8 | 704.187 | 433.201 | 473.996 | 0.204647 | 701.3708 | 433.5558 | 472.6229 | 0.42418 |
| 9 | 707.89 | 432.823 | 475.962 | 0.208602 | 705.2637 | 433.0145 | 474.0916 | 0.430611 |
| 10 | 711.592 | 432.445 | 477.928 | 0.212557 | 709.9689 | 432.2516 | 476.149 | 0.436894 |
| 11 | 715.294 | 432.067 | 479.894 | 0.215523 | 713.1601 | 431.8821 | 477.3909 | 0.442321 |
| 12 | 718.997 | 431.69 | 481.859 | 0.219478 | 717.419 | 431.3206 | 479.2261 | 0.446467 |
| 13 | 722.699 | 431.312 | 483.825 | 0.222444 | 722.3772 | 430.5592 | 481.745 | 0.450186 |
| 14 | 726.402 | 430.934 | 485.791 | 0.226399 | 725.5222 | 430.1922 | 483.1917 | 0.453638 |
| 15 | 730.104 | 430.556 | 487.757 | 0.230354 | 730.4097 | 429.4511 | 485.9553 | 0.455842 |

**Table 4.10:** Tail (15) Values of FF_ BPP MIMO model result

| MIMO Model Comparison for Feed Forward Back Propagation for Data 1 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| S/N | RS | COT K | TOT K | CPR | RS' | COT(K)' | TOT(K)' | CPR |
| 1052 | 3023.041 | 628.346 | 639.544 | 9.20773 | 3030.831 | 626.7852 | 636.2956 | 9.195247 |
| 1053 | 3022.92 | 628.339 | 639.539 | 9.20676 | 3030.825 | 626.787 | 636.3106 | 9.195289 |
| 1054 | 3022.799 | 628.333 | 639.534 | 9.206769 | 3030.822 | 626.7832 | 636.3009 | 9.195274 |
| 1055 | 3022.677 | 628.326 | 639.53 | 9.205789 | 3030.824 | 626.7842 | 636.3073 | 9.195309 |
| 1056 | 3022.556 | 628.32 | 639.525 | 9.205808 | 3030.826 | 626.7817 | 636.3065 | 9.195322 |
| 1057 | 3022.435 | 628.313 | 639.521 | 9.204828 | 3030.833 | 626.7829 | 636.3132 | 9.195365 |
| 1058 | 3022.314 | 628.306 | 639.516 | 9.204846 | 3030.849 | 626.7513 | 636.2009 | 9.195249 |
| 1059 | 3022.193 | 628.3 | 639.512 | 9.204855 | 3030.86 | 626.7526 | 636.2079 | 9.195299 |
| 1060 | 3022.071 | 628.293 | 639.507 | 9.203885 | 3030.883 | 626.7555 | 636.2245 | 9.195392 |
| 1061 | 3021.95 | 628.287 | 639.502 | 9.203894 | 3030.895 | 626.7521 | 636.2155 | 9.195402 |
| 1062 | 3021.829 | 628.28 | 639.498 | 9.202924 | 3030.928 | 626.7553 | 636.2333 | 9.195515 |
| 1063 | 3021.708 | 628.274 | 639.493 | 9.202933 | 3030.946 | 626.752 | 636.2248 | 9.195536 |
| 1064 | 3021.587 | 628.267 | 639.489 | 9.201954 | 3030.969 | 626.7231 | 636.1173 | 9.195454 |
| 1065 | 3021.465 | 628.26 | 639.484 | 9.201972 | 3031.018 | 626.7268 | 636.1366 | 9.195599 |
| 1066 | 3021.344 | 628.254 | #NAME? | 9.201981 | 3031.042 | 626.7236 | 636.1286 | 9.195636 |
| 1067 | 3021.223 | 628.247 | 639.475 | 9.201011 | 3031.104 | 626.7275 | 636.1497 | 9.195809 |

**Table 4.11:** Top (15) heads of NARX MIMO model results

| MIMO Model Comparison for NARX Network for Data 1 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| S/N | RS | COT K | TOT K | CPR | RS' | COT( K)' | TOT(K)' | CPR' |
| 1 | 678.271 | 435.845 | 460.235 | 0.179929 | | | | |
| 2 | 681.973 | 435.467 | 462.201 | 0.183884 | | | | |
| 3 | 685.675 | 435.09 | 464.167 | 0.18685 | 691.458 | 436.7084 | 468.5506 | 0.509953 |
| 4 | 689.378 | 434.712 | 466.133 | 0.190805 | 692.5294 | 435.8549 | 470.0631 | 0.497508 |
| 5 | 693.08 | 434.334 | 468.099 | 0.19476 | 696.5338 | 435.6037 | 471.3708 | 0.493491 |
| 6 | 696.783 | 433.956 | 470.064 | 0.197726 | 699.2625 | 435.0837 | 472.8381 | 0.481519 |
| 7 | 700.485 | 433.579 | 472.03 | 0.201681 | 700.6496 | 434.2642 | 474.4134 | 0.471372 |
| 8 | 704.187 | 433.201 | 473.996 | 0.204647 | 704.7983 | 433.9948 | 475.8218 | 0.464514 |
| 9 | 707.89 | 432.823 | 475.962 | 0.208602 | 706.4455 | 433.1707 | 477.4852 | 0.45474 |
| 10 | 711.592 | 432.445 | 477.928 | 0.212557 | 710.9155 | 432.9507 | 478.8837 | 0.452591 |
| 11 | 715.294 | 432.067 | 479.894 | 0.215523 | 714.3921 | 432.4945 | 480.4466 | 0.443932 |
| 12 | 718.997 | 431.69 | 481.859 | 0.219478 | 716.4302 | 431.7223 | 482.1747 | 0.437018 |
| 13 | 722.699 | 431.312 | 483.825 | 0.222444 | 721.0521 | 431.4808 | 483.6745 | 0.434753 |
| 14 | 726.402 | 430.934 | 485.791 | 0.226399 | 725.396 | 431.1153 | 485.2783 | 0.429757 |
| 15 | 730.104 | 430.556 | 487.757 | 0.230354 | 727.4384 | 430.314 | 487.1426 | 0.422371 |

**Table 4.12:** Tail (5) Values of NARX MIMO model result

| MIMO Model Comparison for NARX Network for Data 1 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| S/N | RS | COT K | TOT K | CPR | RS' | COT( K)' | TOT(K)' | CPR' |
| 1062 | 3021.829 | 628.28 | 639.498 | 9.202924 | 3022.489 | 626.5634 | 639.8031 | 9.168717 |
| 1063 | 3021.708 | 628.274 | 639.493 | 9.202933 | 3023.071 | 626.5222 | 639.8198 | 9.157459 |
| 1064 | 3021.587 | 628.267 | 639.489 | 9.201954 | 3022.658 | 626.5258 | 639.8623 | 9.160968 |
| 1065 | 3021.465 | 628.26 | 639.484 | 9.201972 | 3023.812 | 626.5403 | 639.8304 | 9.164133 |
| 1066 | 3021.344 | 628.254 | #NAME? | 9.201981 | 3023.026 | 626.4322 | 639.9032 | 9.147716 |
| 1067 | 3021.223 | 628.247 | 639.475 | 9.201011 | 3022.696 | 626.4495 | 639.9472 | 9.15185 |

**Table 4.13:** Top (15) heads of LayRecNet MIMO model result

| MIMO Model Comparison for LayRecNet for Data 1 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| S/N | RS | COT K | TOT K | CPR | RS' | COT( K)' | TOT(K)' | CPR' |
| 1 | 678.271 | 435.845 | 460.235 | 0.179929 | 681.9149 | 435.3848 | 463.9469 | 0.253311 |
| 2 | 681.973 | 435.467 | 462.201 | 0.183884 | 684.1152 | 435.2762 | 465.275 | 0.262177 |
| 3 | 685.675 | 435.09 | 464.167 | 0.18685 | 685.5479 | 435.1447 | 466.2224 | 0.267041 |
| 4 | 689.378 | 434.712 | 466.133 | 0.190805 | 688.0576 | 435.1194 | 467.5453 | 0.273586 |
| 5 | 693.08 | 434.334 | 468.099 | 0.19476 | 691.887 | 434.7058 | 469.3411 | 0.281809 |
| 6 | 696.783 | 433.956 | 470.064 | 0.197726 | 694.1992 | 434.4431 | 470.731 | 0.286557 |
| 7 | 700.485 | 433.579 | 472.03 | 0.201681 | 698.9032 | 433.9016 | 472.7465 | 0.294178 |
| 8 | 704.187 | 433.201 | 473.996 | 0.204647 | 701.5843 | 433.5979 | 474.2048 | 0.298469 |
| 9 | 707.89 | 432.823 | 475.962 | 0.208602 | 705.5444 | 433.2962 | 475.9105 | 0.303703 |
| 10 | 711.592 | 432.445 | 477.928 | 0.212557 | 710.887 | 432.6393 | 478.0333 | 0.310322 |
| 11 | 715.294 | 432.067 | 479.894 | 0.215523 | 713.9349 | 432.2982 | 479.8427 | 0.314441 |
| 12 | 718.997 | 431.69 | 481.859 | 0.219478 | 717.9868 | 431.9305 | 481.5866 | 0.318924 |
| 13 | 722.699 | 431.312 | 483.825 | 0.222444 | 723.3507 | 431.2418 | 483.7081 | 0.324727 |
| 14 | 726.402 | 430.934 | 485.791 | 0.226399 | 726.3266 | 430.8991 | 485.4353 | 0.328283 |
| 15 | 730.104 | 430.556 | 487.757 | 0.230354 | 731.5453 | 430.2208 | 487.5246 | 0.333669 |

**Table 4.14:** Tail (7) values of LayRecNet MIMO model result

| MIMO Model Comparison for LayRecNet for Data 1 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| S/N | RS | COT K | TOT K | CPR | RS' | COT( K)' | TOT(K)' | CPR' |
| 1062 | 3021.829 | 628.28 | 639.498 | 9.202924 | 3030.661 | 626.9451 | 637.6246 | 9.211655 |
| 1063 | 3021.708 | 628.274 | 639.493 | 9.202933 | 3030.728 | 626.9317 | 637.6204 | 9.211403 |
| 1064 | 3021.587 | 628.267 | 639.489 | 9.201954 | 3030.773 | 626.8796 | 637.5285 | 9.210873 |
| 1065 | 3021.465 | 628.26 | 639.484 | 9.201972 | 3030.834 | 626.886 | 637.6727 | 9.21086 |
| 1066 | 3021.344 | 628.254 | #NAME? | 9.201981 | 3030.901 | 626.8739 | 637.6845 | 9.210629 |
| 1067 | 3021.223 | 628.247 | 639.475 | 9.201011 | 3030.961 | 626.8864 | 637.8589 | 9.210695 |

**Table 4.15:** details of neural network weights and biases for FF_BPP

| Input to hidden layer weights for FF_BPP | | | | b1 | b2 | Hidden to Output layer Weights | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0.196747 | 3.691256 | -0.53686 | -0.73332 | -2.62725 | -0.20611 | -2.86732 | -2.48394 | 3.645195 | -2.02702 |
| -1.99188 | -3.68401 | 4.925835 | 5.617045 | 5.932284 | -0.7315 | 0.059412 | -0.03382 | -0.30024 | -0.0792 |
| 0.117399 | -3.42427 | -2.69477 | -0.86862 | -1.48002 | 2.199546 | -0.15031 | -0.0119 | 0.764508 | 0.01084 |
| 0.217799 | 4.117375 | 0.0195 | -0.68361 | -2.70018 | 0.100346 | 1.607953 | 0.824047 | -4.62631 | 0.624066 |
| -0.12646 | 3.021087 | 0.328456 | -1.5944 | -0.50432 | | 0.411222 | 0.313463 | -0.28004 | 0.235124 |
| 0.163984 | 1.873126 | -0.72689 | -0.37432 | -1.24077 | | 2.162946 | 2.067854 | -1.45811 | 1.6486 |
| 0.091695 | 2.486372 | -0.93974 | 2.023255 | 1.255417 | | -0.27701 | -0.26308 | -1.30189 | -0.18161 |
| -0.73995 | -14.8527 | -8.46121 | 8.017257 | 0.36996 | | 0.033457 | -0.00765 | -0.17206 | 0.017816 |
| -0.04263 | -1.87267 | -0.01095 | -0.10455 | 1.136873 | | 0.172179 | -0.85424 | -4.44475 | -0.87437 |
| -0.04646 | -1.35269 | -0.85813 | -1.80672 | -3.00882 | | -0.3534 | 1.031938 | -4.1675 | -0.02849 |
| -0.27411 | -6.34287 | -4.32901 | 3.022064 | 1.882777 | | 0.041448 | 0.000516 | -0.35098 | -0.05758 |
| 0.278934 | 0.743966 | 0.930445 | 1.818239 | 3.29709 | | 0.118716 | 1.932284 | -4.10807 | 0.261106 |

**Table 4.16:** Details of neural network weights and biases for LayRecNet

| Hidden Layer Weights | | | | b1 | b2 | Hidden to Output Layer Weights | | | |
|---|---|---|---|---|---|---|---|---|---|
| -0.49162 | -0.81916 | -0.69161 | 0.286656 | 1.569899 | -1.28377 | 1.16929 | 1.125467 | -0.77221 | 0.00467 |
| -0.00489 | -0.84062 | -0.86191 | -0.32152 | 1.333351 | -0.76496 | 0.726728 | 0.242834 | -2.66 | -0.00271 |
| -0.70214 | 1.616273 | 0.788832 | 0.974573 | 1.535235 | 1.68724 | -0.90546 | -1.06755 | 1.73341 | -0.35685 |
| 0.011339 | -1.83846 | 0.012348 | 0.274828 | 0.267743 | -0.09152 | -0.96172 | -1.05522 | 0.94547 | -0.68864 |
| -0.46789 | -2.41224 | 2.080309 | 0.229953 | 2.442631 | | 0.293948 | 0.284558 | -0.57322 | 0.146277 |
| 0.937764 | -0.80766 | -0.40118 | -2.0818 | -2.06158 | | -0.4706 | -0.6117 | 0.930457 | -0.17945 |
| 0.640006 | -0.3344 | -2.17041 | 1.447366 | 2.120337 | | -0.24993 | -0.27242 | 1.212615 | -0.11347 |
| -0.03749 | -0.39253 | 5.300193 | 1.311971 | 1.318496 | | 0.022962 | -0.00058 | -0.32955 | 0.036077 |
| -1.24265 | -0.32496 | 0.952798 | -0.83878 | -0.45499 | | -0.23784 | -0.51003 | 0.093106 | -0.34305 |
| 0.398713 | 0.487283 | -2.63076 | 1.525183 | 1.615231 | | 0.113781 | 0.108987 | -1.52462 | 0.191854 |
| -0.13656 | 2.65516 | 0.35245 | 0.753944 | 2.666987 | | 0.924717 | 0.355418 | 0.821948 | 0.403915 |
| 0.541793 | 1.164145 | 1.258275 | -0.08181 | -1.17497 | | 0.545971 | 0.443265 | -0.64584 | -0.024 |

**Table 4.17:** Details of neural network weights and biases for NARX

| Input to Hidden Layer Weights | | | | | | | | b1 | b2 | Hidden Layer to Output Weights | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.486765 | 2.933132 | 0.851983 | 0.99019 | -0.58721 | 2.189282 | 0.592851 | -0.75298 | 4.559648 | -0.04237 | 0.788755 | 0.909892 | -0.11794 | 0.112807 |
| -3.20316 | -0.52679 | 1.161424 | 2.143368 | 1.713255 | -0.81939 | 3.568773 | -2.91209 | -0.55955 | -0.57579 | -2.71792 | -2.35219 | -0.54044 | -0.99931 |
| 4.36E-06 | 0.677077 | 0.986761 | -0.95302 | -0.02431 | 0.214812 | -0.52212 | 0.299064 | 0.043286 | -0.24518 | 0.131261 | 0.293841 | -0.52432 | -0.45853 |
| -0.10837 | -0.58532 | -0.52525 | 0.988461 | 0.353224 | -1.19386 | -1.48815 | -0.32202 | -1.1151 | -0.80994 | 0.307502 | 0.588734 | -0.05049 | 0.134231 |
| 0.060133 | -0.37213 | -0.86708 | -1.67632 | 0.19054 | -0.66236 | -0.52287 | -0.67506 | -0.04308 | | 1.000568 | 0.594478 | 0.347131 | 0.130521 |
| 0.244978 | -0.8748 | 0.855151 | 1.06818 | 0.013171 | 0.703496 | 0.078957 | -0.95634 | 0.301871 | | -0.29912 | -0.31829 | 0.07092 | -0.43136 |
| 1.076022 | 3.030763 | 2.999009 | 4.044999 | -2.3741 | 1.163872 | 0.584164 | 2.613684 | 1.058451 | | -0.05352 | -0.07192 | 0.046868 | -0.02078 |
| 0.15309 | 0.368849 | 0.308909 | -1.81811 | -0.31281 | 0.335756 | -0.61856 | -0.81419 | 2.391238 | | -0.79199 | -0.21351 | 0.360172 | 0.702821 |
| -0.78204 | 1.498199 | -4.03475 | -1.13471 | 0.879547 | 0.284393 | -3.14921 | -1.67807 | -0.83482 | | -0.2166 | -0.21591 | -0.01172 | -0.17457 |
| -2.8487 | -0.20404 | -0.0978 | 1.603423 | 3.537623 | -0.69924 | 0.638069 | -0.11975 | 5.615362 | | 2.568458 | 2.098065 | 0.396651 | 0.167314 |
| -1.25319 | -0.23684 | -0.77659 | 0.661877 | 0.126785 | -0.05447 | -0.79843 | -0.65367 | 0.090508 | | 0.148235 | 0.154604 | -0.06536 | 0.083401 |
| -0.07548 | -0.85653 | -1.15311 | -0.70607 | 0.380286 | -0.19276 | -0.81378 | -0.59463 | 4.692159 | | -0.16687 | -0.1871 | -0.12462 | -0.10392 |

**Table 4.18:** Top (10) heads of the simulation of LayRecNet MIMO model with test data

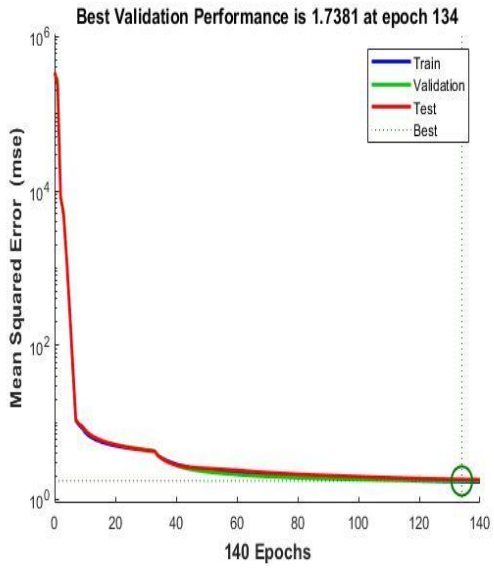
| MIMO Simulation data | | | | | | | |
|---|---|---|---|---|---|---|---|
| REAL | | | | SIMULATED | | | |
| RS | COT K | TOT K | CPR | RS | COT K | TOT K | CPR |
| 641.247 | 439.623 | 440.577 | 0.144336 | 644.6489 | 439.0187 | 439.5133 | 0.152972 |
| 644.949 | 439.245 | 442.543 | 0.148291 | 648.7399 | 438.6524 | 441.8011 | 0.157642 |
| 648.652 | 438.868 | 444.509 | 0.152246 | 651.7988 | 438.4143 | 443.4036 | 0.161465 |
| 652.354 | 438.49 | 446.475 | 0.155212 | 655.8715 | 438.0401 | 445.6963 | 0.166072 |
| 656.056 | 438.112 | 448.441 | 0.159166 | 658.9452 | 437.7974 | 447.3103 | 0.169898 |
| 659.759 | 437.734 | 450.406 | 0.162133 | 662.0104 | 437.3251 | 449.4024 | 0.172641 |
| 663.461 | 437.356 | 452.372 | 0.166087 | 666.1503 | 436.945 | 451.7258 | 0.177324 |
| 667.163 | 436.979 | 454.338 | 0.169053 | 669.2464 | 436.6974 | 453.3528 | 0.181165 |
| 670.866 | 436.601 | 456.304 | 0.173008 | 673.3665 | 436.3107 | 455.6767 | 0.185788 |
| 674.568 | 436.223 | 458.27 | 0.176963 | 676.1645 | 435.8553 | 457.6371 | 0.188139 |
| 678.271 | 435.845 | 460.235 | 0.179929 | 679.5873 | 435.5832 | 459.4172 | 0.192435 |

**Table 4.19:** Data 1 MIMO model result showing neurons and performance

| Network | Neurons | Performance (MSE) | Performance (MAPE) | Rank 1(MSE) | Rank 2(MAPE) | Rank 3 (r1+r2)/2 |
|---|---|---|---|---|---|---|
| LayRecnet | 12 | 1.12 | 0.7310 | 3 | 3 | 3 |
| FF BPP | 12 | 1.74 | 1.4249 | 2 | 2 | 2 |
| NARX | 12 | 19.8 | 1.7358 | 1 | 1 | 1 |

**Table 4.20:** MIMO simulation performance of LayRecNet model

| Neurons | MSE | MAPE | RMSE |
|---|---|---|---|
| 12 | 0.6065 | 1.5925 | 0.778 |



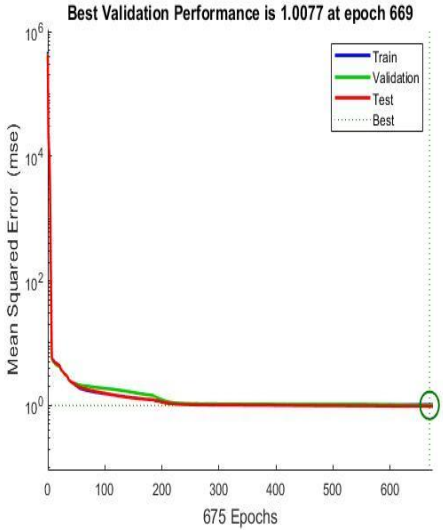




**Figure 4.8:** FF BPP performance per epoch. **Figure 4.9:** MIMO LayRecNet MSE per epoch.

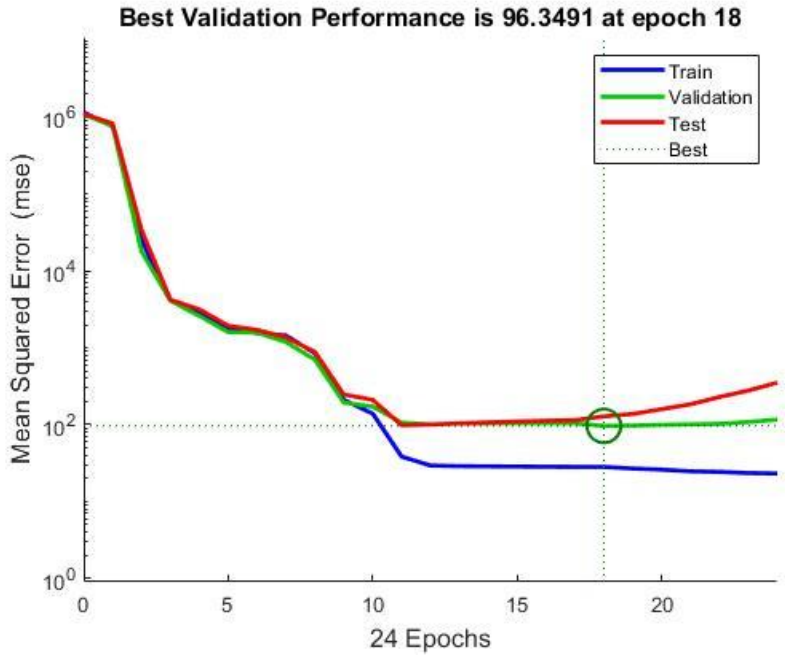



**Figure 4.10:** MIMO NARX performance per iteration/epoch for sample 1 data.

### 4.1.2 MISO model result

Data 1 having a total of 1067 samples is used to model the outputs separately because it gives a wide coverage of the whole operational range of the start-up/ manouver.

**Table 4.21:** Top (7) heads of MISO model result for compressor outlet temperature

| | COT(K) comparison for Targets and Outputs | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | FF BPP | | | NARX | | | LayRecNet | |
| S/N | Targets | Outputs | | Targets | Outputs | | Targets | Outputs |
| 435.845 | 435.63 | | | | | | 435.845 | 436.264 |
| 435.467 | 435.193 | | | | | | 435.467 | 435.6394 |
| 435.09 | 434.9855 | | 435.845 | 435.0751 | | | 435.09 | 435.3101 |
| 434.712 | 434.6312 | | 435.467 | 434.7307 | | | 434.712 | 434.8263 |
| 434.334 | 434.1737 | | 435.09 | 434.3452 | | | 434.334 | 434.2478 |
| 433.956 | 433.9638 | | 434.712 | 433.9594 | | | 433.956 | 433.9456 |
| 433.579 | 433.4958 | | 434.334 | 433.5868 | | | 433.579 | 433.3982 |
| 433.201 | 433.2783 | | 433.956 | 433.1954 | | | 433.201 | 433.121 |
| | | | 433.579 | 432.8397 | | | | |
| | | | 433.201 | 432.4502 | | | | |

**Table 4.22:** Tail (7) values of MISO model results for COT

| | COT(K) comparison for Targets and Outputs | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | FF BPP | | | NARX | | | LayRecNet | |
| S/N | Targets | Outputs | | Targets | Outputs | | Targets | Outputs |
| 628.293 | 627.2263 | | 628.293 | 628.5813 | | | 628.293 | 627.7101 |
| 628.287 | 627.2663 | | 628.287 | 628.5539 | | | 628.287 | 627.6554 |
| 628.28 | 627.2957 | | 628.28 | 628.5242 | | | 628.28 | 627.7256 |
| 628.274 | 627.3408 | | 628.274 | 628.4921 | | | 628.274 | 627.6703 |
| 628.267 | 627.1863 | | 628.267 | 628.4571 | | | 628.267 | 627.4938 |
| 628.26 | 627.2228 | | 628.26 | 628.4383 | | | 628.26 | 627.562 |
| 628.254 | 627.2746 | | 628.254 | 628.3972 | | | 628.254 | 627.4913 |
| 628.247 | 627.3243 | | 628.247 | 628.3557 | | | 628.247 | 627.5549 |

**Table 4.23:** Top (7) heads of MISO model result for compressor pressure ratio

| | CPR Targets and Outputs Comparison | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | FF_BPP | | | NARX | | | LayRecNet | |
| S/N | Targets | Outputs | | Targets | Outputs | | Targets | Outputs |
| 1 | 0.1799 | 0.1736 | | | | | 0.179929 | 0.185539 |
| 2 | 0.1839 | 0.1787 | | | | | 0.183884 | 0.187586 |
| 3 | 0.1868 | 0.1823 | | 0.1799 | 0.213002 | | 0.18685 | 0.18914 |
| 4 | 0.1908 | 0.1871 | | 0.1839 | 0.212408 | | 0.190805 | 0.191598 |
| 5 | 0.1948 | 0.1919 | | 0.1868 | 0.210957 | | 0.19476 | 0.194675 |
| 6 | 0.1977 | 0.1959 | | 0.1908 | 0.210403 | | 0.197726 | 0.197079 |
| 7 | 0.2017 | 0.2005 | | 0.1948 | 0.210598 | | 0.201681 | 0.200733 |
| 8 | 0.2046 | 0.204 | | 0.1977 | 0.210621 | | 0.204647 | 0.203314 |
| 9 | 0.2086 | 0.2083 | | 0.2017 | 0.211516 | | | |
| 10 | 0.2126 | 0.2126 | | 0.2046 | 0.210891 | | | |

**Table 4.24:** Tail (7) values of MISO model results for CPR

| | CPR Targets and Outputs Comparison | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | FF_BPP | | | NARX | | | LayRecNet | |
| S/N | Targets | Outputs | | Targets | Outputs | | Targets | Outputs |
| 1062 | 9.2039 | 9.2231 | | 9.2039 | 9.20275 | | 9.203885 | 9.222947 |
| 1063 | 9.2039 | 9.2231 | | 9.2039 | 9.202152 | | 9.203894 | 9.222805 |
| 1064 | 9.2029 | 9.2233 | | 9.2029 | 9.201513 | | 9.202924 | 9.223258 |
| 1065 | 9.2029 | 9.2233 | | 9.2029 | 9.201007 | | 9.202933 | 9.223121 |
| 1066 | 9.202 | 9.2229 | | 9.202 | 9.200227 | | 9.201954 | 9.223191 |
| 1067 | 9.202 | 9.2232 | | 9.202 | 9.198844 | | 9.201972 | 9.223595 |
| 1068 | 9.202 | 9.2232 | | 9.202 | 9.198987 | | 9.201981 | 9.223445 |
| 1069 | 9.201 | 9.2235 | | 9.201 | 9.198338 | | 9.201011 | 9.223807 |

**Table 4.25:** Top (7) heads of MISO model result for turbine outlet temperature

| | TOT (K) Targets and Outputs Comparison | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | FF_BPP | | | NARX | | | LayRecNet | |
| S/N | Targets | Outputs | | Targets | Outputs | | Targets | Outputs |
| 1 | 460.235 | 462.9217 | | | | | 460.235 | 462.462 |
| 2 | 462.201 | 464.5648 | | | | | 462.201 | 463.6734 |
| 3 | 464.167 | 465.9568 | | 460.235 | 466.158 | | 464.167 | 464.4159 |
| 4 | 466.133 | 466.9565 | | 462.201 | 467.8349 | | 466.133 | 466.0521 |
| 5 | 468.099 | 468.7537 | | 464.167 | 469.568 | | 468.099 | 468.2196 |
| 6 | 470.064 | 470.4707 | | 466.133 | 471.2891 | | 470.064 | 469.386 |
| 7 | 472.03 | 472.3911 | | 468.099 | 472.9871 | | 472.03 | 471.8781 |
| 8 | 473.996 | 474.0178 | | 470.064 | 474.7299 | | 473.996 | 473.1559 |
| 9 | | | | 472.03 | 476.4683 | | | |
| 10 | | | | 473.996 | 478.2612 | | | |

**Table 4.26:** Tail (7) values of MISO model results for TOT

| | TOT (K) Targets and Outputs Comparison | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | FF_BPP | | | NARX | | | LayRecNet | |
| S/N | Targets | Outputs | | Targets | Outputs | | Targets | Outputs |
| 1062 | 639.507 | 637.816 | | 639.507 | 639.6336 | | 639.507 | 638.2117 |
| 1063 | 639.502 | 637.7497 | | 639.502 | 639.6379 | | 639.502 | 638.1064 |
| 1064 | 639.498 | 637.9801 | | 639.498 | 639.6458 | | 639.498 | 638.2416 |
| 1065 | 639.493 | 637.9332 | | 639.493 | 639.6616 | | 639.493 | 638.1369 |
| 1066 | 639.489 | 637.4989 | | 639.489 | 639.6708 | | 639.489 | 637.6738 |
| 1067 | 639.484 | 637.778 | | 639.484 | 639.7457 | | 639.484 | 637.8143 |
| 1068 | 639.48 | 637.7581 | | 639.48 | 639.6972 | | 639.48 | 637.6941 |
| 1069 | 639.475 | 638.0994 | | 639.475 | 639.7052 | | 639.475 | 637.8478 |

**Table 4.27:** Top (7) heads of MISO model result for rotational speed

| | ROT (K) Targets and Outputs Comparison | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | FF_BPP | | | NARX | | | LayRecNet | |
| S/N | Targets | Outputs | | Targets | Outputs | | Targets | Outputs |
| 1 | 678.271 | 679.7783 | | | | | 678.271 | 681.5042 |
| 2 | 681.973 | 683.2319 | | | | | 681.973 | 683.6139 |
| 3 | 685.675 | 685.9899 | | 678.271 | 705.2541 | | 685.675 | 685.3218 |
| 4 | 689.378 | 689.4363 | | 681.973 | 713.4893 | | 689.378 | 687.9567 |
| 5 | 693.08 | 693.2211 | | 685.675 | 708.7306 | | 693.08 | 691.8892 |
| 6 | 696.783 | 696.4491 | | 689.378 | 708.6811 | | 696.783 | 694.8759 |
| 7 | 700.485 | 700.413 | | 693.08 | 717.4921 | | 700.485 | 699.6199 |
| 8 | 704.187 | 703.4076 | | 696.783 | 711.7212 | | 704.187 | 702.7943 |
| 9 | | | | 700.485 | 720.8429 | | | |
| 10 | | | | 704.187 | 715.8395 | | | |

**Table 4.28:** Tail (7) values of MISO model results for ROT

| | ROT (K) Targets and Outputs Comparison | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | FF_BPP | | | NARX | | | LayRecNet | |
| S/N | Targets | Outputs | | Targets | Outputs | | Targets | Outputs |
| 1062 | 3.02E+03 | 3.03E+03 | | 3.02E+03 | 3.03E+03 | | 3022.071 | 3028.961 |
| 1063 | 3.02E+03 | 3.03E+03 | | 3.02E+03 | 3.03E+03 | | 3021.95 | 3028.908 |
| 1064 | 3.02E+03 | 3.03E+03 | | 3.02E+03 | 3.03E+03 | | 3021.829 | 3029.032 |
| 1065 | 3.02E+03 | 3.03E+03 | | 3.02E+03 | 3.03E+03 | | 3021.708 | 3028.968 |
| 1066 | 3.02E+03 | 3.03E+03 | | 3.02E+03 | 3.03E+03 | | 3021.587 | 3029.32 |
| 1067 | 3.02E+03 | 3.03E+03 | | 3.02E+03 | 3.03E+03 | | 3021.465 | 3029.379 |
| 1068 | 3.02E+03 | 3.03E+03 | | 3.02E+03 | 3.03E+03 | | 3021.344 | 3029.31 |
| 1069 | 3.02E+03 | 3.03E+03 | | 3.02E+03 | 3.03E+03 | | 3021.223 | 3029.296 |

**Table 4.29:** Details of neural network weights and biases for FF_BPP

| hidden layer weights for FF_BPP for COT | | | | b2 | b1 | Out weigt |
|---|---|---|---|---|---|---|
| -1.37045 | -1.52647 | -1.72813 | 2.544906 | 3.506784 | 0.359547 | -0.90454 |
| 0.372387 | 4.635095 | 1.233162 | 0.61742 | 1.543948 | | 0.334973 |
| 1.872532 | 1.275235 | 2.908633 | -0.82946 | -2.47757 | | -0.50173 |
| 0.334084 | -0.31049 | -0.15156 | -0.65881 | 0.104419 | | 1.544945 |
| 1.722503 | -1.10574 | 2.110073 | -3.96992 | -0.78783 | | -0.02955 |
| 0.007117 | -0.54177 | 1.048119 | -1.41254 | 0.896881 | | -0.95998 |
| -1.80437 | -0.03123 | -3.14295 | 1.244662 | 2.156762 | | 0.200338 |
| 0.027444 | -1.93509 | 2.467805 | -1.07377 | -0.89199 | | -0.21727 |
| -1.34247 | 1.836653 | -1.59014 | -2.50852 | -2.28076 | | 0.189592 |
| 0.269556 | 0.050995 | -2.90708 | -2.62041 | -0.11555 | | -0.31404 |
| 1.460219 | -1.925 | 1.494512 | -0.18467 | 2.210713 | | -0.22404 |
| -1.37173 | 0.924891 | -1.27411 | -1.2031 | -3.08277 | | -0.15894 |
| **hidden layer weights for FF_BPP for CPR** | | | | **b1** | **b2** | **Out weigt** |
| 0.641083 | 0.372116 | 0.582326 | -2.43337 | -3.65747 | 0.439968 | 0.08359 |
| 0.25931 | -2.52045 | 1.5506 | 0.404106 | 2.803518 | | 0.358809 |
| 0.157171 | 0.631333 | -0.66758 | -0.35266 | 0.534952 | | -2.34229 |
| 1.22286 | 2.653802 | -0.98057 | 0.329727 | -0.16584 | | 0.368621 |
| -0.10978 | 1.10882 | -0.39355 | -0.9828 | -0.06605 | | 1.928164 |
| 0.145679 | -1.16626 | -0.15595 | 1.443561 | -0.02223 | | 1.099767 |
| -0.1239 | 2.284392 | 0.091121 | 1.691461 | -0.37577 | | 0.027337 |
| -1.11942 | -1.72649 | 1.762468 | 0.799213 | 0.1101 | | 0.583423 |
| 0.438426 | 0.54074 | -0.42149 | -0.62492 | 0.771748 | | 0.962225 |
| -0.49127 | -1.51065 | 1.079427 | -0.87648 | -1.21455 | | -1.25116 |
| 0.871125 | -1.04757 | 1.916657 | 0.490496 | 2.580167 | | -0.16731 |
| 0.052856 | 3.131463 | 2.762845 | 2.905311 | -3.21912 | | 0.09532 |
| **hidden layer weights for FF_BPP for TOT** | | | | **b1** | **b2** | **Out weigt** |
| -0.25518 | -0.68427 | -0.31294 | 2.216338 | -1.24389 | 3.083704 | 5.581487 |
| 2.851233 | -3.63316 | -5.26874 | 7.973024 | -8.8142 | | -0.98255 |
| 2.560425 | 0.582795 | -6.71393 | -3.24961 | -2.9043 | | 0.821179 |
| -0.59643 | -4.12744 | 5.126182 | 9.120133 | -1.22691 | | 1.948919 |
| -0.95491 | 1.633573 | 1.158518 | 2.633816 | -1.35629 | | -4.81857 |
| -0.31145 | -1.01125 | 2.391204 | -4.34013 | -3.52087 | | -2.87005 |
| 0.351798 | 6.865899 | -2.16727 | -8.50354 | 0.046478 | | 1.679683 |
| 0.411397 | -9.62093 | -0.19688 | -0.77748 | 0.943939 | | -0.15676 |
| 0.731755 | 0.284199 | -2.2386 | 3.780202 | 3.228637 | | -6.03883 |
| 1.241966 | -5.16241 | -3.16558 | -2.69355 | 2.335865 | | -2.35594 |
| -1.57851 | 0.793044 | 2.26591 | 0.295501 | 0.414139 | | -1.68659 |
| -3.08738 | 5.514339 | -4.56542 | 0.117496 | -5.91317 | | 0.153926 |

**Table 4.30:** Details of neural network weights and biases for FF_BPP

| hidden weights for FF_BPP for ROT | | | | b1 | b2 | Out weigt |
|---|---|---|---|---|---|---|
| 0.046549 | -2.87521 | -1.09784 | -0.02631 | 2.442514 | 1.539038 | 2.385959 |
| -0.14411 | 0.354057 | -2.63533 | 3.690821 | 5.52724 | | 1.519256 |
| 0.488486 | -2.41009 | 0.402553 | 2.812839 | -0.69729 | | -0.62328 |
| -0.00771 | -3.70214 | -0.23506 | 0.234895 | 3.54097 | | -1.51532 |
| -0.17153 | 0.295362 | -2.05671 | -1.65563 | 0.280957 | | -1.15228 |
| -1.02012 | -0.99463 | 0.899536 | -2.3078 | -1.82893 | | -0.29796 |
| 2.988736 | 2.620302 | -0.66224 | -0.96962 | 0.59631 | | -0.12143 |
| -2.80862 | -0.3679 | -3.12232 | -2.47094 | -0.20833 | | -0.00876 |
| -0.01221 | 0.167506 | 1.029314 | -4.19922 | -5.13824 | | 3.63417 |
| -0.86092 | 2.68325 | -3.27288 | -0.09279 | -2.43868 | | 0.020112 |
| 0.069308 | 1.669972 | -1.19525 | 5.143466 | 5.243921 | | 1.073483 |
| -1.72255 | 1.487541 | -4.36451 | 0.084359 | -4.8092 | | 0.017189 |

**Table 4.31:** MISO model result for FF BPP showing neurons and performance

| Variables | Neurons | Performance (MSE) | Performance (MAPE) |
|---|---|---|---|
| Rotational Speed | 12 | 2.55 | 0.0357 |
| Compressor Outlet Temp | 12 | 0.158 | 0.0448 |
| Turbine Outlet Temperature | 12 | 0.296 | 0.495 |
| Compressor Pressure Ratio | 12 | 2.03e-5 | 0.0671 |

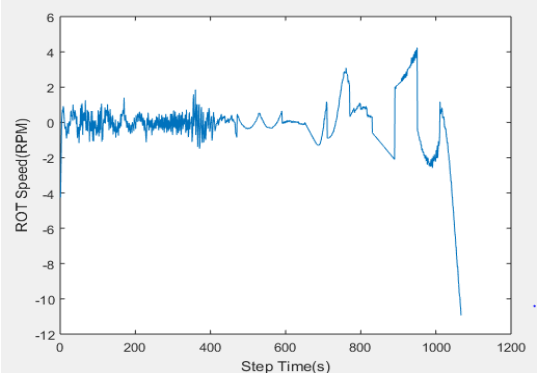



**Figure 4.11:** Error plot of ROT speed.

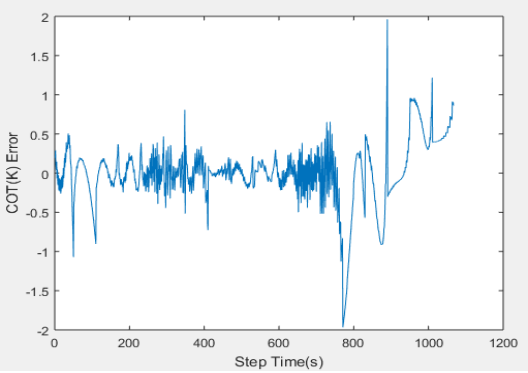



**Figure 4.12:** Error plot of COT (K).

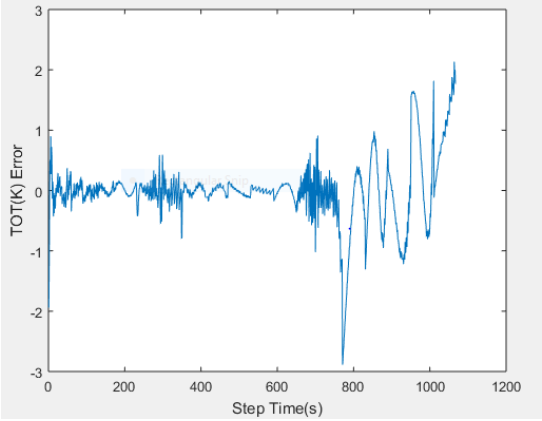



**Figure 4.13:** Error plot of TOT (K).

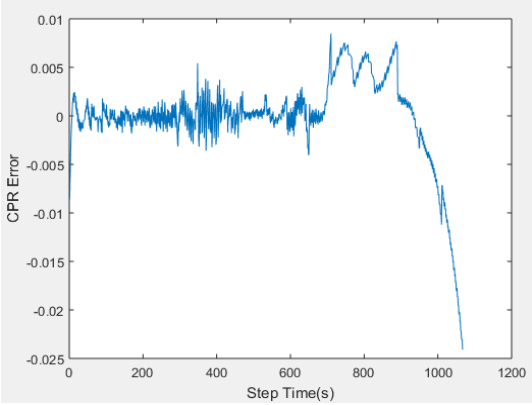
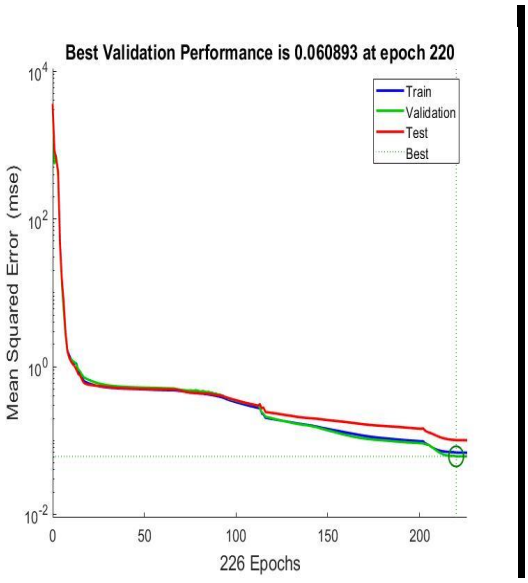



**Figure 4.14:** Error plot of CPR.

**MISO model result for Layer Recurrent Network**



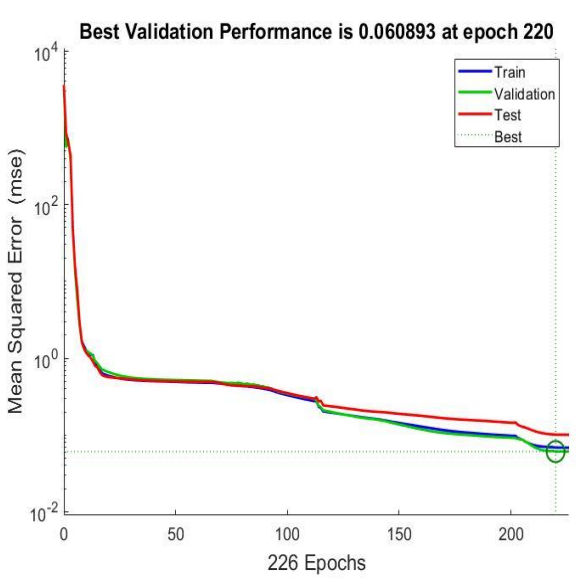
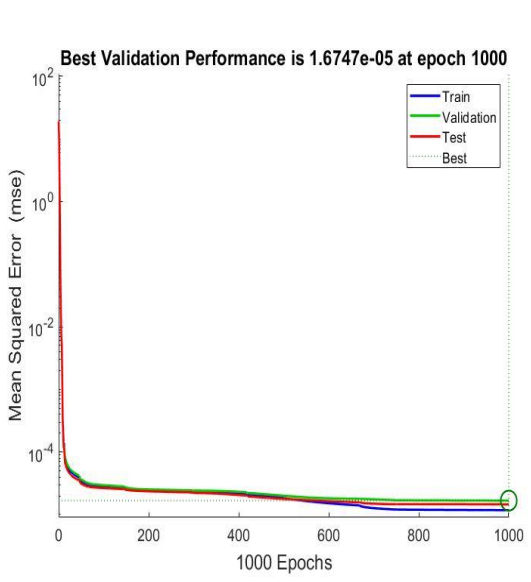




**Figure 4.15:** MISO COT MSE plot per epoch. **Figure 4.16:** MISO CPR MSE plot per epoch.



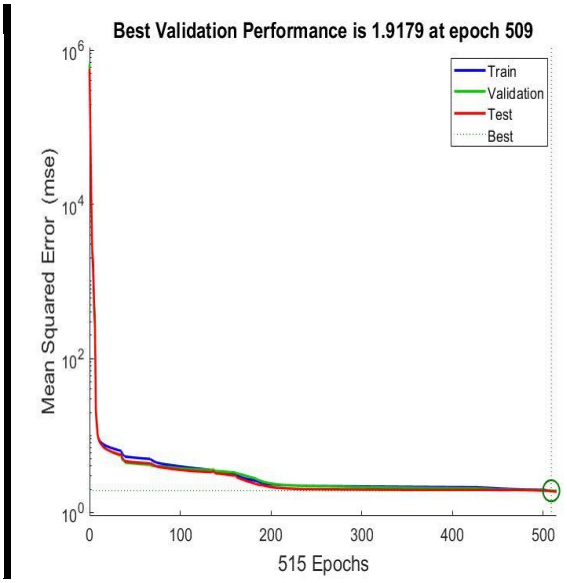




**Figure 4.17:** MISO TOT MSE plot per epoch. **Figure 4.18:** ROT MSE plot per epoch.

**Table 4.32:** Details of neural network weights and biases for Layrecnet

| hidden weights for LayRecNet for COT | | | | b1 | b2 | out weigt |
|---|---|---|---|---|---|---|
| -0.00343 | 1.908291 | -1.4722 | 0.358046 | -3.9615 | 1.618028 | -3.09168 |
| 0.051346 | -1.44624 | 0.995758 | -0.14834 | -1.30757 | | 0.727243 |
| 0.190494 | -1.90663 | 0.099082 | -1.68793 | 2.205439 | | 0.832248 |
| -0.4994 | 0.485101 | -1.23457 | -0.02704 | 0.692633 | | 0.34441 |
| -0.84002 | -0.45441 | -0.60864 | 1.93508 | -1.3207 | | 0.708436 |
| 0.82977 | -0.70406 | -1.761 | -2.14287 | 1.292897 | | 0.91816 |
| -1.04002 | -0.17985 | 0.982691 | -2.41551 | -1.49791 | | -1.15698 |
| 0.284634 | -0.19686 | -1.12283 | 1.139912 | -1.81132 | | 2.532715 |
| 0.407168 | -1.18218 | -0.5227 | 1.710075 | 1.635758 | | -1.58314 |
| -1.12738 | 0.053004 | 0.386045 | -0.78186 | -1.22941 | | 0.611864 |
| 0.694334 | -3.91886 | -1.74551 | 0.118677 | 3.063655 | | -2.89501 |
| -0.87178 | 0.183284 | 0.764391 | -1.62259 | -2.27121 | | -1.3603 |
| **hidden weights for LayRecNet for TOT** | | | | **b1** | **b2** | **Out weigt** |
| -0.20721 | 2.924356 | 0.725954 | 0.652311 | 3.492152 | -0.77607 | 1.714622 |
| -0.2027 | 0.478011 | 0.30339 | -0.19219 | 1.153044 | | -0.79464 |
| -0.54939 | -0.76798 | 3.600734 | 4.806152 | -2.62119 | | 2.275484 |
| 0.57105 | 3.36134 | 2.669077 | -2.23219 | -0.79139 | | 2.066671 |
| 0.5869 | 2.640004 | 2.359706 | -1.5071 | 0.120068 | | -3.31147 |
| -0.30179 | 2.360471 | 3.143314 | -1.88551 | 1.528267 | | 2.780388 |
| 0.513643 | 2.875739 | -0.46205 | -1.71704 | -2.95199 | | -4.60443 |
| 1.379083 | 2.565461 | -0.29558 | 0.27491 | -0.23163 | | 1.948936 |
| -0.24899 | -3.35044 | 4.372173 | 0.553518 | 5.166049 | | -1.82548 |
| 0.354172 | 1.688012 | -0.49197 | 1.437523 | 0.556948 | | -1.52821 |
| 0.596299 | 4.300414 | 4.598053 | -2.22663 | 1.057947 | | 1.338264 |
| -1.08206 | 2.238774 | 2.101731 | -1.18684 | 1.368014 | | -2.13408 |

**Table 4.33:** Details of neural network weights and biases for LayRecNet

| hidden weights for LayRecNet for CPR | | | | b1 | b2 | Out weigt |
|---|---|---|---|---|---|---|
| -2.34437 | -0.53706 | -0.73381 | -1.85139 | 4.036934 | 0.094789 | -1.95623 |
| -0.45058 | 0.055151 | -0.46081 | -0.08421 | -1.76258 | | 0.726959 |
| 0.948887 | 0.154307 | -1.91974 | 0.799581 | -0.98148 | | 0.098372 |
| -0.00475 | -0.63496 | -0.19095 | 1.281767 | 0.445729 | | 1.256742 |
| -1.91479 | 4.583266 | 0.963542 | -3.2395 | 1.804767 | | 3.391936 |
| 0.757401 | 0.783639 | -1.24502 | 1.154687 | -0.61912 | | -0.82052 |
| 1.208797 | 1.936678 | -1.08466 | 1.254975 | 0.253512 | | 0.731864 |
| 0.504411 | -0.48878 | -0.82533 | 0.0807 | -1.02892 | | 1.451263 |
| -0.40057 | 0.908482 | 0.698699 | -0.39734 | -0.33221 | | 1.024732 |
| -0.26646 | -1.21618 | 0.409489 | -0.34293 | -1.1717 | | -1.04885 |
| 0.849597 | -1.7744 | -1.09266 | -3.60505 | 3.656939 | | -0.91661 |
| 0.29016 | 1.907127 | 0.938934 | -0.64735 | 1.532561 | | 1.173155 |
| **hidden weights for LayRecNet for ROT** | | | | **b1** | **b2** | **Out weigt** |
| -5.10381 | 2.563452 | -0.01337 | -0.29085 | 2.655763 | 8.685486 | -0.54971 |
| 2.218089 | 25.81723 | -10.9646 | -14.6222 | -21.5719 | | 0.229421 |
| -0.64225 | -5.17466 | 2.826066 | -2.66269 | 0.668116 | | -2.02413 |
| 4.86293 | -2.13813 | -8.56383 | -3.09675 | 14.91798 | | 7.866276 |
| -4.20303 | 4.367637 | 2.691767 | -1.26925 | 3.198628 | | 1.025661 |
| -16.3146 | 3.813542 | -0.12536 | 27.79055 | -13.8164 | | -0.01051 |
| -12.3279 | 13.02865 | 27.05068 | 5.81529 | 25.55502 | | -0.06119 |
| 0.038587 | -1.64507 | 0.029578 | -0.52323 | -0.67305 | | -10.6943 |
| 47.75799 | 29.61516 | 82.89103 | -38.6149 | 9.41413 | | 0.001295 |
| 0.021181 | 2.289975 | 0.990496 | -3.92081 | -1.89852 | | 3.984538 |
| 34.48971 | -40.2364 | 41.7184 | 38.17518 | -2.65777 | | -0.03303 |
| -4.31663 | -30.1456 | -2.15113 | -52.9773 | -18.144 | | -0.84048 |

**Table 4.34:** MISO model result for layer recurrent network

| Variables | Neurons | Performance(MSE) | Performance (MAPE) |
|---|---|---|---|
| Rotational Speed | 12 | 1.64 | 0.0286 |
| Compressor Outlet Temp | 12 | 0.0684 | 0.0286 |
| Turbine Outlet Temperature | 12 | 0.3371 | 0.0591 |
| Compressor Pressure Ratio | 12 | 1.18e-5 | 0.0503 |

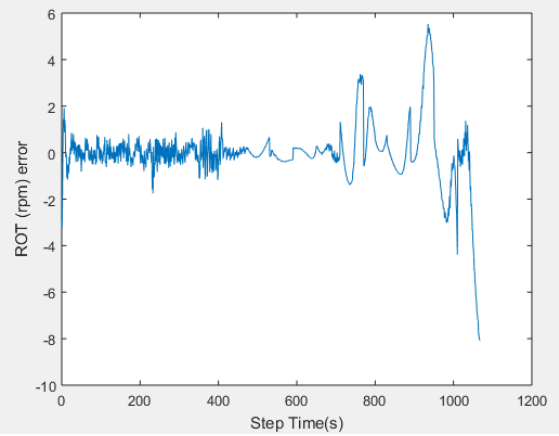



**Figure 4.19:** Error plot of ROT (rpm).

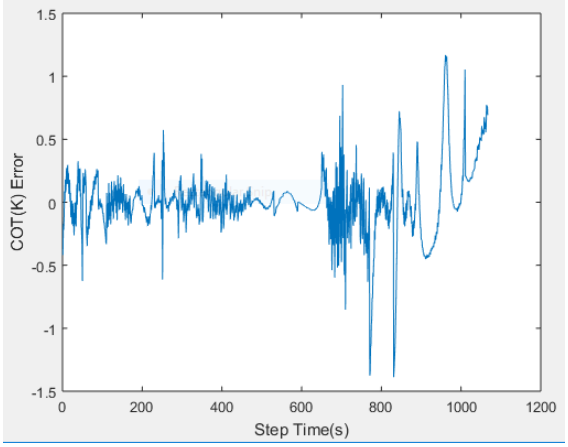



**Figure 4.20:** Error plot of COT (K).

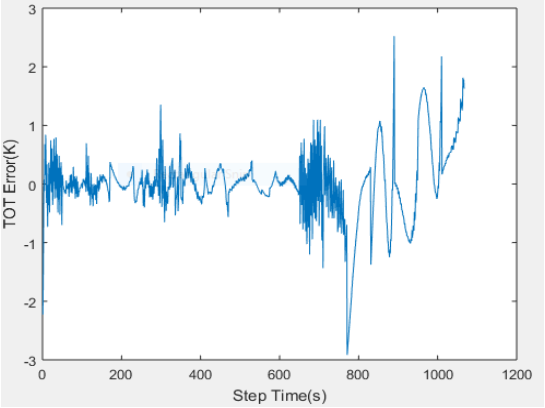



**Figure 4.21:** Error plot of TOT (K).

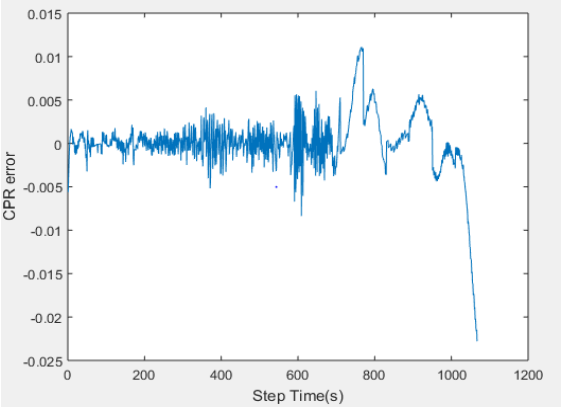



**Figure 4.22:** Error plot of CPR.

**Table 4.35:** Details of neural network weights and biases for NARX

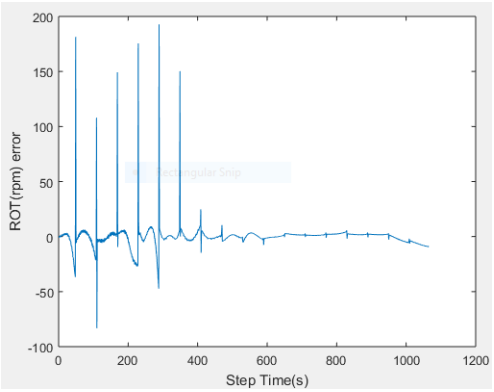
| Input to hidden layer weights for NARX for COT | | | | | | | | b1 | b2 | Out weigt |
|---|---|---|---|---|---|---|---|---|---|---|
| -0.4982 | 0.73946 | 0.924711 | 0.442791 | 0.318943 | 0.086603 | 0.972433 | -0.40678 | 2.446304 | -0.57997 | -0.15717 |
| 0.777078 | 1.035263 | 0.861387 | -0.54447 | -0.316 | -0.67403 | -0.76527 | 0.431999 | -1.66464 | | -1.05147 |
| -0.30753 | 0.136092 | 0.93553 | -0.80818 | -0.66904 | -0.94543 | -0.37224 | -0.13681 | 1.033616 | | 0.088548 |
| 0.049794 | 0.408561 | 0.441646 | 0.085163 | 0.384108 | 1.396868 | 0.830676 | -0.91233 | -0.86133 | | 0.086073 |
| -0.22067 | 0.41689 | 0.740909 | -0.58354 | -1.11963 | 0.246638 | -0.35532 | 0.262824 | -1.27161 | | 0.340617 |
| -0.18499 | -0.15228 | 0.323295 | -0.29819 | -0.12568 | 0.820045 | 0.054749 | 0.058704 | 0.62886 | | -0.47117 |
| 0.407261 | 0.701912 | -0.43261 | -0.56021 | -0.51097 | -0.26072 | 0.368014 | 0.16571 | 0.33514 | | -0.30624 |
| 1.937006 | -0.04188 | 0.023342 | -1.209 | 0.154597 | -0.34583 | -0.20941 | 0.049768 | 0.71092 | | -0.05141 |
| 0.876277 | -0.18581 | 0.458507 | 0.054918 | -0.28426 | 0.318239 | -0.66898 | -0.27045 | 0.400365 | | 0.664188 |
| 0.377356 | -0.65948 | 0.190885 | 0.800095 | 0.914688 | -0.48207 | -0.76217 | -0.3139 | -0.78917 | | -0.14465 |
| -0.63106 | -0.89855 | -0.45292 | -0.89034 | -0.30084 | -0.26773 | -0.55786 | -0.13749 | -1.57163 | | -0.05032 |
| 1.005751 | -1.19639 | -0.65598 | 1.44371 | 0.188859 | -0.74279 | -0.87168 | 0.623772 | 1.426166 | | -0.00022 |
| **Input to hidden layer weights for NARX for CPR** | | | | | | | | **b1** | **b2** | **Out weigt** |
| 0.503323 | -0.71638 | -0.47701 | 0.125033 | 0.507502 | 0.72037 | 0.468527 | -0.18588 | -1.90242 | 0.421806 | 0.212496 |
| -0.11146 | 0.073585 | -0.4376 | 0.853556 | 0.586071 | 0.163054 | 1.188285 | -0.50563 | -1.63039 | | 0.413975 |
| -0.67478 | -0.11342 | -0.41249 | -0.00595 | 0.788535 | 0.641279 | 0.553195 | -1.26158 | 1.321382 | | 0.091161 |
| -0.97988 | -1.21134 | -0.82727 | -0.19104 | -0.59935 | 0.564876 | -0.38709 | -0.36939 | 0.879461 | | -0.02051 |
| -0.8517 | -0.45204 | 1.70241 | -0.36506 | 0.690614 | -0.1475 | 0.466457 | -0.88556 | 0.575445 | | -0.21187 |
| -0.01814 | -0.64667 | 0.160584 | -0.47242 | -1.01391 | 0.164566 | -0.55218 | 0.444837 | 0.668483 | | 0.285847 |
| 0.401385 | -0.08495 | 0.249054 | 1.17595 | -0.03958 | 0.68003 | -0.23164 | -0.84265 | 0.237699 | | -0.12729 |
| 0.364545 | -0.77468 | -0.81799 | -0.73763 | -0.74132 | 0.434145 | -0.54196 | 0.534379 | 0.544685 | | -0.21285 |
| 0.044248 | 0.288256 | -0.83473 | 1.038335 | -0.488 | 0.470336 | -0.69567 | -0.53784 | 0.808644 | | -0.24943 |
| 0.006685 | -0.21683 | 0.572938 | -0.36418 | 0.161576 | 0.388783 | 1.240253 | -0.3904 | 1.474853 | | -0.12172 |
| 0.95895 | -0.54928 | 1.013793 | 0.832018 | 1.305433 | -0.13551 | 1.366962 | 0.651144 | 0.801844 | | -0.01882 |
| -0.25134 | -0.13347 | 0.30388 | -0.79301 | 0.293759 | -0.65055 | 0.203446 | 0.425995 | -0.75708 | | -1.03736 |

**Table 4.36:** Details of neural network weights and biases for NARX

| Input to hidden layer weights for NARX for ROT | | | | | | | | b1 | b2 | Out weigts |
|---|---|---|---|---|---|---|---|---|---|---|
| 0.029772 | 0.705284 | -0.30602 | -0.35425 | -0.29525 | -0.50931 | 0.216753 | 0.434033 | 0.510367 | 2.288865 | 1.584375 |
| 0.174956 | 1.138612 | -0.41932 | 0.065062 | -1.16675 | 1.003772 | -0.5229 | 0.660104 | 2.668973 | | -0.10273 |
| -0.90139 | 0.993572 | -2.48943 | -0.30514 | -0.93401 | 0.582078 | -0.81605 | -1.00172 | 0.509277 | | -0.04335 |
| 0.542385 | -0.93331 | 1.755901 | 0.001055 | -0.13889 | 0.420421 | 0.681695 | 0.443185 | -1.73339 | | 0.003188 |
| 1.536829 | -0.28014 | 0.116769 | -1.26529 | -1.62675 | -0.65721 | -0.96397 | -0.75372 | 0.3584 | | 0.006362 |
| 0.929333 | -0.99958 | -0.48044 | 0.541373 | 0.377175 | -0.55114 | -1.02919 | -0.73352 | 0.35186 | | 0.180433 |
| -0.07363 | -0.5174 | -0.32896 | 0.547945 | -2.03032 | 0.833088 | -0.49197 | -0.03778 | 0.035708 | | -0.14221 |
| 1.167861 | -0.45878 | -0.14524 | -0.13484 | 2.740678 | 0.640929 | 0.680655 | -1.09811 | -0.16433 | | -0.08487 |
| 0.634306 | -0.63064 | 1.563144 | -0.41896 | 0.871549 | 0.439827 | 1.675112 | 0.072399 | -0.52895 | | 0.086457 |
| 0.063244 | 1.004131 | -0.14891 | -1.03107 | -1.52035 | -0.68153 | -1.01494 | -0.42296 | 0.932719 | | 0.033425 |
| -3.25777 | 0.431661 | -1.23404 | 0.468031 | 0.557453 | -0.84408 | -0.9327 | 0.263094 | 3.640762 | | -2.90193 |
| 0.363839 | -0.86273 | -0.05265 | 0.862713 | -0.53155 | 0.154161 | -0.96584 | 0.359778 | 1.414064 | | 0.278643 |
| **Input to hidden layer weights for NARX for TOT** | | | | | | | | **b1** | **b2** | **out weigts** |
| 0.339865 | -0.36473 | 0.82251 | -0.24941 | -0.456 | -0.5815 | -0.78126 | 0.241962 | -1.40923 | 0.001599 | -0.21899 |
| -0.38103 | -0.57202 | 0.900245 | -0.3966 | 0.29889 | -0.49352 | 1.171371 | -0.40872 | 0.964682 | | 0.05336 |
| -0.00249 | -0.13105 | 0.399095 | -0.05656 | -0.49229 | -0.40359 | -0.36121 | 0.55407 | 1.806974 | | -0.33165 |
| 1.469415 | -0.15934 | -1.03295 | -0.01021 | -1.12394 | -0.5959 | 0.152363 | 0.494607 | -0.19113 | | 0.067932 |
| 0.253826 | -1.81284 | -0.57635 | -0.08533 | 1.916763 | -1.41291 | -0.19332 | -0.49876 | -0.44921 | | 0.00852 |
| 1.10645 | 0.817234 | 1.398355 | -0.45191 | 0.821758 | 0.414097 | 1.012514 | -1.32947 | 0.882598 | | -0.08858 |
| -0.18372 | -0.10922 | 0.021646 | -0.30466 | 0.4823 | 0.716844 | -0.24093 | -0.68763 | -0.68817 | | 0.180843 |
| -0.58877 | 0.163902 | 0.077397 | 0.683796 | -0.6929 | 0.337611 | 1.296028 | 2.141586 | -0.65154 | | 0.005869 |
| -0.17324 | 0.101746 | 0.122606 | -0.30085 | 0.263101 | -0.88286 | -0.1696 | -0.20387 | 0.083356 | | -0.09119 |
| -0.53432 | 1.081056 | 0.107225 | -0.4357 | -1.14923 | -0.09638 | 0.025883 | -0.83094 | -0.4253 | | 0.007224 |
| 0.076845 | -0.14955 | 0.803665 | -0.70628 | 1.002142 | 0.416533 | -0.39514 | 0.09392 | 1.61008 | | 0.57946 |
| 0.054525 | -0.08531 | 0.064555 | 0.136871 | -0.25501 | -0.23515 | -0.05025 | 0.16814 | -0.27436 | | 0.812564 |

**Table 4.37:** MISO model result for NARX showing neurons and performance

| Variables | Neurons | Performance (MSE) | Performance (MAPE) |
|---|---|---|---|
| Rotational Speed | 12 | 192 | 0.3660 |
| Compressor Outlet Temp | 12 | 0.782 | 0.0856 |
| Turbine Outlet Temperature | 12 | 2.07 | 0.1491 |
| Compressor Pressure Ratio | 12 | 0.00395 | 1.0384 |



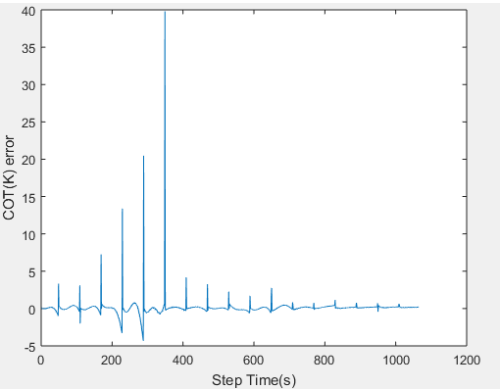
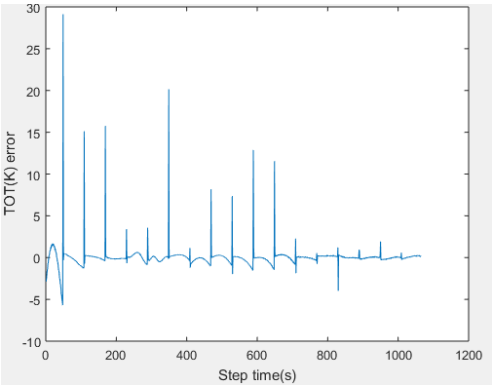




**Figure 4.23:** Error plot of ROT (rpm). **Figure 4.24:** Error plot of COT (K).



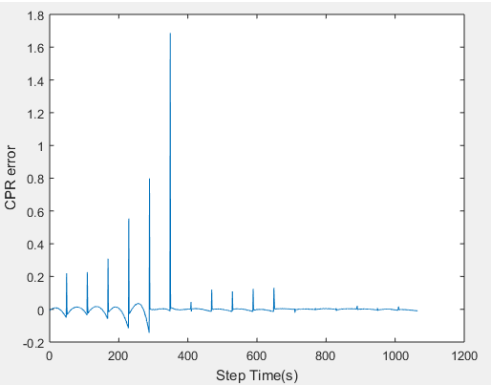




**Figure 4.25:** Error plot of TOT (K). **Figure 4.26:** Error plot of CPR.

**Table 4.38:** Summary and Ranking of ANN Performance for the MISO models

| | COT | | | CPR | | | TOT | | | ROT | | | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Network** | **r1** | **r2** | **r** | **r1** | **r2** | **r** | **r1** | **r2** | **r** | **r1** | **r2** | **r** | **r** |
| FF_BPP | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 2 | 2 | 2 | 9 |
| LayRecNet | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 3 | 3 | 3 | 11 |
| NARX | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 |

**Table 4.39:** Top (7) heads of simulated MISO results with LayRecNet

| **Result for data simulated with LayRecnet sample 1 models** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | COT Comparison | | CPR Comparison | | TOT Comparison | | ROT Comparison | |
| S/N | Targets | SimOutput | Targets | Simoutput | Targets | SimOutput | Real Rot | Simoutput |
| 1 | 439.623 | 441.9814 | 0.144336 | 0.180016 | 440.577 | 460.3395 | 641.247 | 678.2905 |
| 2 | 439.245 | 441.1731 | 0.148291 | 0.180093 | 442.543 | 460.3718 | 644.949 | 678.3044 |
| 3 | 438.868 | 440.7295 | 0.152246 | 0.180171 | 444.509 | 460.394 | 648.652 | 678.319 |
| 4 | 438.49 | 439.9552 | 0.155212 | 0.180359 | 446.475 | 460.4565 | 652.354 | 678.3614 |
| 5 | 438.112 | 439.5357 | 0.159166 | 0.180537 | 448.441 | 460.5007 | 656.056 | 678.4071 |
| 6 | 437.734 | 438.9323 | 0.162133 | 0.180923 | 450.406 | 460.6153 | 659.759 | 678.5116 |
| 7 | 437.356 | 438.2187 | 0.166087 | 0.181522 | 452.372 | 460.8223 | 663.461 | 678.7501 |

**Table 4.40:** Top (7) tails of simulated MISO results with LayRecNet

| Result for data simulated with LayRecnet sample 1 models | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | COT Comparison | | | CPR Comparison | | | TOT Comparison | | | ROT Comparison | |
| S/N | Targets | SimOutput | | Targets | Simoutput | | Targets | SimOutput | | Real Rot | Simoutput |
| 473 | 602.344 | 602.2987 | | 8.267845 | 8.267733 | | 545.133 | 545.1883 | | 3025.632 | 3025.475 |
| 474 | 602.397 | 602.3033 | | 8.267853 | 8.26729 | | 545.123 | 545.2403 | | 3025.787 | 3025.525 |
| 475 | 602.451 | 602.3726 | | 8.267861 | 8.267334 | | 545.113 | 545.3888 | | 3025.942 | 3025.701 |
| 476 | 602.504 | 602.4668 | | 8.268858 | 8.268654 | | 545.103 | 545.4633 | | 3026.098 | 3025.985 |
| 477 | 602.557 | 602.5801 | | 8.268858 | 8.271106 | | 545.093 | 545.3956 | | 3026.253 | 3026.312 |
| 478 | 602.611 | 602.5848 | | 8.268866 | 8.270497 | | 545.083 | 545.4679 | | 3026.408 | 3026.384 |
| 479 | 602.664 | 602.5896 | | 8.269863 | 8.269786 | | 545.073 | 545.5453 | | 3026.563 | 3026.464 |
| 480 | 602.718 | 602.5943 | | 8.269871 | 8.268912 | | 545.063 | 545.628 | | 3026.719 | 3026.553 |

**Table 4.41:** Test/Simulated result from layrecnet model showing neurons and performance

| Variables | Neurons | RMSE | MSE | MAPE |
|---|---|---|---|---|
| Rotational Speed | 12 | 3.3794 | 11.4206 | 0.0885 |
| Compressor Outlet Temp | 12 | 0.2455 | 0.0593 | 0.0263 |
| Turbine Outlet Temperature | 12 | 1.8238 | 3.3262 | 0.0847 |
| Compressor Pressure Ratio | 12 | 0.0035 | 1.2052 | 0.3469 |

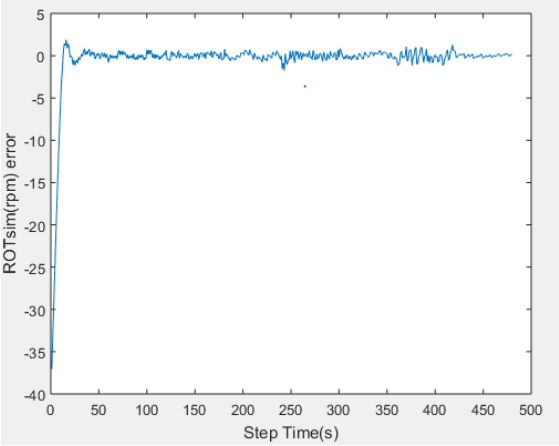



**Figure 4.27:** Error plot of ROTsim (rpm**).**

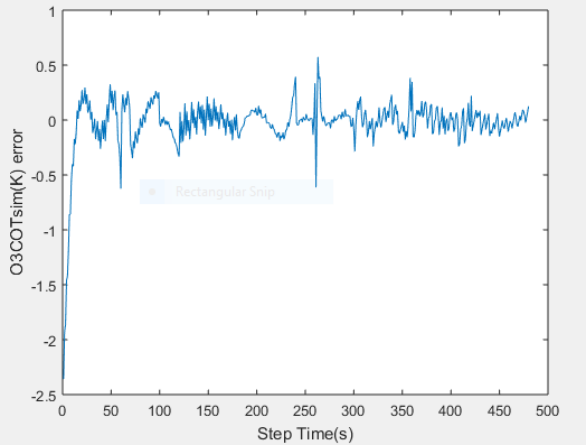



**Figure 4.28:** Error plot of COTsim (K).

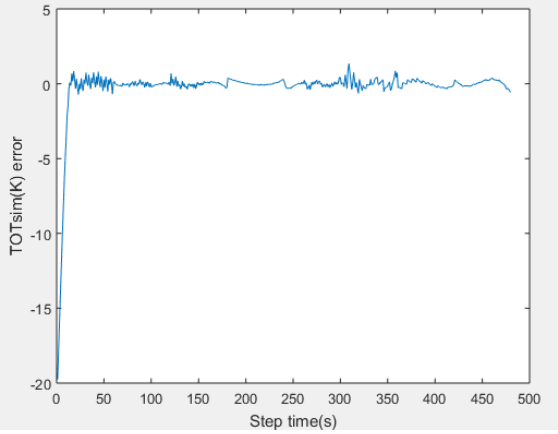



**Figure 4.29:** Error plot of TOTsim (K).

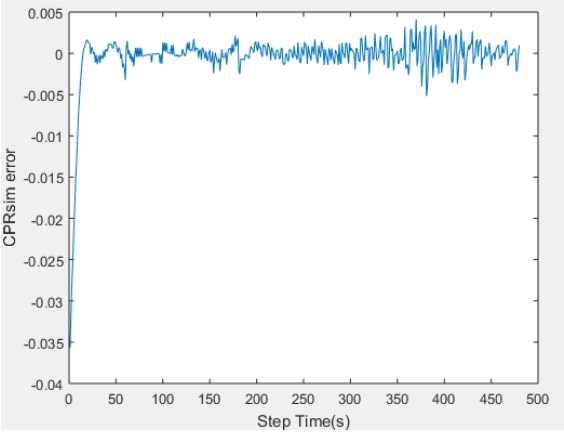



**Figure 4.30:** Error plot of CPRsim.

## 4.2 DISCUSSION

### 4.2.1 MIMO model discussion

**FF_BPP MIMO model:** The model give a close fit with the real data as depicted by a MSE of 1.74 and MAPE error of 1.4249. Furthermore the plots which shows the performance per epoch (fig. 4.8) reveals that the model was able to learn the data well as the validation, training and test error plots is closely aligned with one another with gradual descent of its

error(the likelihood of local minima is slim). Moreover the MIMO model is ranked in the second position in comparison with the two other ANN networks.

**LayRecNet model:** The model is ranked as the best performing model among the networks used for modelling turbine start up. This is as a result of the network having a MSE of 1.12 and MAPE of 0.7310 outperforming the two other networks. Furthermore from the performance plots it shows that the model did not overfit the data as the validation, test and training performance per epoch were closely aligned together with its gradual descent after which the training stop.

**NARX MIMO model**: The NARX network is the least performing network because it gives a MSE performance of 19.8 and MAPE performance of 1.7358 respectively. Furthermore; from the performance plot as shown in figure 4.10 the network struggled to learn the data well as the test and validation performance is distant away from the training performance. The plot further shows that the training data set is slightly able to learn from the data as the error is on a downward decrease that is not the case for the validation and test error as the MSE was on an upward rise which further proves that the NARX network could not learn the data well.

**MIMO model simulation:** The highest ranking network in the modelling of gas turbine start up was used to simulate the developed MIMO model with the Layrecnet network. Three performance metrics is given and their performance using these metrics are 0.6065 for MSE, 1.5925 for MAPE and 0.7788 for RMSE respectively. The results as indicated by the performance metrics implies that the layer recurrent network model was well able to generalize even as new or unseen data is introduced it is able to give predicted outputs very close to the real or target data.

### 4.2.2 MISO model discussion

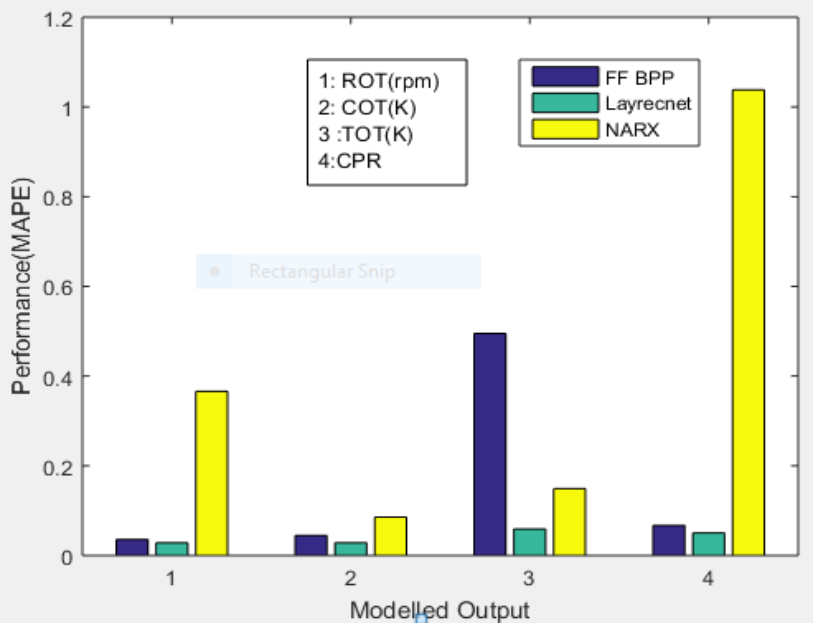



**Figure 4.31:** Summary of MAPE performance of the three networks.

**Rotational speed (rpm):** The MISO model for the rotational speed using the three different networks gave a fairly good performance, judging with the NARX model because its MSE is 192 but the MSE of FF BPP network and the layer recurrent network is 2.55 and 1.64 respectively. Layer recurrent network had the least error with MAPE of 0.0286 and MSE of 1.64.

**Compressor outlet temperature (k):** The COT MISO model from the three networks had a good performance with an excellent marginal error. Layer recurrent network gives the best performance for the MISO model of the COT with MSE of 0.0684 and MAPE of 0.0286 followed by FF BPP network with MSE of 0.158 and MAPE of 0.0448, while the least performing model for this manouver is the NARX model with 0.782 MSE and 0.0856 MAPE.

**Turbine outlet temperature (k):** FF BPP has the least MSE, while layer recurrent model has the least MAPE for the TOT model. It has MSE of 0.296 and layer recurrent with MAPE of

0.0591. The NARX model for TOT was also satisfactory but was the least performing model of the three networks with MSE of 2.07 and MAPE of 0.149.

**Compressor pressure ratio:** Layer recurrent network outperforms the two other networks with MSE of 1.18E-5 and MAPE of 0.0503 followed by FF BPP network with MSE of 2.03E-5 and MAPE of 0.0671. The NARX model also has a good performance with MSE of 0.00395 and MAPE of 1.0384.

### 4.2.3 Simulated MISO model discussion

Layer recurrent neural network from the manouver of the start-up data ranked highest among the three netowrks because it has the least minimum MSE and MAPE compared to the two other networks. The test data has 480 data points with same interval as the modelled data. The test data is similar to the data used for simulation but was obtained when no load had been given to the turbine. The RMSE of the simulated model is satisfactory as the variables gave a close fit to their real values with RMSE for ROT speed being 3.3794, COT (K) of 0.2455, TOT (K) of 1.8238 and CPR with 0.0035.The MAPE for the test data has an exceptional performance where ROT IS 0.0885, COT (K) is 0.0263, TOT (K) being 0.0847 and CPR with a slightly higher error of 0.3469.

# CHAPTER FIVE

# CONCLUSION, AND RECOMMENDATION

## 5.1 Conclusion

This research was motivated by the frequent and incessant shutdown of gas power plants in Nigeria; because of lack of a proper condition monitoring and predictive maintenance tool being identified as one of the banes facing her power industry. The development of these tools requires the set-up of models that depict the operation of the turbine. However faults or problems arising from turbines are difficult to detect during the steady state operation of turbines but are best detected at transient or during the dynamic operation. The research describes the investigation of ANN capacities for gas turbine modelling using operational data from a General Electric (formerly Alston) GT13E2 power gas turbine . Operational data was primarily chosen to evaluate the possibilities of creating an accurate performance-predictive ANN model representing an actual machine. Besides creating a precise model, various real-life aspects, associated with the modelling of an actual machine, were examined. These real-life aspects included everything from data acquisition and filtering to specific operational conditions, such as start-up and loading condition.

MIMO and MISO models were developed for the GT13E2 formerly Alston HDGT. Towards this end it was observed that the MIMO models produced a higher deviation using mean square error (MSE), and mean of absolute percentage error (MAPE) across all the three ANN models compared to the MISO models, w here individual dependent features were treated individually. The result from MIMO models showed that FF BPP, and layer RNN is a better ANN tool for dynamic time series modelling compared to the much touted Non Linear Auto Regressive network with Exogenous Input(NARX), with layer RNN outperforming the other two networks in MSE and MAPE, followed by FF BPP network respectively. Prediction

using dynamic MISO models aligns closely with the target data, with layer RNN taking the lead in rotational speed, compressor outlet temperature, compressor pressure ratio and turbine outlet temperature as output variables respectively using MAPE deviation, followed by FF BPP in turbine outlet temperature using MSE standard deviation and the least performing model is the NARX network.

MIMO and MISO models developed from data 1 manouver was validated by testing the model with the new unseen data similar in maneuver to data 1 and the deviation was minimal and satisfactory. Similar works on modelling and simulation of turbine start-up operation (Asagri et al 2014b) where the author simulated three maneuvers with three trained models and had a satisfactory result. However; the best performing model was obtained with a deviation of 3.5 for compressor outlet temperature while the least performing simulation model in this research has a MAPE of 0.3469 for COT. This improvement was a result of the access to an additional measured input variable (network load) and exploitation of alternative ANN networks.

The outcome from the research shows that layer RNN can reproduce a very complicated and usually difficult-to-model unsteady behaviour in gas turbines and can capture system dynamics with acceptable accuracy with FF BPP coming a close second while the least performing network is the NARX network which captured system dynamics with moderate accuracy. It was shown that neural networks can be considered a reliable alternative to conventional modelling methods in system identification and modelling. Therefore, the use of ANN rendered it possible to create tailor-made models, with very high prediction accuracies, using operational data for specific gas turbines.

With proper training, data and parameter selection, it was also feasible to achieve very high prediction accuracies. The described ANN models are plant specific; however, the method is general and thereby applicable to other plants and power plant configurations. The results of

this black box method can be used for real-time control and diagnostics of gas turbine. With the developed tool, the condition of a plant could thus be monitored by simultaneously evaluating possible deviations, such as degradation or faults. Hence, by using the developed tool, an optimization of the plant operation and maintenance is rendered possible. It was demonstrated that realistic models, for condition monitoring, can be developed and utilized on-site, with the aid of artificial neural network.

## 5.2 Recommendation

The comparative study using three different artificial neural networks for dynamic modelling yielded an excellent performance with the modelled and simulated data giving outcome very close to the real data. However, for a gainful application of the study emphasis needs to be given to the following:

- Elimination of administrative bottlenecks in the procurement and assessment of company's data as these constraints makes it a herculean task for researchers to assess data for research purpose.
- Effort has to be put in place to ensure that the models developed are utilized by these plants for online condition monitoring through proactive synergy between plant owners and researchers alike (In the industry and academia).

In the field of turbine modelling further works and attention still need to be drawn to the following areas:

- Development of transient models for aero derivative and combined heat and power turbine Plants for condition monitoring, sensor validation and predictive maintenance.
- Application of deep neural network (DNN) in modelling erosion in turbine exhaust and blades with optical boroscopy.

- ❖ Application of industrial internet of things (IIOT) in the development of online gas turbine condition monitoring tool using ANN transient models.
- ❖ Development of CovNets for predicting fouling models in turbine compressors; a computer vision approach.

### 5.3 Contributions to knowledge

The modelling and simulation of GE gas turbine was achieved with the aid of black box modelling technique. The research was able to model the start-up/ dynamic behaviour of the GE turbine using three ANN networks. From the predicted outcome layer RNN is therefore prescribed as a computational tool for gas turbine modelling. This is because the network ranked the highest from the chosen performance metrics (MSE and MAPE) among the three utilized networks. The research probed further into simulation of the developed model with unseen data to test its ability to predict outcomes. Its RMSE and MAPE result for ROT speed, TOT, COT and CPR range from 0.0035 and 3.3794 for the MISO models while its MIMO model produced a RMSE of 0.7788. Furthermore the research succeeded in developing a start-up dynamic model for Afam power gas plant. If the models are deployed as a standalone application or on a web server they can be used in real time for online condition monitoring of the Plant.

## REFERENCES


Aklilu B., Sidahmed M., Suleiman S., Amare F., Ghazali S., (2016). Development And Validation Of A Twin Shaft Industrial Gas Turbine Performance Model. *ARPN Journal of Engineering and Applied Sciences, 22*(11), 13365–13371

Arriagada, J. (2001). Introduction of intelligent tools for gas turbine based, small-scale cogeneration, Licentiate thesis, Lund University, Sweden

Asgari, H. (February, 2014). Modelling and simulation of a single-shaft gas turbine. *University of Canterbury. Mechanical Engineering*. https://doi.org/10.26021/3555

Asgari, H. (2014). Modelling, simulation and control of gas turbines Using Artificial Neural Networks. *University of Canterbury. Mechanical Engineering*. https://doi.org/10.26021/3555

Asgari, H., Chen, X. and Sainudiin, R.(2013). Modelling and simulation of gas turbines. . *International Journal of Modelling, Identification and Control*, *20*(3), 253. https://doi.org/10.1504/IJMIC.2013.057137

Banzhaf, W. (1998). Genetic Programming: An introduction on the automatic evolution of computer programs and its Applications.

BENYOUNES A., HAFAIFA A., KOUZOU A., GUEMANA M., (2017). Fuzzy Modeling and Simulation of Gas Turbine using Fuzzy Clustering Algorithm. Technical University Sofia, Plodiv, Bulgaria.

Caudill, M. (1989). Neural Network Primer: Part I," AI Expert, vol. 2, no. 12.

Chaibakhsh A., Amirkhani S., (March, 2018). A Simulation Model for transient behaviour of Heavy-duty Gas Turbines. *Journal of Applied Thermal Engineering.*

Chow, T., & Cho, S. (2007). Neural Networks and Computing: Learning Algorithms and Applications. *World Scientific Publishing Company Incorporated* vol. 7.

Cohen, H., Rogers, G., & Saravanamuttoo, H. (1996). Gas Turbine Theory, *4*th ed. *Longman Group Limited*. Longman House Burnt Mill, Harlow Essex CM20 2JE, England.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals, and Systems (MCSS).

Darell, D.M., (2001). Neural Network fundamentals for scientist and Engineers. ECOS20.01.

Duch, W. & Jankowski, N. (1999). Survey of neural transfer functions," *Neural Computing Surveys*, vol. 2, no. 1

Dynamic models for combined cycle plants in power system studies. (1994). *IEEE Transactions on Power System*, Vol. *9*.

*Encyclopedia of Physical Science and Technology—3rd Edition*. (n). Retrieved November 2, 2021, from https://www.elsevier.com/books/T/A/9780122274107

Efficiency for Next Generation Nuclear Power Plants. Retrieved from https://www.research gate.net/publication/300857212

Energy Solutions Center. (2016). Understanding combined heat and power . Retrieved from https:// understandingchp.com

Engaged in Thermodynamics (n.d). Basic Gas Turbine Engines. Retrieved from https://cset.mnsu.edu/engagethermo/components_gasturbine.html

Engineering Training (n.d). Basic Gas Turbine Engines. Retrieved from https://fas.org/man/dod-101/navy/docs/swos/eng/60b-104.html

Fast, M., Assadi, M., & De, S. (2009). Development and multi-utility of an ANN model for an industrial gas turbine. *Journal of Applied Energy*, Vol. 86, Issue 1

Fast, M. (December, 2010) Artificial Neural Networks for Gas Turbine Monitoring. Doctoral Thesis Division of Thermal Power Engineering, Department of Energy Sciences Faculty of Engineering Lund University Sweden.

Fundamentals of Gas Turbine Engines [PDF] . Retrieved from htpps://www.cast-safety.org

Gang, L. & Yu, F.(2005). A hybrid nonlinear autoregressive neural network for permanent-magnet linear synchronous motor identification, in *Proceedings of the Eighth International Conference on Electrical Machines and Systems, ICEMS*, vol. 1. IEEE.

Gas turbine dictionary definition. (2016). Retrieved https://www.yourdictionary.com

Gas Turbine Tutorials (n.d). Retrieved from http://gasturbinetutorial.blogspot.com

General Electric. (2019). Combined cycle power plant: How it works. Retrieved from https://www.GE.com>knowledge base

GlobalEnergy Observatory(n.d) Energy System Overview. Retrieved from http://globalenergyobservatory.org/list.php?db=PowerPlants&type=Gas.

Google Images (n.d). Retrieved from http:// www. cs. csi. cuny. Edu /~ natacha / Publication /URC07 /BP_Ohnmar.pdf.

Guru99 (n.d) Back Propagation Neural Network: Explained with Simple Example. Retrieved from https://wwww.guru99.com/backpropogationneuralnetwork.

Hagan, M.T, Demuth, H.B & Jesus, O.D.(2002). An Introduction to the Use of Neural Networks in Control Systems," *International Journal of Robust and Nonlinear Control*, vol. 12, no. 11

Hagan, M. T., Demuth, H.B., & Beale, M. H. (2002). Neural Network Design, Boulder, Colorado: *Campus Publishing Service, Colorado University Bookstore*, University of Colorado at Boulder

Hagan, M., Demuth, H.B., Mark, B.(1997). Neural network design. *Association for Computing Machinery*. PWS Publishing Co. Boston, Massachusets, USA

Hagan, M.T. & Demuth, H.B. (1999). Neural Networks for Control, in American Control Conference, San Diego, CA, USA

Haykin S., (1999) Neural networks, a comprehensive foundation. 2nd ed. New Jersey, USA: Prentice-Hall Inc.

Haykin, S. (1994). Neural networks: a comprehensive foundation. *Prentice Hall PTR* Upper Saddle River, NJ, USA;

Horne B., and C. L. Giles C.,(1994) An Experimental comparison of Recurrent Neural Networks," *Advances in Neural Information Processing Systems.*

Kim, Y., Lewis, F., & Abdallah, C. (1997). A dynamic recurrent neural-network based adaptive observer for a class of nonlinear systems," *Automatica*, vol. 33, no. 8,

Kodogiannis, V., Lisboa P., & Lucas J. (n.d). Neural network modelling and control for underwater vehicles, *Artificial Intelligence in Engineering*, vol. 10, no. 3

Krose, B., & Smagt, P. (1996). An introduction to neural networks: The University of Amsterdam.

Kumar, G. (2007). Artificial Neural Nework, vol. 2.

Loboda, I., & Feldshteyn, Y. (2011). Polynomials and Neural Networks for Gas Turbine Monitoring: a Comparative Study. *International Journal of Turbo and Jet-Engines*, Vol. 28.

Malikar, K. (January, 2018). 6 types of ANN currently being used in ML. *Analytics India Magazine*. Retrieved from https://www.analyticsinddiamag.com

Mandic, D. & Chambers, J. (2001). Recurrent neural networks for prediction: Learning algorithms, architectures and stability. *John Wiley & Sons, Inc*.

Masdi M.1, Mohammadreza T. B. & Abdul K. Z.(2015). Methodology for short-term performance prognostic of gas turbine using recurrent neural network

Mathworks (n.d) Levenberg-Marquardt backpropagation. Retrieved from https://www.mathworks.com/help/deeplearning/ref/trainlm.html;jsessionid=795d970ec83ba23a246c89e9151c.

Meherwan, P.B. (2017). *GasTurbine Engineering Handbook 3$^{RD}$ edition*. *Gulf Professional Publishing*, Burligton MA, USA.

Menezes, J. & Barreto, G. (2006). A new look at nonlinear time series prediction with narx recurrent neural network, in *Proceedings of the Ninth Brazilian Symposium on Neural Networks*. IEEE, 2006

Mishra, P. (June, 2016). Closed cycle turbine: construction, working diagram. Retrieved from http://www.mechanicalbooster.com

Mohieddine, J. & Kroll, A. (2004). Hydraulic Servo-Systems: Modelling, Identification, and Control. *Springer Verlag Londom*

Norgaard, M. Ravn, O. Poulsen N.K., & Hansen L.K. (2000). Neural networks for modelling and control of dynamic systems: *A Practitioner's Handboo*k. Springer-Verlag London

Olausson, P. (2003). On the selection of methods and tools for analysis of heat and power plants, *Division of thermal power engineering* ,Lund Institute of Technology, P.O.Box 118, SE-221 00 LUND, Sweden.

Paoli, N. (2009). Simulation models for analysis and optimization of gas turbine cycles. Pisa University. Retrieved from http:etd.adm.unipi.it/theses/available/etd-04062009-132133/

Ren Y., Wana J., Zheng Q., Liu Y., and Hu Z., (2019) Static and Dynamic Performance Modelling and Simulation of a Micro-Turbine. In: Proceedings of 2018 Chinese Intelligent Systems Conferernce. Lecture Notes in Eectrical Engineering; Springer Singapore.

Robb, D. (2017). Aeroderivative gas turbines. *The global journal of energy equipment Internatinal Turbomachinery*. *Turbomachinery Magazine.* Retrieved from www.turbomachinerymag.com

Rowen, W.I. (1983). Simplified mathematical representations of heavy-duty gas turbines. *ASME journal of Engineering for Power* vol. *105*

Rumelhart, D.E., Hinton, G.E & Williams, R.J. (1986). Learning internal representations by error propagation, Parallel distributed processing: *Explorations in the microstructure of cognition*, vol. 1: foundations. MIT Press, Cambridge, MA

Russell S., and Norvig P., (1995) Artificial intelligence: A modern approach. Prentice-Hall, Inc.

Seyab, R. & Cao Y. (2008). Differential recurrent neural network based predictive control, *Computers & Chemical Engineering*, vol. 32, no. 7

Siegelmann, H., Horne B., & Giles C.(2008). Computational capabilities of recurrent narx neural networks."

Soares, C. M. (n.d).Gas Turbines in simple cycle and combined cycle applications [PDF File]. Retrieved from https://www.netl.doe.gov.

Sohlberg, B. (1998). Supervision and Control for industrial Processes: Using Grey Box Models Predictive Control, and Fault Detection Methods. *Advances in Industrial Control, Perspective in Neural Computing*. Springer-Verlag London Limited.

Thiago S. P, Manuel E. C, Marcelo J. C, & Marco A. C(2018). Nonlinear model predictive control applied to transient operation of a gas turbine

Thirunavukarasu, E. (2013). Modeling and Simulation Study of A Dynamic GasTurbine System In A Virtual Test Bed Environment.(Master's thesis).Retrieved from https://scholarcommons.sc.edu/etd/2254.

USAID (n.d). Nigeria power Africa fact sheet. Retrieved from https://www.usaid.gov>powerafrica.

Vatani, A. (August, 2013). Degradation prognostics in gas turbine engines using neural networks. Concordia University Montrteal, Quebec, Canada.

Wartisila (n.d). Gas turbine for power generation-introduction. Retrieved from https://www.wartsila.com>learnmore.

Zeng, D., Zhou D., Tan, C., & Jiang, B. (January, 2018). Research on Model-Based Fault Diagnosis for a Gas Turbine Based on Transient Performance. *Journal of applied science*

Zohuri, B. (March, 2015). Gas Turbine working principles[PDF FILE]. Retrieved from https://www.researchgate.net/publication/300857212

## APPENDICES

**Appendice A**

**MATLAB ANN Code used for Model Development**

**Layer Recurrent Network for MIMO Model**

Sample_1 = Xlsread(‘sample_1’);

I1 = sample_1(:,2:5);

%copy xlsx data from colum1:column 4 symbolizing input variables

O1 = sample_1(:,6:8); %copy column 5-9 with all the rows showing the output variables

I1 = I1’; % transpose the input to suit the network function

O1 = O1’ ;%transpose output to suit network function

```
% This script assumes these variables are defined:

%   I1 - input data.
%   O1 - target data.
rng('default')
x = I1;
t = O1;

% 'trainlm' is often faster.
% 'trainbr' takes longer but may be better for challenging problems.
trainFcn = 'trainlm';  % Levenberg-Marquardt backpropagation.
Delay = 1;
% Create a Fitting Network
hiddenLayerSize = 12;
net = layrecnet(Delay,hiddenLayerSize,trainFcn);

% Choose Input and Output Pre/Post-Processing Functions
% For a list of all processing functions type: help nnprocess
net.input.processFcns = {'removeconstantrows','mapminmax'};
net.output.processFcns = {'removeconstantrows','mapminmax'};
```

```
% Setup Division of Data for Training, Validation, Testing
% For a list of all data division functions type: help nndivide
net.divideFcn = 'dividerand';  % Divide data randomly
net.divideMode = 'sample';  % Divide up every sample
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

net.performFcn = 'mse';  % Mean Squared Error

% Choose Plot Functions
% For a list of all plot functions type: help nnplot
net.plotFcns = {'plotperform','plottrainstate','ploterrhist', ...
    'plotregression', 'plotfit'};

% Train the Network
[net,tr] = train(net,x,t);

% Test the Network
y = net(x);
e = gsubtract(t,y);
performance = perform(net,t,y)

% Recalculate Training, Validation and Test Performance
trainTargets = t .* tr.trainMask{1};
valTargets = t .* tr.valMask{1};
testTargets = t .* tr.testMask{1};
trainPerformance = perform(net,trainTargets,y)
valPerformance = perform(net,valTargets,y)
testPerformance = perform(net,testTargets,y)

% View the Network
view(net)

% Plots
% Uncomment these lines to enable various plots.
%figure, plotperform(tr)
%figure, plottrainstate(tr)
%figure, ploterrhist(e)
%figure, plotregression(t,y)
%figure, plotfit(net,x,t)

if (false)

    genFunction(net,'myNeuralNetworkFunction');
    y = myNeuralNetworkFunction(x);
end
if (false)
    % Generate a matrix-only MATLAB function for neural network code
    % generation with MATLAB Coder tools.
```

```
    genFunction(net,'myNeuralNetworkFunction','MatrixOnly','yes');
    y = myNeuralNetworkFunction(x);
end
if (false)
    % Generate a Simulink diagram for simulation or deployment with.
    % Simulink Coder tools.
    gensim(net);
end
```

**Appendice B**

**Feed Forward Back Propagation Network for MIMO**

```
Sample_1 = Xlsread('sample_1');

I1 = sample_1(:,2:5);

%copy xlsx data from colum1:column 4 symbolizin input variables

O1 = sample_1(:,6:9); %copy column 5-9 with all the rows showing the output variables

in = I1'; % transpose the input to suit the network function

rot = O1' ;%transpose output to suit network function

%   in - input data.
%   rot - target data.

x = in;
t = rot;

trainFcn = 'trainlm';  % Levenberg-Marquardt backpropagation.

% Create a Fitting Network
hiddenLayerSize = 12;
net = feedforwardnet(hiddenLayerSize,trainFcn);

% Setup Division of Data for Training, Validation, Testing
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

% Train the Network
[net,tr] = train(net,x,t);

% Test the Network
y = net(x);
e = gsubtract(t,y);
```

```
performance = perform(net,t,y)

% View the Network
view(net)
```

**Appendice C**

**Non Linear Auto Regressive Network with Exogenous Input for MIMO**

```
Sample_1 = Xlsread('sample_1');

I1 = sample_1(:,2:5);

%copy xlsx data from colum1:column 4 symbolizin input variables

O1 = sample_1(:,6:9); %copy column 5 with all the rows showing 1 of the output

I5 = I1'; % transpose the input to suit the network function

O5 = O5' ;%transpose output to suit network function


rng('default')
X = tonndata(I5,true,false);
T = tonndata(O5,true,false);

trainFcn = 'trainlm';  % Levenberg-Marquardt backpropagation.

% Create a Nonlinear Autoregressive Network with External Input
inputDelays = 1:2;
feedbackDelays = 1:2;
hiddenLayerSize = 12;
net = narxnet(inputDelays,feedbackDelays,hiddenLayerSize,'open',trainFcn);

% Customize output parameters at: net.outputs{i}.processParam
net.inputs{1}.processFcns = {'removeconstantrows','mapminmax'};
net.inputs{2}.processFcns = {'removeconstantrows','mapminmax'};

% Prepare the Data for Training and Simulation
 [x,xi,ai,t] = preparets(net,X,{},T);

% Setup Division of Data for Training, Validation, Testing
% For a list of all data division functions type: help nndivide
net.divideFcn = 'dividerand';  % Divide data randomly
```

```
net.divideMode = 'time';  % Divide up every sample
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

% Choose a Performance Function
% For a list of all performance functions type: help nnperformance
net.performFcn = 'mse';  % Mean Squared Error

net.plotFcns = {'plotperform','plottrainstate', 'ploterrhist', ...
    'plotregression', 'plotresponse', 'ploterrcorr', 'plotinerrcorr'};

% Train the Network
[net,tr] = train(net,x,t,xi,ai);

% Test the Network
y = net(x,xi,ai);
e = gsubtract(t,y);
performance = perform(net,t,y)

% Recalculate Training, Validation and Test Performance
trainTargets = gmultiply(t,tr.trainMask);
valTargets = gmultiply(t,tr.valMask);
testTargets = gmultiply(t,tr.testMask);
trainPerformance = perform(net,trainTargets,y)
valPerformance = perform(net,valTargets,y)
testPerformance = perform(net,testTargets,y)

view(net)


.
netc = closeloop(net);
netc.name = [net.name ' - Closed Loop'];
view(netc)
[xc,xic,aic,tc] = preparets(netc,X,{},T);
yc = netc(xc,xic,aic);
closedLoopPerformance = perform(net,tc,yc)

numTimesteps = size(x,2);
knownOutputTimesteps = 1:(numTimesteps-5);
predictOutputTimesteps = (numTimesteps-4):numTimesteps;
X1 = X(:,knownOutputTimesteps);
T1 = T(:,knownOutputTimesteps);
[x1,xio,aio] = preparets(net,X1,{},T1);
[y1,xfo,afo] = net(x1,xio,aio);
.
x2 = X(1,predictOutputTimesteps);
[netc,xic,aic] = closeloop(net,xfo,afo);
[y2,xfc,afc] = netc(x2,xic,aic);
multiStepPerformance = perform(net,T(1,predictOutputTimesteps),y2)
```

```
.
nets = removedelay(net);
nets.name = [net.name ' - Predict One Step Ahead'];
view(nets)
[xs,xis,ais,ts] = preparets(nets,X,{},T);
ys = nets(xs,xis,ais);
stepAheadPerformance = perform(nets,ts,ys)

if (false)
.
   genFunction(net,'myNeuralNetworkFunction');
   y = myNeuralNetworkFunction(x,xi,ai);
end
if (false)

   genFunction(net,'myNeuralNetworkFunction','MatrixOnly','yes');
   x1 = cell2mat(x(1,:));
   x2 = cell2mat(x(2,:));
   xi1 = cell2mat(xi(1,:));
   xi2 = cell2mat(xi(2,:));
   y = myNeuralNetworkFunction(x1,x2,xi1,xi2);
end
if (false)

   gensim(net);
end
```

**Appendice D**

**Layer Recurrent Network for MISO model**

Sample_1 = Xlsread(‘sample_1’);

I1 = sample_1(:,2:5);

%copy xlsx data from colum1:column 4 symbolizin input variables

ROT = sample_1(:,6); %copy column 6 symbolizing rotational speed with all the rows showing the output variables

I1 = I1’; % transpose the input to suit the network function

ROT = ROT’ ;%transpose output to suit network function

```
rng('default')
x = I1;
t = ROT;

trainFcn = 'trainlm';  % Levenberg-Marquardt backpropagation.
Delay = 1;
% Create a Fitting Network
```

```
hiddenLayerSize = 12;
net = layrecnet(Delay,hiddenLayerSize,trainFcn);

net.input.processFcns = {'removeconstantrows','mapminmax'};
net.output.processFcns = {'removeconstantrows','mapminmax'};

% Setup Division of Data for Training, Validation, Testing

net.divideFcn = 'dividerand';  % Divide data randomly
net.divideMode = 'sample';
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

net.performFcn = 'mse';  % Mean Squared Error

% Choose Plot Functions
net.plotFcns = {'plotperform','plottrainstate','ploterrhist', ...
    'plotregression', 'plotfit'};

% Train the Network
[net,tr] = train(net,x,t);

% Test the Network
y = net(x);
e = gsubtract(t,y);
performance = perform(net,t,y)

% Recalculate Training, Validation and Test Performance
trainTargets = t .* tr.trainMask{1};
valTargets = t .* tr.valMask{1};
testTargets = t .* tr.testMask{1};
trainPerformance = perform(net,trainTargets,y)
valPerformance = perform(net,valTargets,y)
testPerformance = perform(net,testTargets,y)

% View the Network
view(net)

if (false)

.
    genFunction(net,'myNeuralNetworkFunction');
    y = myNeuralNetworkFunction(x);
end
if (false)
.
    genFunction(net,'myNeuralNetworkFunction','MatrixOnly','yes');
    y = myNeuralNetworkFunction(x);
end
```

```
if (false)
   % Generate a Simulink diagram for simulation or deployment with.
   % Simulink Coder tools.
   gensim(net);
end
```

**Appendice E**

**Feed Forward Back Propagation Network for MISO model**

```
% This script assumes these variables are defined:
%
%   in - input data.
%   rot - target data.
Sample_1 = Xlsread('sample_1');

I1 = sample_1(:,2:5);

%copy xlsx data from colum1:column 4 symbolizin input variables

CPR = sample_1(:,9); %copy column 9 with all the rows showing the output variables

in = I1'; % transpose the input to suit the network function

CPR = CPR' ;%transpose output to suit network function



x = in;
t = CPR;

trainFcn = 'trainlm';  % Levenberg-Marquardt backpropagation.

% Create a Fitting Network
hiddenLayerSize = 12;
net = feedforwardnet(hiddenLayerSize,trainFcn);

% Setup Division of Data for Training, Validation, Testing
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

% Train the Network
[net,tr] = train(net,x,t);
```

```
% Test the Network
y = net(x);
e = gsubtract(t,y);
performance = perform(net,t,y)

% View the Network
view(net)
```

**Appendice F**

**MATLAB Code to Simulate the Neural Network**

```
Test = Xlsread ('Test_data');

% Read excel data for MATLAB

Test_Input =  Test(:,1:4) ; % extract all the rows and colums1:4 for the input data

Test_Input = Test_Input';

%Transpose input variables

Test_Target = Test(:,5);

% extract all the rows in column 5/for rotational speed in RPM

Test_Target = Test_Traget'; %Transpose the output column

Test_Output = net(Test_Input);

% net is the neural model
%Mean of Absolute Percentge Error

MAPE = mean(abs((Test_Target – Test_Output) ./Test_Target))*100;

MSE = mean ((Test_Target – Test_Output).^2));

%Mean Squared Error

RMSE = sqrt (MSE)
```

%Root Mean Squared Error

**Appendice G**

**Link to Time Series Data and Test Data Saved in Google Sheet of Google Drive**

https://drive.google.com/file/d/17o_GQGMxapMliYUT09yN9yiKGJ91AGsI/view?usp=sharing

**Appendice H**

**Link to MIMO model results of data modelled with FF BPP, LayRecNet and NARX Networks Saved in Google Sheet of Google Drive**

https://drive.google.com/file/d/1QMK2tj9lmStsxHcrczNLmrbSHuy0bGJG/view?usp=sharing

**Appendice I**

**Link to MISO Results using Sample 1 Modelled with FF BPP, LayRecNet and NARX**

**Networks Saved in Google Sheet of Google Drive**

**FF BPP Network**

https://drive.google.com/file/d/1zKtfmXtL1cxAWRyQ-O0lPisLMiZNgbiy/view?usp=sharing

**LayRecNet Network**

https://drive.google.com/file/d/1jE4157m6MKNGTwcFtf4CvxuRXMF7g6Qi/view?usp=sharing

**NARX Network**

https://drive.google.com/file/d/146-VXAv3ZMAenoH2G_dmgcr7Xu9ajsNk/view?usp=sharing

**Appendice J**

**Link to MISO and MIMO Simulation Done using MISO LayRecNet amd MIMO LyRecNet Models with Unseen/Test Data**

**https://drive.google.com/file/d/1MMgyCCa-32wCcTQhefpd58d35vL_59jc/view?usp=sharing**