Evolutionary Algorithms For System Identification

BY

Mohammed Hussain AlBahrani

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

SYSTEMS AND CONTROL ENGINEERING

October 2021

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN- 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **Mohammed Hussain AlBahrani** under the direction of his thesis advisor and approved by his thesis committee, has been presented and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS & CONTROL ENGINEERING.

Thesis Committee

Dr. Samir Hasan Al-Amer
(Advisor)

Dr. Fouad Al-Sunni
(Member)

Dr. Sami Elferik
(Member)

Dr. Mujahed AlDhaifallah
Department Chairman

Dr. Suliman Al-Homidan
Dean of Graduate Studies

Date

The thesis work is dedicated to my parents, family, and daughter Eileen

ACKNOWLEDGMENT

First, and for the most, praises and adoration to almighty Allah for the blessing throughout all the academic learning concepts and courses plan to reach and complete the thesis successfully.

Full sincere gratitude expression to my thesis advisor Dr. Samir Al-Amer, the thesis committee members Dr. Fouad Al-Sunni and Dr. Sami El-Ferik, for giving the opportunity and providing the guidelines and right direction to do this thesis. Thanks also to Dr. Hesham Al-Fares and the Chairman of System Engineering Department Dr. Mujahed AlDhaifallah.

Special thanks for my parents, wife, and friends for keeping me motivated and patient over all the difficult circumstances and challenges all the time. And more to extend the thanks and appreciation to the two best models of the learning enhancement and encouragement my uncle Mohammed Ali AlBahrani and the direct managing to me at Emerson Process Management Ltd. Mr. Komal Damania the MEA Director of Quality and Operations in measurement solutions.

Finally, I would say thank you so much for all the ones who supported me to reach this level of completion.

Mohammed Hussain AlBahrani

TABLE OF CONTENT

CHAPTER 2: LITERATURE REVIEW

CHAPTER 3: MATHEMATICAL ANALYSIS

CHAPTER 4: THE PROPOSED ALGORITHM AND METHODOLOGY

CHAPTER 5: SIMULATION RESULT

CHAPTER 6: SUMMARY AND CONCLUSION

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABC : Artificial Bee Colony

ABC-PS: Artificial Bee Colony and Tissue P Systems

ANN : Artificial Neural Network

ARX : Auto-Regressive Exogenous

ARMAX: Autoregressive Moving-Average with Exogenous Inputs Model

BJ : Box-Jenkins

CA : Coevolutionary Algorithm

CBO : Colliding Bodies Optimization

DE : Differential Evolution

DOF : Degrees of Freedom

EA : Evolutionary Algorithm

GA : Genetic Algorithm

MSE : Mean Square Error

NARX : Nonlinear Autoregressive Exogenous

PID : Proportional Integral Derivative

PSO : Particle Swarm Optimization

RCGA : Real Coded Genetic Algorithm

RLS : Recursive Least Squares

SI : System Identification

AI : Artificial Intelligence

UPSO : Unified Particle Swarm Optimization

DE-Q : Response Surface Approach with Differential Evolution Algorithm

ABSTRACT

Full Name : [Mohammed Hussain AlBahrani]

Thesis Title : [Evolutionary Algorithms for System Identification]

Major Field : [System and Control Engineering]

Date of Degree : [Oct 2021]

The thesis focuses on the use of evolutionary algorithm in system identification. The selection of the model structure and estimating the model parameters are two important steps in the system identification. For nonlinear system, there are infinite number of model structure and obtaining the best model structure can be very difficult.

Evolutionary algorithms can be viewed as algorithm to solve optimization problems. In this work, evolutionary algorithm will be used to search for the best model structure. They will also be used in estimating the model parameters for a given model structure. The proposed algorithm will be used for identifying turbojet engine, Hammerstein model and other nonlinear system. The result will be compared with artificial neural network with different training methods.

ملخص الرسالة

الاسم الكامل: محمد حسين البحراني

عنوان الرسالة: الطرق التطويرية للتعرف على النظم

التخصص: هندسة النظم والتحكم

تاريخ الدرجة العلمية: اكتوبر 2021

هذه الرسالة تركز على إستخدام الطرق التطويرية في التعرف على النظم. إختيار هيكل النموذج الرياضي و تقدير قيم العوامل المحددة للنموذج الرياضي خطوتان مهمتان في التعرف على النظم.للنظم غير الخطية هناك عدد لا نهائي من هياكل النموذج الرياضي و يعتبر الحصول على أفضل هيكل مشكلة صعبة.

يمكن النظر إلى الطرق التطويرية على أنها طرق لإيجاد الحل الأفضل. في هذا العمل سوف يتم إستخدام الطرق التطويرية للبحث عن أفضل هيكل للنموذج الرياضي. وسوف يتم إستخدام الطرق التطويرية للحصول على عوامل النموذج الرياضي في هيكل تم إختياره مسبقاً. الطريقة المقترحة سوف يتم إستخدامها للتعرف على المحركات النفاثة و نموذج همرشتاين و غيرها من النظم الغير خطية. سوف تتم مقارنة النتائج بإستخدام الشبكات العصبية بطرق تدريب مختلفة.

CHAPTER 1

INTRODUCTION

1.1 Introduction

To control a process and make sure it is safe and efficient, one needs to have a good model of the process. There are two main approaches for obtaining the model: mathematical modeling and systems identification. Mathematical modeling uses the physical laws and properties to obtain the model. The systems identification approach uses the input-output data to obtain the model.

The system identification problem can be defined in different ways. If the system is totally unknown, black box approach is used. If the structure is known to some degree but some of the parameters are unknown, then gray box identification is used. The models used are classified as parametric models or non-parametric models. Transfer function is an example of a parametric model. Step response, impulse response, and artificial neural networks (ANN) are examples of non-parametric models.

1.2 Systems Identification

1.2.1 System Identification History

The origin of the theory of identification starts with introduction of the least squares method by Gauss in 1795 [1]. Since then, huge amount of research was done. The huge development in the modeling by Kalman was the main step of reaching the theory of the prediction modeling with the filtering and control based on the pole's placement. The theory was applied to obtain models for electrical and mechanical dynamic systems. Moreover, the wide application of the theory pushed further to use the same theory for all the applications where no physical modeling bases are available. Here, the system identification became important to start based on input data and noise or noise only to predict the output data of the dynamic system. Kalman and Ho [2] developed the state variable models based on the historical input and output data from the dynamic system experiments.

Astrom and Bohlin [3] developed the numerical system identification for the linear dynamic systems in 1965 when the error prediction identification started, which led to the ARMAX modeling structure, being available. At the beginning, there were two approaches: state space models or input and output data models based on the minimization of the error but after 1975, Lennart Ljung worked more on the software computation features which was successful to drive prediction error approaches [4]. Then, Gevers and Wahlberg led to the idea of considering the system identification as a designing problem after observing that all the found best approximated models were still have errors [5] [6].

1.2.2 System Identification Process

The system identification process is outlined in Figure 1.1. The first step is to design the experiment and collect data. This includes selection and generation of the input signal and sampling interval to be used. It also involves the assurance that the selected input is persistently exciting of sufficient order. It also includes the duration of the experiment, the data to be collected and weather the system being identified is operating in open loop case or closed loop case.

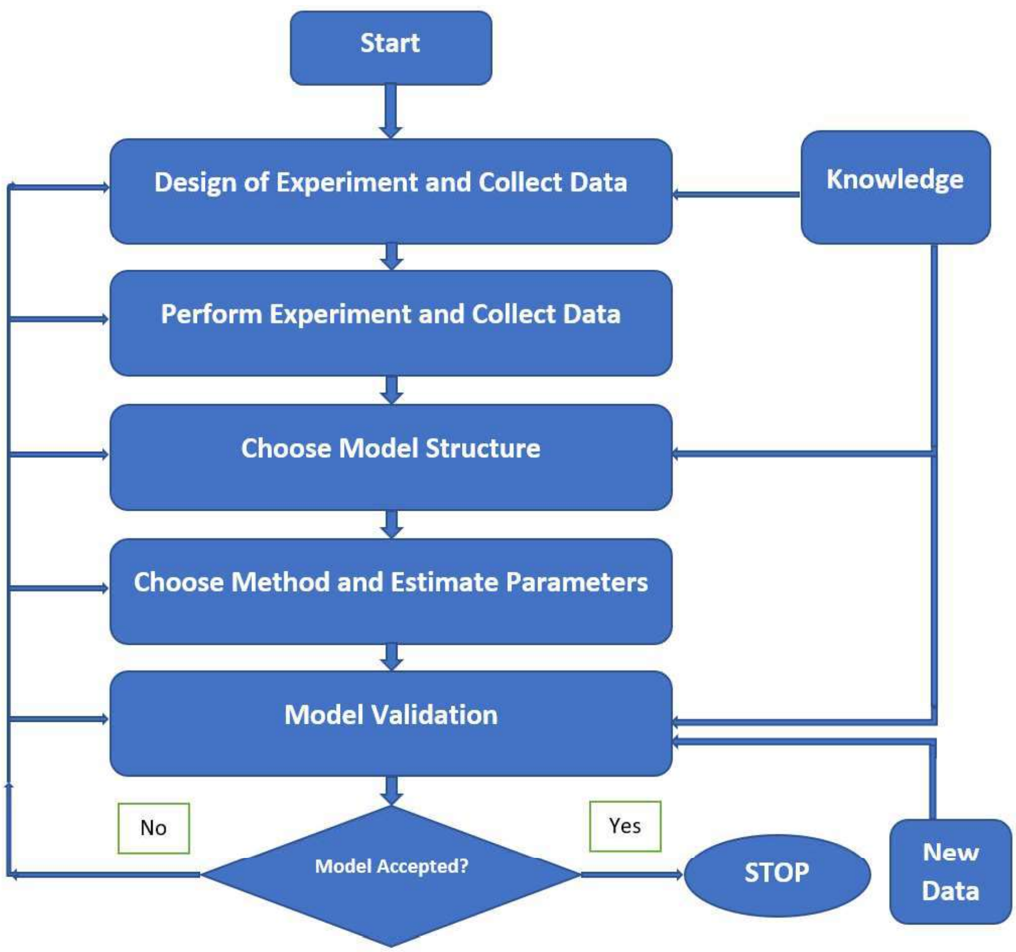


Figure 1.1: System Identification Procedure

Often it is required to get an initial idea about the main characteristics of the system such as presence of time delay and nonlinearities. A step signal, a ramp or sinusoidal inputs are applied. This will help in understanding the general behavior of the system. The system identification can drive accurate estimated model based on the persistently exciting input signal. The pseudo random sequence is commonly used as it has good energy content with wide scale of frequencies. The amplitude of the signal should be large enough to have a good ratio between the signal and noise and to overpass the friction problems. The amplitude might take the system into the non-linearity region. Normally, the input signal saturation gives upper amplitude bound where the mean values on several cases are non-zeros to eliminate or reduce the frication problem or to produce the linear model around the stationary point. The data is usually divided into two parts. The first part is used to conduct the identification procedure to get the model. The second part of the data is used for the validation tests [7].

The model structure can be parametric and non-parametric. Selecting the model structure is a critical step. The estimation quality is totally depending on the model structure. One needs a model that is flexible enough to represent the system but not too complex. For linear systems, the model dimension or the number of parameters involved can be done by trial and error, the objective is to have a simple model that provides good representation of the data. For nonlinear systems, the problem is much more complicated. Having more parameters in the model means more calculations are needed to obtain the model but this does not necessarily mean it is a better model. It is again a challenge to select the best suitable model structure to represent the actual system. Once the input-output data is available and the model structure is selected, the next step is to estimate the parameters of the model. There are many methods to estimate the model parameters. They include least squares, maximum likelihood, prediction error and others.

Once the model is obtained, the next step is to do model validation. The main purpose of model validation is to determine if the obtained model is acceptable or not. In model validation there are two main concerns: the flexibility of the model and its complexity. We need to have the model flexible enough to accurately represent the relationship between the input and the output, but not more complex than needed. The validation of the selected estimated model structures can be done by different ways and criteria. Commonly used approaches include simulation and residual analysis.

1.2.3 System Identification Theory

Theoretically, online system identification can be used for online modeling of the complex and noncomplex dynamic system. It is a mathematical technique driven based on the regression theory of the collected experimental output and input data from the plant or dynamic system experiment. The system identification methods are classified into traditional and artificial methods. The dynamic system identification can be considered to consist of two stages. The first step is to have the model structure and the second step is to have the estimated parameters for that model structure. The problem of system identification can be considered and designed as an optimization problem to find the estimated model parameters. And to reduce the predicted errors for all the parameters between the captured outputs from the system and the output of the obtained model. Mostly all the current system identification methods are built based on the mathematic calculations that gives more analytical form. Firstly, prior historical knowledge is required to determine the class of models that belong to the targeted dynamic system that may be used. If there is no prior knowledge, the research might be done by trial and error. The focus in system identification is the process and mathematical procedure of parameter identification.

The estimated models of the system identification are particularly important and required in many applications for many reasons some of which are listed below [4]:

- The accurate models can explain more about the behavior of the process.
- It is highly required to design better controllers.

- The estimated model could be one part of the designed controller such as the feed forward controllers and model reference controllers.
- The accurate estimated model is valuable for the operating conditions optimization and better controller performance.

1.2.4 System Identification Facts and Limitations

- The cost of the system identification model is still an important part of the control designing projects.
- The identification of the MIMO dynamic systems is still exceedingly difficult problem.
- Finding the optimal point in the case of the non-convex prediction error is a difficult problem.
- Handling the system identification process is still not friendly and not fancy.
- The search of the best model structure is key block in identification.
- Any model estimated based on historical data may not be precise.
- One record of the historical data may not be enough for precise estimation of the model.
- The input data type, and noise to signal ratio are particularly important to reach accurate estimated model [8].

1.3 Evolutionary Algorithms

1.3.1 Evolutionary Algorithms History

Darwin evolution theory was an inspiration to develop many approaches in different science fields such as the control engineering. The evolutionary algorithms were introduced in the 1960s based on the same theory as a process of population generations that evolves and optimize the best solution for the started problem [9]. The process simulated on complete repeated cycles of candidate selections that goes through steps of operations to end with a better new candidate solution and the cycle is repeated. This operation differs from algorithm to another with different properties, advantages, and disadvantages. The evolved solution is based on the fitness function. The candidate with better fitness is selected.

Evolution algorithms are used as optimization methods and several groups of evolutionary algorithms have been proposed. They include Genetic Algorithm, differential algorithms, evolution strategies, evolutionary programming, and others. Historically, Genetic Algorithm (GA) was introduced in 1962 by Holland as an adaptive reproduction plan of crossover as searching operator with mutation of small probability as background operator [10]. The selection is used as an operator depending on binary coding. Then, the created evolution strategies which applied in distributed mutation, modified the real value vectors and the recombination as main operators for the searching space. The selection operator and parameter strategy are deterministic with population generation size changing over time. The Evolutionary Programming is working with the extension of the parameter strategy and mutation while the selection operator is probabilistic on the search space [11].

1.3.2 Evolutionary Algorithms Applications

Evolutionary algorithms are used in many science fields including the social studies, economics, mathematics, biology control engineering and others [12], [13]. In addition, some of the applications and study areas are critical such as the system learning, aircraft designing, neural networks training, image analysis, DNA structure analysis, and selection.

1.3.3 Importance of Evolutionary Algorithms

Today in science fields and the industrial systems, millions of the multi objective optimization problems which are difficult to meet and so hard to be calculated or obtained analytically. This optimization solution is affecting the quality of the optimization application performance and all related functions. Moreover, some of the optimization problems might have variants that changing overtime and the optimal solution is required in relatively short time. More challenging optimization problems are those that have non-linear constraints to be satisfied along with the final solution. EA are open to problem parameters and operator configurations of numbers, mixed integers etc. Furthermore, the combination of problems and similar evolutionary approaches can be done easily to strengthen the global search for the optimal solution. The search needs a good initial populations list with diverse candidates that allow more searching areas of the global to be included during the optimization conversion process. The quality of the population leads to drive more than one solution and robust on the global search where they can be used for the multi objective problem. The old traditional methods can only reach one final solution. More advantages for some of the problems to find the boundary and range of the solution where can be possible of the wide global population search. In the worst-case evolutionary techniques can deliver the solution location and width where simpler approaches can be applied and designed to reach the optimal solution in a reasonable time. It can be considered as a tuning parameter technique.

1.3.4 Evolutionary Algorithms Process and Structure

After definition of the problem and the clear-targeted fitness function, the structure can be designed simply as below [14]:

1- Setting up the population of the first generation randomly from a wide range possible candidate.

2- Finding the fitness of each candidate using the defined fitness function.

3- Selecting a random pair of the population to prepare the parent selected for the crossover and mutation process.

4- Based on the probability of the crossover rate the parents crossover the pairs at chosen points for all the parents.

5- Based on the probability of the mutation rate the patents mute the pairs at each candidate chosen points.
6- After the previous process, the new population is made to replace the old one based on the best fitness scores.
7- Repeating the same cycle until the robust optimized solution is reached after many generations.
8- Terminate the cycle if no improvement on the global searching space after few generations.

1.4 Turbojet Engines

The turbojet engine system was invented before the Second World War. Turbojet engines are essential in modern life. It is particularly important to operate them safely and efficiently. A good model of the turbojet engines is needed to ensure good operation. Turbojets are air-breathing engines that consist of gas turbine and propelling nozzle. Physically, the cold air is entering into the compressor to increase the pressure and temperature within several stages. And during the compression the air will be mixed with the fuel resulting in a mixture of combustible gas. The combustion is increasing the pressure and temperature to the highest level to generate the required force that move the blades and rotate the turbine shaft.

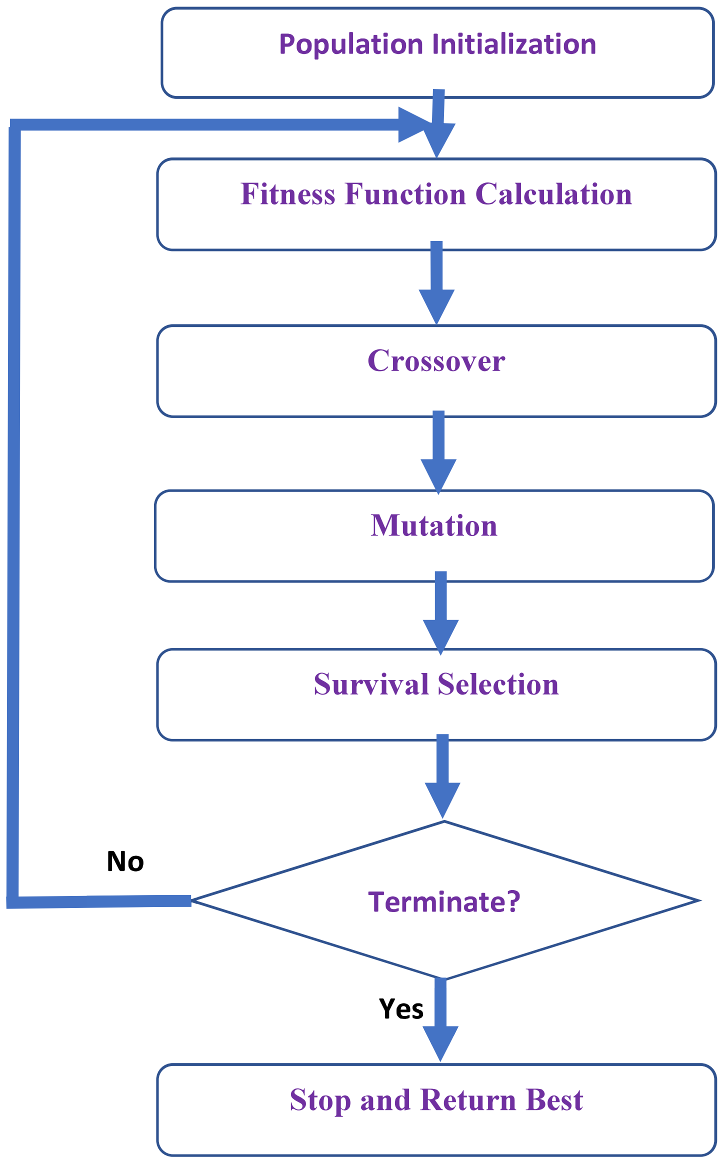


Figure 1.2: Evolutionary Algorithm Process Structure

The exhausted gas is leaving the engine through a nozzle with exceedingly high speed that produce huge amount of thrust to power the aircraft. The turbojets are noisy and not efficient for the low-speed vehicles. The first gas turbine was developed in 1921 by Maxime Guillaume. In 1932 Whittle developed the design into a two-stage axial compressor that feeding a sided single centrifugal compressor. Earlier the German turbojets removed some limitations of the operation hours due to the suitable combustible materials. Rolls-Royce Welland Britch engine introduced better combustible material to reach 500 operation hours. The structure of a gas turbine jet engine is shown in Figure 1.3. Later, General Electric modified the engine further to have more run hours and efficient fuel material during the Second World War in U.S.A [15].

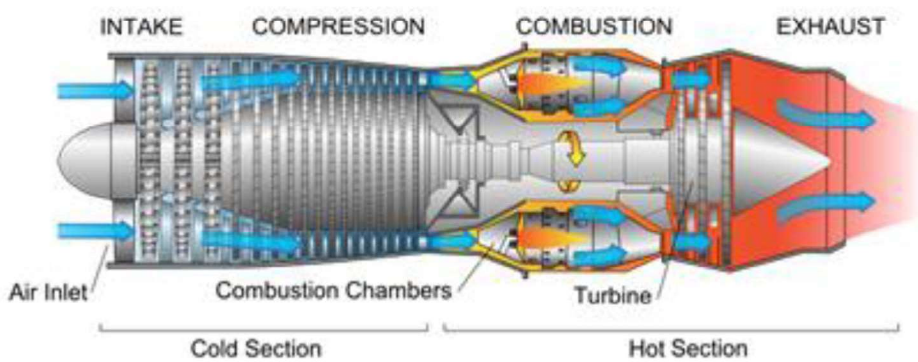


Figure 1.3: Gas Turbine Jet Engine

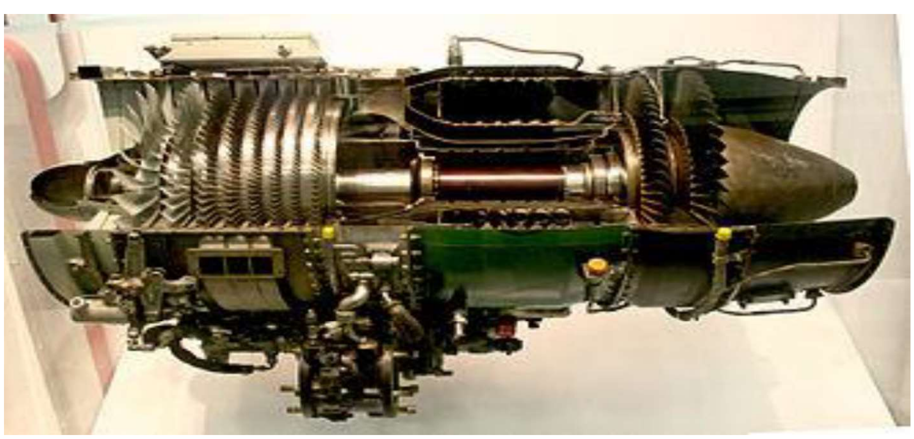


Figure 1.4: General Electric Turbojet Engine (Model J85-GE-17A).

The turbojet dynamic system starts of the inlet air with compressor, then the fuel injection with turbine and the nozzles exhausts of the hot gases. The turbojet engines have more power efficiency than the piston engines and they can accept high temperature operation and produce more thrust [16]. The mechanical design of the turbojet engines needs many things to be satisfied including the requirement of the high compressor operation range and high efficiency. The turbojet engines involve high pressures and temperature. Optimization methods are used to get better engine structural design and avoiding failures. Besides, the turbojet engine is a nonlinear system that requires accurate model structure and identification to improve the stability and reliability of the engine. The engine is switching on different operation modes for different performance requirements which means engine will have different models. In such cases, the turbojet engine is considered as a multiple model system [17].

Due to the physical complexity of the dynamic interacted systems on the turbojet engine, it was one of the interesting areas for manufacturers and researchers in terms of proper entire system modeling and control. The physics laws are so long and required a deep focused background on the physical and mechanical science to just develop mathematical model representing the system and might not be exactly proper during the practical applications in aerospace, marines, industrial process, and vehicles. The wide application users concern the best control of the engine during the operation and one of the main effecting factors is the accurate mathematical model. Today, system identification theories and methods can estimate variety mathematical models based on the regression concepts of the dynamic system historical data in easy steps. The data regression techniques are developed into many standard algorithms such the least squares method, instrumental variable method and furthermore to intelligent artificial techniques such as the neural network methods. The modelling and identification process become efficient for more accurate modeling in shorter time and less knowledge of the dynamic system physics.

1.5 Problem Statement and Thesis Motivation

1.5.1 Problem Statement

This thesis focuses on the system identification of nonlinear dynamic systems using the parametric black box method modelling and as extensive focus to search on the suitable nonlinear model structures selection as well as determining the system's parameters using evolutionary algorithms and if possible, to work further on the non-parametric system identification such as the neural network methods for more comparisons study. Moreover, the evolutionary algorithm will be applied to identify parameters for Hammerstein model and other nonlinear system.

1.5.2 Thesis Motivation

In this thesis the focus will be on the use of the evolutionary algorithms to select the best model structure for system identification. Evolutionary algorithms have been used to solve complex optimization problems in many different fields. The evolutionary algorithms are expected to perform better than conventional system identification techniques. The thesis objectives include comparing the classical parametric system identification application with the system identification when the evolutionary algorithms are used. In addition, the best system identification using the evolutionary algorithms with some criterion restrictions are introduced. The artificial nonparametric neural network methods will be applied and compared with the proposed algorithms. The evolutionary computations are highly promising integrated approaches, because of the simple little required knowledgeable information on the system. They are viewed as a future alternative to the system identification methods. Moreover, it simply can be combined with other techniques to form intelligent learning approaches. The online identification and the Evolutionary Algorithms are both considered as learning integrated artificial techniques. The artificial techniques offer more flexibility over the traditional techniques.

Moreover, interesting observations are given below:

- Limitation of the system identifications developments in the procedural learning steps and fixed model structures. It is highly analytical and based on mathematical derivations.
- Combining the smart genetic algorithms with the recursive online identification algorithms to support and develop the control system solutions.
- Evolutionary techniques can be used on the model structure investigation and parameter optimization.

1.6 Thesis Objectives and Proposed Goals

1. Applying system identification using evolutionary algorithms approaches and compare performance results with classical methods.
2. Developing an evaluation and test part for the evolutionary algorithm to be used in obtaining the model.
3. Applying non-parametric artificial neural networks methods to identify a model from the data and compare the result with that obtained using the evolutionary algorithms.
4. Apply evolutionary algorithm in identifying nonlinear system described by Hammerstein model.

1.7 Outlines of Thesis

The remaining parts of the thesis are outlined here. In Chapter 2, the literature review gives an overview of the evolutionary algorithms and their applications in systems identification. Identifications of jet engines and other applications are also considered. In Chapter 3, the mathematical methods, problem description and analysis will be defined. In Chapter 4, the proposed algorithm and methodology will be discussed. In Chapter 5, simulations and results found are compared and analyzed for each proposed method. Finally, in Chapter 6 the summary and conclusions of the thesis will be given.

CHAPTER 2

LITERATURE REVIEW

2.1 Literature Overview

In this chapter, we provide literature review on the evolutionary algorithms and their uses in system identification. Section 2.2 covers evolutionary algorithms and applications. System identification and applications are covered in 2.3. In Section 2.4, Jet Engine modeling is covered. The evolutionary algorithms for system identification are discussed in Section 2.5.

2.2 Evolutionary Algorithm and Applications

An evolutionary algorithm defines a class of statistical optimization processes that imitates the natural evolution process. Evolutionary algorithms originated in the early 1960s, and since then, several evolutionary processes have been proposed [12]. These are mostly genetic algorithms, strategic evolutionary methods, and evolutionary programming. All these methods function on a class of prospect solutions. This class is usually modified by the fundamental evolutionary principles using powerful simplifications. The necessary evolution principles are selection and evolution.

The two principles of evolution algorithm, selection, and evolution, anchor on different aspects of organisms. Selection is the way living organisms compete for resources. Some microorganisms have more competitive tactics than others and have better survival chances than others. They survive and reproduce, and this leads to the continuation of their genetics. A stochastic methodology is used in the evolutionary algorithms processes to simulate natural selection [18]. The process allows every solution to reproduce a specific number of times, depending on its quality. Typically, a scalar fitness function is calculated for evaluating the individual's quality [19]. The second principle in evolution refers to the imitation of the natural capability to create new living organisms by mutation and recombination. Evolutionary algorithms have been a powerful and robust search operation. Evolutionary algorithms are suited to optimizing multi-objective solutions because they can capture many Pareto-optimal results in one simulation run.

They also exploit the resemblance of resolutions by the process of recombination. Evolutionary algorithm application in optimization is getting a lot of interest from various scientific and engineering disciplines worldwide. Several different evolutionary algorithms are applied differently depending on the different optimization methodologies. They include genetic algorithms, differential algorithms, evolution strategies, and evolutionary programming.

The genetic algorithms are applied in civil engineering, operations research, management science, electronics, and electrical engineering. For instance, a GA-based optimization model has been developed to reduce energy consumption in gas purification plants. Besides, in this technique, solution space search simulates the natural process in the environment, and Darwin's species evolution theory is contemplated [20]. In a genetic algorithm, the principle of evolution is applied in the selection. The genetic algorithm ensures that the fittest individual is selected to produce the next generation. The evolution strategy is also applied in the selection principle though it is different from the genetic algorithm because its procedure is different [12]. In the genetic evolution, the new generation is made from the parent population by selecting individuals based on their fitness, and the population size is kept constant. In the evolution strategy, the approach is to create a temporary individual, and its size is different from the original population. The beings in the transient population are taken through mutations. Differential evolution algorithm is applied in heating, cooling, and in energy storage systems. Differential algorithm is applied in a multi-objective algorithm used to aid in analyzing the probability of the loss of power supply and the cost of electricity [21]. Evolutionary programming is applied in computer science, software engineering, civil engineering, transportation science technology, building construction technology. For instance, Yan et al [22], proposed a self-adaptive bi-subgroup algorithm that aids in solving the hybrid electric vehicle's Pareto optimal solution. The algorithm has been used to determine the most suitable degree of hybrid for the hybrid electric vehicle. Hongjun Zhang et al [23], focused on the chaotic system estimation that is a critical issue in the nonlinear research science. The parametric estimation is designed as varying model's optimization problem. The search and investigation of the solution is based on the learning optimization of the initial population and generations evolution. The obtained parameters of the chaotic system are tested to make sure it is a local optimal of the defined domain. The integrated simplex algorithm with the main one to be used as a tool to support the global optimal solution investigation. The searching process over all reflections and contractions, the results prove the best solution over the chaotic identification and PSO optimization. West et al [24], worked on higher degree of hydraulic manipulator modeling and the estimation of the model parameters. The model development simulated in special toolbox in MATLAB within the not found parameters. The objective of the work focused on developing the methodology to find the parameters feasibility for the consistent inputs, outputs, and the states of the unknown environment of the system operation and under the noise. A genetic algorithm is derived to resolve the error in the output into the required accepted level of accuracy for the manipulator joint. Olteanu Marius et al [25], worked on the same basis of the modeling process which was going through many iterations of calculations and estimations to reduce the error for a real system. Similarly, the genetic algorithm is an optimizing tool that is used here for the modeling evolution to find the proper parameters of the industrial process. Daniel R.

Ojeda G. et al [26], worked on a methodology to find the estimated model of the thermal apparatus for the thermos-electrical forms. The particle swarm optimization was combined with the Monte Carlo mechanism to estimate the parameters of the model. The advantages of the approach were to drive the feedback controllers that is designed in the Gaussian quadratic linear controller which is good for the uncertainty of the dynamic systems. The simulations and performance of the estimated parameters convergence proves the effectiveness of the results.

2.3 System Identifications and Applications

System identification is used in developing mathematical models. These models are implemented to enhance physical understanding of systems, analyzing the properties of a system, simulations, state estimation, filtering, prediction, monitoring, and diagnosing faults. Historically, system identification originated by Gauss and his least-squares methods that was used to build models that describe heavenly objects' movement. The methods were not implemented by Gauss. Such methods were not possible until the emergence of technological computers and the extension availability of measurement hardware and instrumentation in the 20th century. The rising requirement in the recent decades for high performance and reliable systems has further enhanced the need for valid empirical models and, therefore, a direct need for system identification techniques. It is essential to apply system identification methodologies even when the availability of a model structure from the physical principles is high. It is important to use system identification in the estimation of parameter values. These schemes make system identification an essential tool in many engineering applications, from the suppression of vibration to process control [27]. The system identification technology is widely used in many engineering fields, but it is also relevant in other areas like economics, medicine, ecology, biology, and geology. The general purpose of system identification is to provide an interface that meets the model user's needs. The system identification consists of a set of activities, which include a selection of model structure, experiment design, parameter estimation, and validation of models. Estimation of parameters is the determination of the parameter values of the model at a particular accuracy level. There are many algorithms used in the parameter estimation process. One can configure the algorithms by configuring the minimization criterion to focus the parameter estimation on the required range of frequency. To configure the estimation algorithm, one can also specify the options of optimization for the repetitive algorithm. This includes options like the maximum and a minimum number of loops [28]. Experiment design is applied when it is possible to produce a rich data set and provide accurate information. This technique uses elementary models. It is dependent on the approach of the frequency domain. In this model, one parameter is not known, and the system is used to produce an informative data collection. The selection of model structure is the selection of a proper form of a model for a particular purpose. This is due to the high number of possibilities. This is often a challenging problem. The choice is based on prior knowledge about the structure of the procedure. It can also be influenced by the data analysis process of the data itself. It is essential to put into consideration the order of the model and the input delay. Model validation is the process of confirming whether the model that has been identified suits the intended purpose. System identification is applied in approving the created models. The process involves running several essential tests on the developed model against the intended purpose. System identification will use input signals and the output data to estimate the model structure. The process will tell whether the model is good enough to be used for its intended function. Obtaining a correct model is dependent on how well the measured data reflects the system's behavior. Abdulla Ayyad et al [29], proposed an approach for the parametric identification based on the deep learning with a modified feedback relay. The approach requirements are to provide steady state feedback cycle and ensuring the stability of the identification process. The approach outputs are based on the deep learning model that presents the identification of the underlying parameters in less time. The SoftMax function is updated to confirm more the deep learning model process. The quadrotor unmanned vehicle application performance is used in the proposed approach which gives efficient results and accuracy. Asma Atitallah et al [30], worked on the Wiener time-delay system estimation that shows many difficulties. However, the rounding property of the hierarchical approach and the auxiliary model approach are recommended to solve the mentioned difficulties. The cost function is minimized by the conjugate algorithm with their start parameters. Simulation of the proposed approaches was presented. Mohammad Jahani et al [31], worked on the Hammerstein systems with fractional order and time delay. The authors proposed a recursive identification based on Genetic Algorithm. The model consists of two parts, the first one is the radial basis function neural network, and it is nonlinear and the second one is linear. The genetic algorithm is used to identify the fraction orders. The parameters are updated by the recursive least squares. The proposed method is applied and simulated with good and accurate results. Dongmin Yu et al [32], worked on nonparametric system identification for the fuel cell proton exchange to increase the efficiency of the design. An improved neural network is used in the proposed optimization algorithm which is a combination of the fluid search and the world cup optimization algorithms. The proposed approach is applied for the efficiency improvement of the parameter's identification of the fuel cell proton. The approach is tested in four different operational situations and compared with the other common algorithms. The simulation graphs prove that the approach is providing better accuracy of the selected model. Timothy Sand et al [33], developed an approach for the system identification using the nonlinear adaptive controller. The used methodology is applied on DC motor controlled by the self-tuning regulator. Normally the online system identification can be used for estimating the time-varying plant parameters. The approach is designed to perform the self-tuning regulator by the indirect way where both the feedback and feedforward controls are obtained based on the estimated parameters of the mathematical model of the motor. The parameters are adapted and quickly converged as usual into the right model. Moreover, the same approach is modified with the direct self-tuning regulator to investigate the challenging non-minimum phase system. Result shows that this approach leads to better controllers.

2.4 Jet Engine Modeling

Irina Carmen Andrei et al [34], worked on finding the undetermined engine parameters that affect the thrust of the engine. The method is based on the convergence of a new nonlinear equation of the temperature variables. Another method was developed based on the fan and temperature variables to find the pressure of the fan ratio and the rate of the airflow. The study used two cases approach where completed the mathematical models and the missing parameters and calculated all other rational operational variables on the off-engine design mode. Ladislav Nyulászi et al [35], worked on several traditional methods and nontraditional model identification based on the turbojet engine experimental data. The output of the obtained models is compared with the output of the measured experimental data with the mean error calculations for the best accuracy model. Károly Beneda et al [36], worked on linear state space representation and finding the coefficients for the turbojet engine. The experimental equipment used in lab for the history and operations measurements for the purpose of finding the basis of the quadratic control system using the mathematical model based thermodynamic principle. The results show extension opportunity of the work to be used on the education or industrial sector. It is also supporting the reliability of the plant controllers for the diagnostic studies. K´aroly Beneda et al [37], worked on the state space representation of the micro-jet engine based on the physics principles of the plant. The reliability improvement is considered as the main advantage of the integrated control system. The mathematical model is developed to describe the main factors related to the engine. The obtained model is proposed to be used for the control system after the identification of the micro turbojet with the fixed exhaust nozzle. Evgeny Filinov et al [38], worked on many proposed correlated models based on weight calculations of the small aircraft engine. The feasibility study of the applications is found after the comparison and analysis of the different models results. The accuracy of the estimated weights is different from a model to another model which differs on the numbers of the inputs as well. The main used data is obtained from the turbofan engines and thrust that is below 50 KN and based on the collections, the formula is created to find the weight of the design engine. Khaoula Derbel et al [39], worked on the state space representation to develop the linearized parametric model from the nonlinear performance for the turbojet engine where the thrust is the ratio of the turbofan power. The method was introduced recently by

Rolls Royce engines are also proven in the single turbojet power plant. The second part is to perform the parameters identification based on the engine operations measurements. Finally, the model is tested by the simulations for the auto control mode for the engine and found it to be suitable. Qianjing Chen et al [40], worked on the development of the simple model for the turbofan engine that differs from the nonlinear model which is commonly used in the airplane engines. The purpose is to reduce the long-time calculations required over the iterations. The outputs of the linear model where the solution equals the nonlinear level component model solutions are found at the engine operation points. And combined to form variable state space model. Additionally, its improved further to the complete envelope of the flights and integrated in a calculation form to get satisfactory behavior and assure the accuracy. The simulations comparison shows that the obtained combined model requires less resources during the integration and computation process and maintains the accuracy.

2.5 Evolutionary Algorithms for System Identification Approaches

Many recent papers of different applications and methods strategy applied in system identification and evolutionary algorithms were reviewed. Ecaterina Vladu [41], worked on system identification problem to be formulated as an optimization problem to find the model and related parameters with the minimum predicated error between the measured plant output, and the model output. The system identification methods are highly analytical and based on mathematical derivations of the system model. As an alternative and supportive method, the evolutionary computation seems easier to be combined with different technique of control engineering, machine learning and artificial intelligent methods. The main contribution of this paper is to consider the evolutionary approaches for system identification to prove the performance achievement based on case studies. Wei-Der Chang [42], worked on a real coded genetic algorithm (GA) that was applied on the system identification and the control of the nonlinear system classification. It is known that the genetic algorithm (GA) is globally used as optimization method. In solving the optimization problem there are two kinds of the genetic algorithm as binary and real coded and the last one is more useful in the engineering applications. The main contribution of this paper is to utilize the GA to identify the unknown dynamic system. In addition, an optimal offline PID controller is tuned using the GA, which shows the effectiveness of the method. Songtao Xue et al [43], worked on Particle Swarm Optimization (PSO) as a new method that has yielded to solve the most complex optimization problems. The advantage of the method is that it uses a simple and useful way to check the quality of the solution. The main contribution of this paper is to use the PSO algorithm to identify and estimate the unknown parameters of the system by formulating the problem as a numerical optimization problem with many dimensions. The simulation results are identifying multi-degree of freedom (DOF) linear and non-linear structure to present the effectiveness of the introduced method. Habibinejad et al [44], worked on identification and predicting API-X70 steel to expand the understanding the fundamental of the material quality. The problem is on predicting the behavior of the material which depends on huge number of variables that made it more complex and increased the errors. The problem is formulated to be as an identification problem to find the model and related parameters with the minimum possible error between the measured data, plant output and the model output. The Genetic Algorithm (GA) is used to find the structures of the model based on the history-measured output and input data. The main contribution of this paper is to use the GA method to predict the gradation of API-X70 steel mathematical models, which describe the behavior of the plant with low error. Ozden et al [45], worked on investigation of the development of the system identification using the Artificial Bee Colony (ABC) algorithm. The system identification problem can be formulated to be an optimization problem to find the dynamic model with the related parameters that shows the minimum predicated error between measured plant output and the model output. The traditional common system identification techniques such as the recursive least squares method and autoregressive exogenous method are analytically and based on mathematical deviation of the dynamic model. The Evolutionary computation approaches are used as alternative to methods to show how the ABC algorithm can be combined with system identification, which is used to estimate the unknown parameters of the system. The main contribution of this paper as presented is that the linear system identification has good performance. Moreover, the used identification of the motor shows the performance of the ABC algorithm, was successfully matched the actual plant outputs. Thanh N. et al [46], worked on identifying physical parameters of a large structural system, which gives the challenge of dealing with many unknown parameters. However, a divide and conquer approach is required to simplify the problem into many sub-structures for each one less unknown parameter are to be identified. This leads to more accurate and efficient identification. Moreover, the ill condition and the nature of the inverse analysis are highly beneficial based on non-gradient search methods such as the genetic algorithm (GA). The main contribution of this paper is the use of the GA algorithm based on substructure identification strategy for large structural systems. This strategy is classified into two significant improvements: the use of the acceleration measurements to directly account for interaction between substructures without approximation of interface force and the other participation on the use of an improved identification method based on the features of the GA algorithm. Tavakolpour-Saleh et al [47], worked on turbojet engine input output history data that was recorded by experiment and used for model investigation through the black box identification methods. The focus on deep investigation to obtain a proper nonlinear model structure for the parametric one. Moreover, Neural Network method was applied with configuration to obtain the non-parametric model. Finally, the validity test carried out based on the output selected data and presented the comparison between the two approaches. Andre Felipe et al [48], worked on Particle Swarm Optimization method that introducing new structure and combined with traditional identification method for better convergence results and analysis is based on criterion that can be applied for different systems. Final improvements expanded for high frequency system representations of linear and nonlinear dynamic systems. Partha S. Pal et al [49], worked on the colliding bodies optimization (CBO) approach using the Hammerstein models for the linear and nonlinear process identification of fourth order polynomial based on the minimum mean square error. And tested the optimized output which shows close estimation to the output data. Finally, the precision and accuracy of results confirmed the effectiveness of the CBO application for the system identification problems. The least error and convergence time with results are better compared to the stochastic approaches reviewed. Loris Vincenzi et al [50], worked on different evolution algorithms which were combined with the surrogate model to identify the mechanical parameters of a beam for the minimum global objective function. The Differential Evolution (DE) algorithm is applied with different vectors of parameters and weighted population for random selection and conversion after the mutation operation. The results show good improvement of the performance. Multiple searching parameters are used on the same time. Finally, it introduced as effective approach called DE-Q.

Hong Peng et al [51], worked on artificial bee colony and combined with P tissue system for the infinite impulse dynamic system identification. The P system structure is considered as the computation part. The ABC is improved to include evolution criterion for the structure and sequence mechanisms. The P system designed is intended to identify the infinite impulse system with the optimized filter coefficients. The combined new approach ABC-PS was compared with the normal artificial bee colony and other methods which shows the superior efficiency of the new introduced modified approach. Jinyao Yan et al [52], worked on three model strategies with nonlinear model structure evolution for dynamic system NARX identification. The introduced system models are linear in parameter but not linear in operation signal. The estimation of the parameters is using a set of theoretical analysis data for possible sets of solutions. And the set of parameter solutions are passing through the nonlinear functions for survival compete chance. The best fitness of the function represents the system. Moreover, the nonlinear structure can be identified and estimated in the existence of different noise signals. Finally, the simulation was reviewed with the performance advantages over the traditional identification methods. Yuhao Huang et al [53], worked on literature review that shows the importance and the need of the system identification using the evolutionary algorithm especially in the automatic model structure selection. The used evolutionary algorithm is genetic programming. They were applied for cleaner energy system optimization. The input data is recorded based on the minimum calculated errors. And the validation data are collected from experiment for the model robustness judgement. The study is based on two- and three-dimension parametric procedures. Finally, the optimized pareto is reached with optimal operation condition. Faizal Hafiz et al [54], worked on particle swarm optimization combined with NARX models for the selection of four polynomial structures and identification. The learning method is depending on two-dimension UPSO. The proposed approach is tested by theoretical criterion and compared with the normal genetic algorithm which confirmed the advance performance over classic methods. And different nonlinear systems were used to prove the accuracy and capability of NARX structure selection and robustness results found under the effect of different noise level as well. K. Worden et al [55], worked on a selected challenging nonlinear system identification problem. And, focused on the developed methods for solutions. It is concluded that many general frameworks of different problems range can be addressed for the evolutionary algorithm optimization. The approach was developed on previous work and extended on this paper with a greater number of variants where some of them are used for the first time in system identification. Finally, the authors provided a discussion with many suggestions on the best practices and opportunities. Laura S. de Assis et al [56], worked on genetic algorithm as robust evolutionary method for the nonlinear Volterra series system identification. The obtained parameters were evaluated over different levels of noise signal for many nonlinear dynamic systems were reviewed on the literature study. Helon Vicente et al [57], worked on the nonlinear system identification using the coevolutionary algorithms and the artificial neural network. The NARX identification method was applied, and it gave the most accurate result. It has advantages when applied over the coevolutionary algorithms on the global optimization search.

2.6 Summary

Based on the literature review we found many published papers and research on the evolutionary algorithms for system identification applications. Approaches were developed in different methods with good improvements but not much have been seen on the nonlinear regressions searches around the possible models that might be representing the system more accurately. This will lead to improving the performance. For the turbojet engines, this can improve the stability of operations and allow application of advanced control theories.

CHAPTER 3

MATHEMATICAL ANALYSIS

3.1 Introduction

In this chapter, the mathematical equations, key definitions and concepts and the main algorithm tools are introduced in detail.

3.2 Mathematical Preliminaries and Definitions

3.2.1 Model Structures

A system can be represented by the block diagram such as the one shown in Figure 3.1. The process represents the relationship between the input and the output. A single-input-single-output discrete-time system can be modeled as

$$\mathrm{y(t)} + \mathrm{a}_1\mathrm{y(t-1)} + \cdots + \mathrm{a}_{\mathrm{na}}\mathrm{y(t-na)} = \mathrm{b}_1\mathrm{u(t-1)} + \cdots + \mathrm{b}_{\mathrm{nb}}\mathrm{u(t-nb)} \qquad (3.1)$$

A alternative way to represent the model is given by

$$A(q^{-1})\mathrm{y(t)} = B(q^{-1})\mathrm{u(t)} \qquad (3.2)$$

where

$$A(q^{-1}) = 1 + \mathrm{a}_1 q^{-1} + \mathrm{a}_2 q^{-2} + \cdots + \mathrm{a}_{\mathrm{na}} q^{-na} \qquad (3.3)$$

$$B(q^{-1}) = \mathrm{b}_1 q^{-1} + \mathrm{b}_2 q^{-2} + \cdots + \mathrm{b}_{\mathrm{nb}} q^{-nb} \qquad (3.4)$$

and q^{-1} is known as the backward shift operator. The parameter vector is defined as

$$\theta = [\, \mathrm{a}_1 \;\; \mathrm{a}_2 \;\; \ldots \;\; \mathrm{a}_{\mathrm{na}} \;\; \mathrm{b}_1 \;\; \mathrm{b}_2 \;\; \ldots \;\; \mathrm{b}_{\mathrm{nb}} \,]^T \qquad (3.5)$$

Figure 3.1: Basic Process Block Diagram

The disturbance which is unwanted signal that may enter the system in different ways and affects the response of the system. One way to represent a system with output disturbance is shown in Figure 3.2. In such case the disturbance affects the output directly. The output y(t) can be descriped by the following model.

$$y(t) = G(q^{-1})u(t) + H(q^{-1})e(t) \tag{3.6}$$

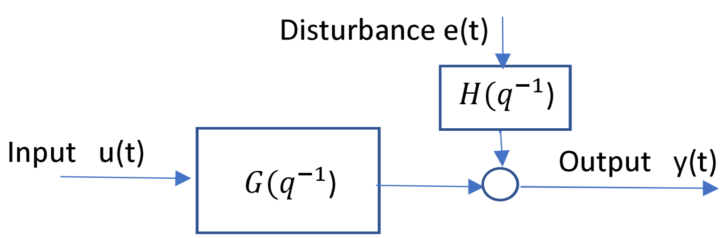


Figure 3.2: System Block Diagram with Disturbance

Representing the relationship between the inputs and the outputs of the system can be done in different ways. Different models structures are possible. A simple system can be described by the classical proportional model

$$y(t) = K\,u(t) \tag{3.7}$$

where $u(t)$ and $y(t)$ are the input and the output of the system. Another structure, the auto-regressive (AR) model is represented by

$$A(q^{-1})\, y(t) = e(t) \tag{3.8}$$

where e(t) is noise. The Auto-Regressive Exogenous model (ARX/NARX) is given by.

$$A(q^{-1})\, y(t) = B(q^{-1})\, u(t) + e(t) \tag{3.9}$$

Other structures include the Auto-Regressive Moving Average Exogenous (ARMAX) model

$$A(q^{-1})\, y(t) = B(q^{-1})\, u(t) + C(q^{-1})\, e(t) \tag{3.10}$$

The Output-Error (OE) model

$$y(t) = \frac{B(q^{-1})}{F(q^{-1})} u(t) + e(t) \tag{3.11}$$

Box-Jenkins (BJ) model

$$y(t) = \frac{B(q^{-1})}{F(q^{-1})} u(t) + \frac{C(q^{-1})}{D(q^{-1})} e(t) \tag{3.12}$$

The model structure is determined by the way different variables are related and by the number of the parameter involved. The parameter vector θ will include all the parameters involved.

Nonlinear models are much more complicated. There are infinite ways to relate the inputs and the outputs of system. The model structure can be selected for the linear and nonlinear dynamic systems based on specific criterion. One commonly used way to characterize the modeling error is the mean square error (MSE) [58] defined as

$$\text{MSE} = \frac{\sum_{k=1}^{N}\left(y(k) - \hat{y}\,(K)\right)^2}{N} \tag{3.13}$$

The MSE is obtained as the sum of the squares of the difference between the measured output $y(k)$ and the predicted output $\hat{y}(k)$ divided by the number of samples N.

In Evolutionary algorithms a fitness function is used to judge the quality of a solution. Typically, the fitness function is giving a positive value indicating the better solution. When Evolutionary algorithms are used in identification, a commonly used fitness function is given by

$$Fitness = 100\left(1 - \frac{\|y - \hat{y}\|}{\|y - \bar{y}\|}\right) \tag{3.14}$$

The fitness function depends on the measured output y(t) and the estimated outputs $\hat{y}$ (t) and the average of the output $\bar{y}$.

3.2.2 System Identification Definition and Methods

The mathematical modeling of the complex dynamic system is an interesting complex challenge. Systems identification can be used to obtain approximate representation of the real system. The system identification is a mathematical technique that generates mathematical models based on the collected experimental output/input data from the plant.

The system identification process consists of four main steps: measurements, estimation, validation, and representation. Each of these steps may involve different steps. In the measurement step input-output data is collected. This may involve design of experiment, selection of input signals used, selection of size of data to be collected. The estimation step involved fitting the data to specified model structure. The validation process involved performing tests to make sure the selected model is a good one. The last step is related to selection of appropriate way to represent the model.

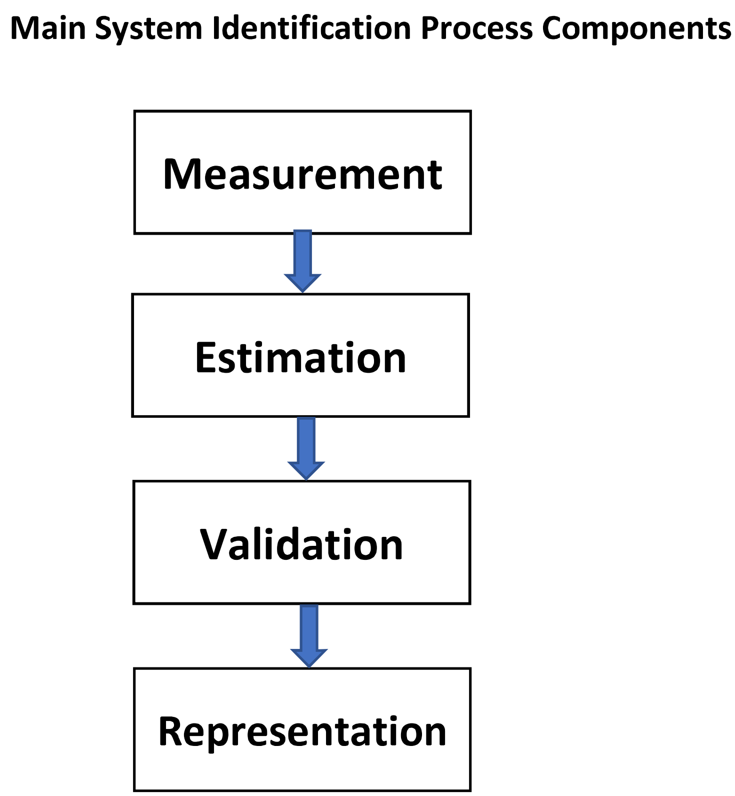


Figure 3.3: System Identification Process Block Diagram

3.2.3 Parametric System Identification

The parametric system identification problem involves determining the parameters of the model. This can be done in different ways. When all the input-output data is available offline techniques can be used. Many offline techniques are available including the Least Squares method. Online techniques in which the parameters are computed as the data is being available. The model parameters are updated in current time based on the last data observed. One such method that will be used in this work is the recursive least squares (RLS) method. In the recursive least squares method, the identification of the parameters is computed recursively over the interval of time. It can be applied in online operation system identification problems. The model of the recursive least squares identification.

$$y(k) = \varphi(k)\,\theta + \varepsilon(k)$$

where $\varphi(k)$ is the regressor vector which depends on the model structure, θ is the parameter vector and $\varepsilon(k)$ is noise. If all the input-output data is available, we can define

$$Y = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{bmatrix}, \quad \Phi = \begin{bmatrix} \varphi(1) \\ \varphi(2) \\ \vdots \\ \varphi(N) \end{bmatrix}, \quad \varepsilon = \begin{bmatrix} \varepsilon(1) \\ \varepsilon(2) \\ \vdots \\ \varepsilon(N) \end{bmatrix}, \quad \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix},$$

The model can be expressed as

$$\boldsymbol{Y} = \boldsymbol{\Phi}\ \boldsymbol{\theta} + \boldsymbol{\varepsilon} \qquad (3.14)$$

The least squares estimate is given by

$$\hat{\boldsymbol{\theta}} = (\boldsymbol{\Phi}^{T}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^{T}\boldsymbol{Y} \tag{3.15}$$

For recursive least squares, we need an initial guess of the parameter vector $\hat{\theta}(0)$ and the estimate of the parameters is updated as

$$\hat{\theta}(n) = \hat{\theta}(n-1) + \mathbf{K}(n)\varepsilon(n) \tag{3.16}$$

where

$$\mathbf{P}^{-1}(n) = \Phi(n)^{T}\Phi(n) \tag{3.17}$$

$$\begin{aligned} \mathbf{P}(n) &= (\mathbf{P}^{-1}(n-1) + \phi(n)\phi^{T}(n))^{-1} \\ \mathbf{K}(n) &= \mathbf{P}(n)\phi(n) \end{aligned} \tag{3.19}$$

$$\varepsilon(n) = y(n) - \phi^{T}(n)\hat{\theta}(n-1) \tag{3.20}$$

3.2.4 Nonparametric System Identification

One of the most important and widely used method of the nonparametric system identification is the artificial neural network algorithm (ANN). The artificial neural network is inspired from the biological neural networks that constitute the animal brains. ANN is a multi-connection weighed branches for unlimited nodes like the brain representation. And each node forms a neuron that feeds another branch as inputs. The typical neural network in applications consists of three layers called the input layer, hidden layer, and output layer. These layers require many training trials with different neuron numbers and weighted network sizes till reaching reasonable results. The more learning time and data, the better results obtained which reflects the same way of the brain learning operation. Today ANN is used for wide interesting digital applications including dynamic system identifications.

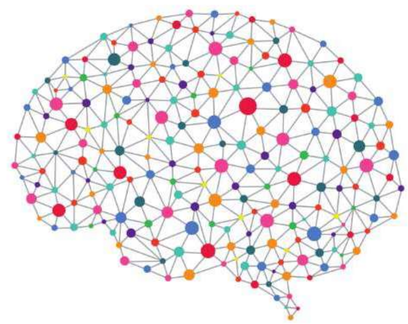


Figure 3.4: Deep Learning Neural Network

Here, we selected the NARX neural network model of the wide forms and sizes to be trained on different trials and algorithms with the historical data that are divided into 70% for training, 15% for validation and 15% for testing. The first Levenberg-Marquardt algorithm requires more memory with less training time, and it is stopping automatically when there is no improvement in the mean square error. The second Bayesian Regularization algorithm consumes more training time but can absorb more generalization and noisy historical training data. The training time is stopped automatically when the adaptive weight is minimized. The third Scaled Conjugate Gradient algorithm requires less memory and stopping when there is no improving on the mean squared error on the validation period.

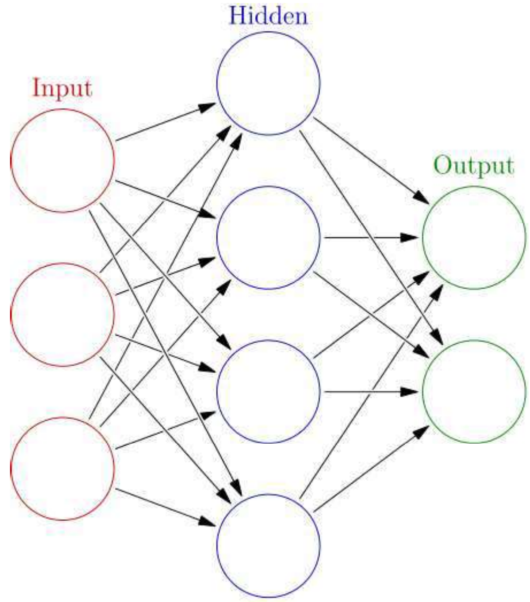


Figure 3.5: Artificial Neural Network

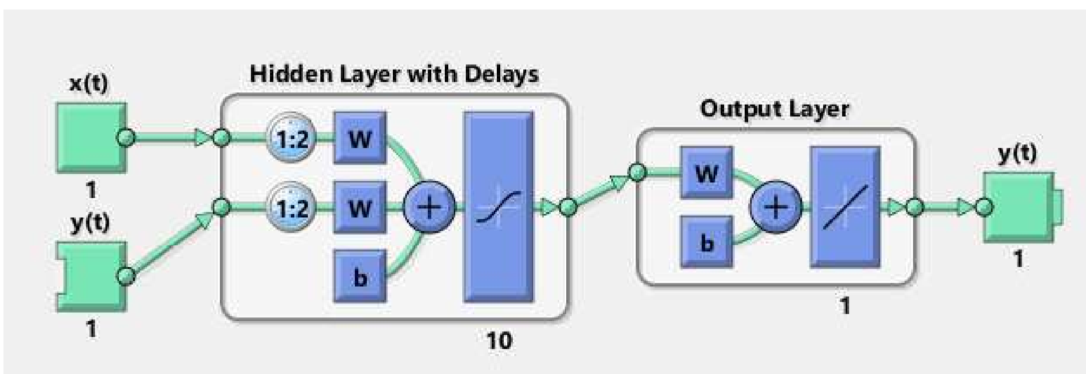


Figure 3.6: NARX Artificial Neural Network (MATLAB)

3.3 Evolutionary Algorithms

3.3.1 Real Coded Genetic Algorithm (RCGA)

Genetic Algorithm is one of the powerful optimizing algorithms that operated based on the process of the natural selection and genetic transfer between the generations. Its evolutionary technique that inspired from the biology principles that passes through section, crossover, and mutation operations. It is computed to solve many optimization problems through the global data population search and generations movements and iteration. The generations candidates evaluated by a single or group of fitness functions.

Population Initialization

The real coded genetic algorithm uses the real numbers to form the genes of the parameters and variables for continuous optimization search in space domain. The chromosome of genes is starting from the operating points are selected by the programmed algorithm criteria. The chromosome vector size is selected based on the problem solution requirements. The chromosome consist of many genes and each gene is representing a variable or parameter of the problem. The coding of the chromosome is described based on the number of M genes decision of the round interval between [0,1] and the same process is repeated for each chromosome of the full vector numbers that forming a population of first generation.

The decoding is processed using the function g(d) where d is between zero and one and m is more than one. The corresponding parameter is transformed from the interval and as example function presented as below.

$$\boldsymbol{di} \in [\mathbf{0}, \mathbf{1}] \; \boldsymbol{for} \; \boldsymbol{i} = \mathbf{1}, \mathbf{2}, \mathbf{3}, ., ., ., \boldsymbol{m} \quad (3.23)$$

$$\mathbf{g(d)} = \frac{\mathbf{1}}{\mathbf{2}^{\boldsymbol{m}-\mathbf{1}} - \mathbf{1}} \sum_{\boldsymbol{j}=\mathbf{1}}^{\boldsymbol{m}} \boldsymbol{d}_{\boldsymbol{j}} \mathbf{2}^{\boldsymbol{j}-\mathbf{1}} \quad (3.24)$$

Fitness Function Calculation

The fitness function is the functions selected and calculated to judge the fitness of the solution candidate based on general or dedicated designed fitness function for the problem where normally to minimize or maximize targeted value.

Crossover

The simple crossover operator is generating many different offspring numbers which result the selected number that deciding the once to be considered on the population set in the selection designed as below.

$$\mathbf{I} \in [\mathbf{1}, \mathbf{2}, \dots, \boldsymbol{n} - \mathbf{1}] \quad (3.25)$$

i: It is selected randomly to build the two new chromosomes.

$$\boldsymbol{H1} = (\boldsymbol{c}_{\mathbf{1},}^{\mathbf{1}} \boldsymbol{c}_{\mathbf{2}}^{\mathbf{1}}, \dots \dots, \boldsymbol{c}_{\boldsymbol{i}}^{\mathbf{1}}, \boldsymbol{c}_{\boldsymbol{i}+\mathbf{1}}^{\mathbf{2}}, \dots \dots .., \boldsymbol{c}_{\boldsymbol{n}}^{\mathbf{2}}) \quad (3.26)$$

$$\boldsymbol{H2} = (\boldsymbol{c}_{\mathbf{1},}^{\mathbf{2}} \boldsymbol{c}_{\mathbf{2}}^{\mathbf{2}}, \dots \dots, \boldsymbol{c}_{\boldsymbol{i}}^{\mathbf{2}}, \boldsymbol{c}_{\boldsymbol{i}+\mathbf{1}}^{\mathbf{1}}, \dots \dots .., \boldsymbol{c}_{\boldsymbol{n}}^{\mathbf{1}}) \quad (3.27)$$

The crossover developed into many equations and concepts between 1990 – 1994 as below listed titles for a reference [59].

-Flat crossover

-Arithmetical crossover

-BLX_α crossover

-Linear crossover

-Discrete crossover

-Extended line crossover

-Extended intermediate crossover

-Wright's heuristic crossover

-Linear BGA crossover

-Fuzzy Connectives Based Crossover

Mutation

The random mutation where a chromosome that the gene **Ci** to be muted into the opposite value as $\boldsymbol{C_i'}$ randomly selected from the chromosome domain.

$$\boldsymbol{c_i} \in [\boldsymbol{a_i}, \boldsymbol{b_i}]$$

$$\boldsymbol{c} = [\boldsymbol{c_1}, .., \boldsymbol{c_i}, \dots, \boldsymbol{c_n}] \qquad (3.28)$$

The mutation is also developed into many equations and concepts as below listed.

-Nonuniform mutation

-Muhlenbein mutation

-Discrete modal mutation

-Continuous modal mutation

Table 3.1: RCGA Algorithms

Algorithms	Mutation	Crossover
RCGA1	Random	Simple
RCGA2	Non-uniform ($b = 5$)	Simple
RCGA3	Random	Arithmetical ($\lambda = 0.5$)
RCGA4	Non-uniform ($b = 5$)	Arithmetical ($\lambda = 0.5$)
RCGA5-α	Non-uniform ($b = 5$)	BLX-α ($\alpha = 0, .15, .3, .5$)
RCGA6	Non-uniform ($b = 5$)	Linear
RCGA7	Mühlenbein	Discrete
RCGA8	Mühlenbein	Extended line
RCGA9	Mühlenbein	Extended intermediate
RCGA10	Modal Discrete ($B_m = 2$, $rang_{min} = 1.0e-05$)	Extended intermediate
RCGA11	Modal Continuous ($B_m = 2$, $rang_{min} = 1.0e-05$)	Extended intermediate
RCGA12	Non-uniform ($b = 5$)	Wright's heuristic
RCGA13	Mühlenbein	Linear BGA
RCGA14-Log	Non-Uniform	Logical FCB ($\psi = 0.5$)
RCGA14-Ham	Non-Uniform	Hamacher FCB ($\psi = 0.5$)
RCGA14-Alg	Non-Uniform	Algebraic FCB ($\psi = 0.5$)
RCGA14-Ein	Non-Uniform	Einstein FCB ($\psi = 0.5$)

New Generation

The new generation will be formed as recombination from the best fitness found on the earlier generation after the crossover and mutation process is completed and the rest to be selected randomly from the main set populated at the starting points.

Termination Criteria

The termination of the process is at the point where the optimized targeted solution found, and this decided based on the specifications and quality if the recent best fitness solution and other extended policy of the problem designer expression.

3.3.2 Particle Swarm Optimization

The Particle Swarm Optimization (PSO) is a searching technique that follow some animal social behavior. In a group of birds flying in a flock, the movement of a bird is affected by other birds in the flock. The Particle Swarm Optimization is an evolutionary algorithm that is searching for the global optimal. PSO was introduced by Kennedy and Eberhart in 1995 [60]. Each candidate of the population is simply getting effected by the success of the other neighbor candidate where at the end the cumulative success can reach the optimal area of the wide searching space. The swarm is like the population while the particle is like the candidate or individual. The position of every particle is changing based on experience from neighbors. As calculated by velocity equation $\mathbf{V_i(t)}$ over time step and **i** is the particle position as below.

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1) \tag{3.29}$$

In the starting point of $X_i(0)$ can be selected between X_{min} and X_{max}

$$Xi(0) \in [Xmin\,, Xmax] \tag{3.30}$$

For the global optimal PSO

$$v_{ij}(t+1) = v_{ij}(t) + c_1 r_{1j}(t)[y_{ij}(t) - x_{ij}(t)] + c_2 r_{2j}(t)[\hat{y}_j(t) - x_{ij}(t)] \tag{3.31}$$

$v_{ij}(t)$: is the velocity of the particle **i** in the dimension j=1,….,n_x over step time

$x_{ij}(t)$: is the position of particle **i** in dimension j overstep time.

C1,C2: positive Acceleration Constant

r1j,r2j: are random values in the range [0,1]

y_i: personal best position

The best personal position in the next step where t+1

$$\mathbf{y}_i(t+1) = \begin{cases} \mathbf{y}_i(t) & \text{if } f(\mathbf{x}_i(t+1)) \geq f(\mathbf{y}_i(t)) \\ \mathbf{x}_i(t+1) & \text{if } f(\mathbf{x}_i(t+1)) < f(\mathbf{y}_i(t)) \end{cases} \tag{3.32}$$

where the f(.) is a fitness function. The global best position of y(t) over time is defined as below.

$$\hat{\mathbf{y}}(t) \in \{\mathbf{y}_0(t), \ldots, \mathbf{y}_{n_s}(t)\} | f(\hat{\mathbf{y}}(t)) = \min\{f(\mathbf{y}_0(t)), \ldots, f(\mathbf{y}_{n_s}(t))\} \tag{3.33}$$

n_s: is the total number on the particle swarm.

The PSO does not have parent selection, recombination nor mutation steps.

PSO Algorithm Procedure Summary

```
Create and initialize an nx-dimensional swarm;
Repeat
   For each particle i=1,…..,ns do
//set the personal best solution
If f(xi) < f(yi) then
   yi=xi;
end
//set the global best position if f(yi) < f(ŷ) then
   ŷ= yi;
         End
End
For each particle i=1,……,ns do
         update the velocity using the equation                    (3.31)
         update the position using the equation                    (3.29)
end
until stopping condition is true;
```

3.4 Problem Statement Description

3.4.1 System Identification Main Problems

The system identification is a technique to reach a mathematical model. Many difficulties exist in identification. The main problems facing the system identification are listed as below.

-Model Structure and Selection Complexity

-Captured noise in the collected measured data

-Difficulty on experiments measurement preparations for some applications.

-Time varying system issues and problems happens in time invariant model

3.4.2 Model Structure and Selection Complexity

The model structure selection is one of the main steps of system identifications due to the infinitely many possible structures that could be linear or nonlinear with simple short equation expression and long combination of equations.

Model Structure Choice

-How is the available knowledge processed to provide the dynamic system model without the noise level effects with a good accuracy?

-How the model choice will be selected to represent the actual real system?

-How many parameters does the selected model have?

-How much captured input and output historical data are enough to converge to the right model?

All these listed questions are valid, and this thesis focuses to drive proposed resolutions for the reasonable answers and better identification process.

3.4.3 Extended Problem Statement

This thesis focuses on the system identification of nonlinear turbojet engine systems and other similar applications using the proposed solution of parametric black box method modelling and as extensive focus to search on the suitable nonlinear model structures selection using different approaches of evolutionary algorithms to satisfy one of the main system identification problems. And possibly to work further on the non-parametric system identification such as the neural network methods and the Hammerstein model identification using the same evolutionary approach for more comparisons study and further motivations and suggestions of works.

3.5 Analysis of The Problem

The analysis describes the problem mapping between the given available information and required solution targets with the proposed process scope as below to introduce the methodology and working procedure in following thesis details.

Proposed Solution Steps:

- Apply Random Parametric Nonlinear System Identification Models
- Selecting the Suitable Evolutionary Algorithm
- Designing the Fitness Function of the Problem
- Applying the new approaches for the system identification and evolutionary algorithm combination.
- Designing the criteria and specification for the best model selection
- Testing the non-parametric system identification methods

CHAPTER 4

THE PROPOSED ALGORITHM AND METHODOLOGY

4.1 Random Parametric Nonlinear System Identification Models

The random parametric nonlinear system identification models contain some mixed ARX and NARX models from infinitely many possible models that assumed nonlinear regressors structure to be used as trial solution for the problem in recursive identification method.

Commonly Used Application

The proposed applications used the NARX, ARX identification models in the recursive least square method based on the historical collected data of the jet engine. The steps are as follows.

The auto-regressive exogenous (ARX) model proposed structure.

$$\hat{y}(t) = -a_1\, y(t-1) \ldots . -a_n\, y(t-n) + b_1\, u(t-1) + \cdots + b_m\, u(t-m) \quad (4.1)$$

where $\hat{y}(t)$ is the predicted output. The input and output data are given by

$u(t)\ and\ \ y(t)$.

The parameter vector is given by.

$$\theta = [a_1\, a_2 \ldots\, a_{na}\ \ b_1\, b_2 \ldots\, b_{nb}\ ]^T \quad (4.2)$$

4.2 Identification and Evolutionary Algorithm Combination Approach

The number of possible nonlinear model structures is infinite with wide range of results. The tested limited trials might miss better ones. Thus, the approach uses the evolutionary genetic algorithms combined with system identification methods to search in selected domain of all proposed possible model structures with optimized results.

4.2.1 RCGA Evolutionary Algorithm Selection

The evolutionary algorithms are differing from one to one on the designed process and optimization features. The problem requires the most applicable algorithm to reach better results. The Real Coded Genetic Algorithm (RCGA) is selected for the best model structure search due to the wide possibility in process modifications and integrations.

4.2.2 Process Design and Creation

The main RCGA block diagram is followed in each step to fit the problem and give proper results.

Population Initialization

The population initialization is designed for the wide range of regressors that is selected randomly for each component of the model structure to form the population from different ten proposed candidates initially that increases with next trials and examples.

-Initialization Parameters

Crossover Parameter Pc=0.9;

Mutation Parameter Pm=0.005;

The initialization parameters are selected and tuned based on results from first to second trial. Here the mutation parameter is almost zero with higher number into the crossover parameter to be major process of the optimization algorithm.

Fitness Function Optimization

The fitness function is basically the objective function to be maximized or minimized in solving this problem. The problem may involve more than one fitness function as in the case of multi-objective functions. In this thesis the fitness function designed as minimization of the mean square error. The evaluation process was the key point to select the new structure candidate populations. The more searching time and regressors choices are the more optimized model structure. The evaluation of each model structure is applied in each generation iteration of the search process.

The evaluation follows the same earlier mentioned process and steps on the commonly used application of finding the estimated parameters for the proposed regressors model structure in the recursive method and calculate the mean square error for the same generation.

-Optimal Model

Finding the optimal and most accurate model in each iteration using if function statement.

-Parent selection

It is selecting each couple candidates of the proposed model structures by the sequence of the generation list. The parents follow the crossover and mutation process to generate the new child generation of new model structures and continues for all new iteration and generation to produce more model structure proposals.

Crossover

The crossover process applies between each created couple of parents of the generation if the random number less than or equal to the crossover parameter set. The crossover is switching the regressors between the model structures proposed in parent one and parent two that forming new candidates. The crossover process is repeated four times to cover the proposed four regressors of each model structure that can be extended into unlimited number of regressors based on the purpose of the example and required trials.

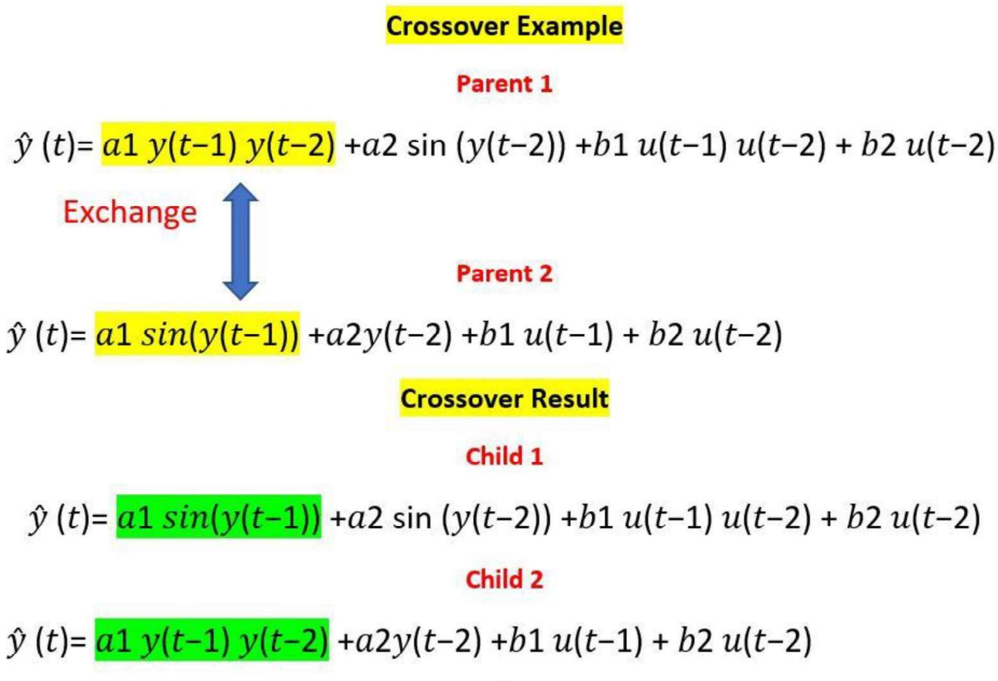


Figure 4.1: Crossover Process Example

Mutation

The mutation process applies between each created couple of parents of the generation if the random number less than or equal to the mutation parameter set. The mutation is switching the regressors into the main NARX model structures regressor in parent one and parent two that forming new candidates.

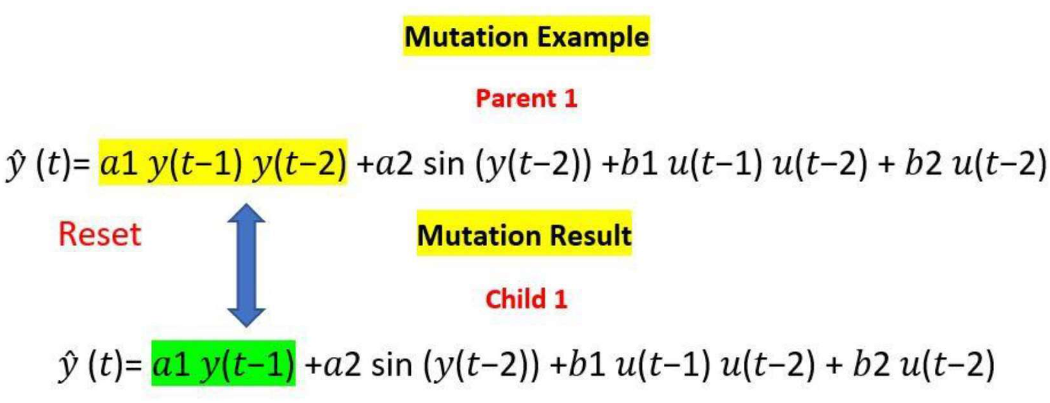


Figure 4.2: Mutation Process Example

The mutation process is repeated four times to cover the proposed four regressors of each model structure that can be extended into unlimited number of regressors based on the purpose of the example and required trials.

New Generation

The new child generation is created after the crossover and mutation with new features and fitness numbers. It is going for evaluation and storage list for the new iteration and generation of generation process.

- **Evaluation**

  The new child generation candidates are passing again through the fitness function and mean square error for the optimal model search and optimization continuous process as mentioned on the population initialization.

- **Storage and Next Iteration**

After the evaluation and fitness function the new child generation is listed for the next iteration to create the child of child candidates. The loop software is storing the best new generated candidates to be used for the next iteration process.

4.2.3 Test Criteria and Process Termination

The termination criteria are selected based in the performance results of the optimization values and optimal model improvements.

4.3 Summary

In this chapter, we presented identification algorithm combining least squares and evolutionary algorithms. The concepts of crossover and mutation were discussed specifically for the genetic algorithm that will be used in the simulation in the coming chapter.

CHAPTER 5

SIMULATION RESULTS

5.1 Introduction

On the simulation framework started with eight random nonlinear NARX models for parametric system identification of the turbojet engine historical data that covering the designed mean squares errors and fitness function criteria for the comparison on the performance results and the most proper model selection. Furthermore, the evolutionary algorithm for system identification proposed approach is applied on the same mentioned framework sequence. Comparison between the classical nonlinear parametric system identification with the derived new approach of the genetic algorithmic and system identification is given.

5.2 First Case Study: Classical Applications Simulation

On the first part of the simulation, we introduced eight nonlinear parametric NARX model structure simulations for the first case study with the supportive simulation figures, mean squares errors and the fitness function results. In the following we apply the recursive least squares method to obtain models of the system assuming the proposed model structures. For each model, the mean square error of the tested model and the fitness function number as a comparison criterion between the different nonlinear models.

5.2.1 Proposed Applications of Methodology

There are infinite number of possibilities of nonlinear models. Here, we are selecting eight different models to be applied on the same common application which are only differs on the steps related to model structure.

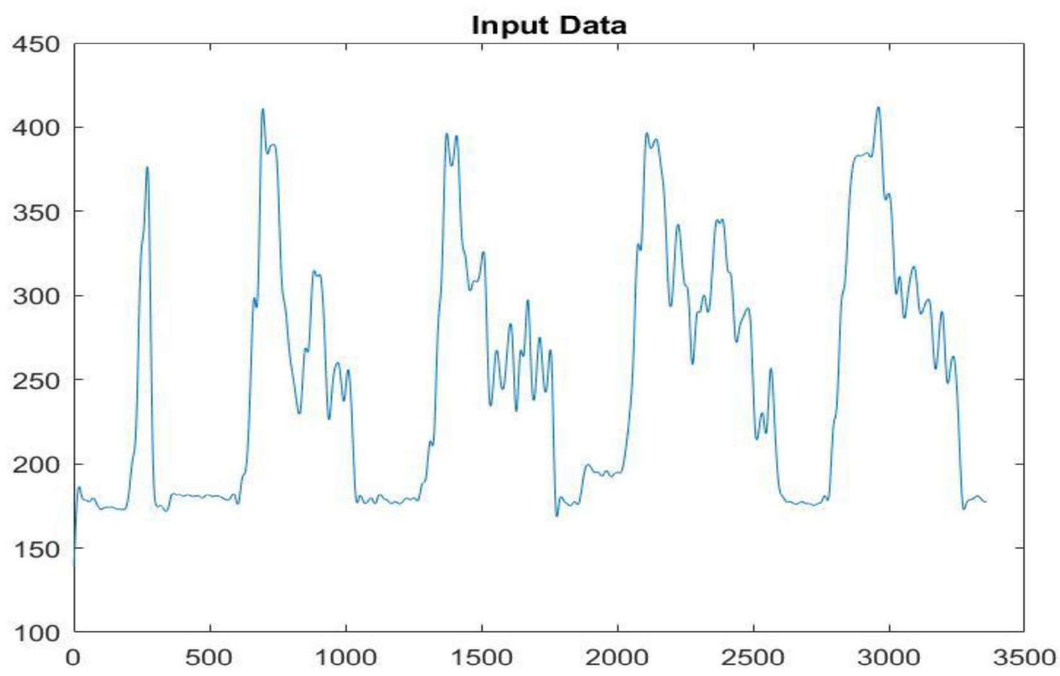


Figure 5.1: Turbojet Engine Historical Input Data

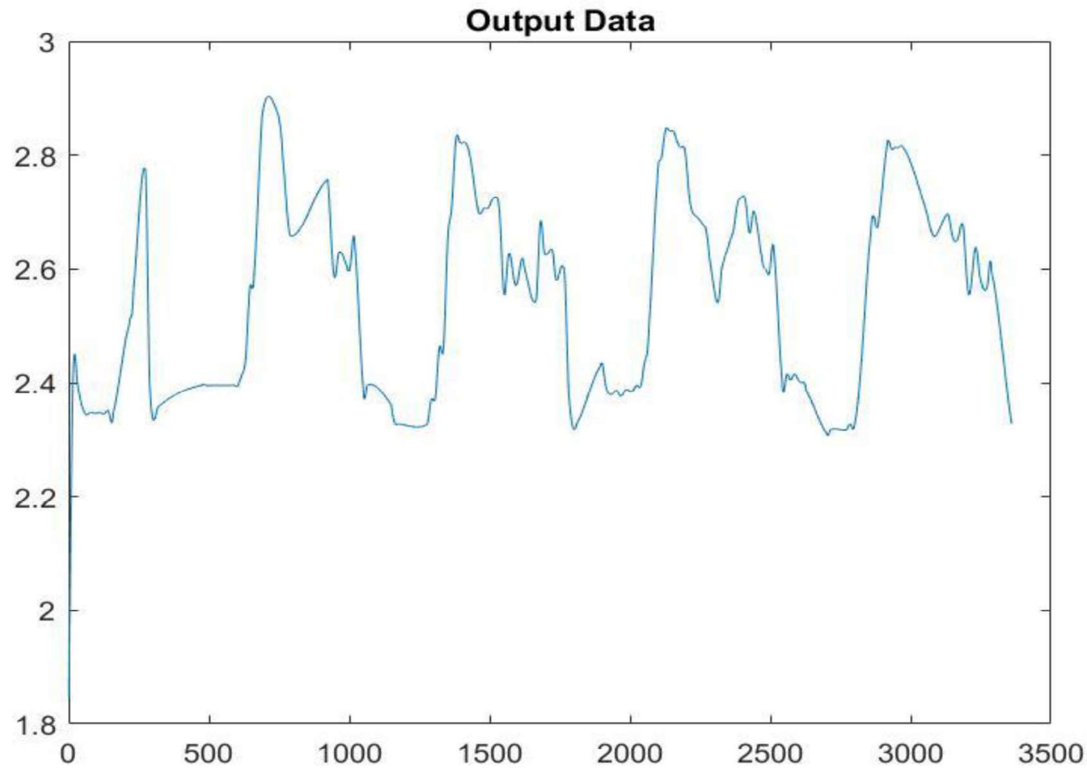


Figure5.2: Turbojet Engine: Historical Output Data

Digitizing the input and output samples from 1 to 3360 historical recorded samples of time.

The eight proposed models are given in the following Table 5.1

Table 5.1: Suggested Model Structures

No.	Suggested Model Structures
1	$\hat{y}\,(t) = a_1\, y(t-1) + a_2 \sin\big(y(t-2)\big) + b_1\, u(t-1)\, y(t-1) + b_2\, u(t-2)\, y(t-2)$
2	$\hat{y}\,(t) = a_1 \frac{y(t-1)}{\exp\,(y(t-1)} + a_2 y(t-2) + b_1\, u(t-1) + b_2\, u(t-2)$
3	$\hat{y}\,(t) = a_1 sin(y(t-1)) + b_1 u(t-1) * y(t-1)$

4	$\hat{y}\,(t) = \; a_1 \;\; exp\big(y(t-1)\big) + a_2 y(t-2) + b_1\, u(t-1) + \; b_2\, u(t-2)$
5	$\hat{y}\,(t) = a_1 \;\; \frac{y(t-1)}{y(t-2)} + b_1\, u(t-1) + b_{21}\, u(t-2))$
6	$\hat{y}\,(t) = \; a_1 \;\; y(t-1) + a_2 \sin\,(y(t-2)) + b_1\, u(t-1) + \; b_2\, u(t-2)$
7	$\hat{y}\,(t) = \; a_1 \; \frac{y(t-1)}{y(t-2)} + a_2\, \sin\,(y(t-2)) + b_1\, u(t-1)u(t-2) + \; b_2\, u(t-2)$
8	$\hat{y}\,(t) = a_1 y(t-1) + a_2\, \sin(y(t-2)) + b_1 u(t-1) + b_2\, u(t-2)y(t-2)$

The identified models are given in the following pages.

Model Structure No.1

$$\hat{y}\,(t) = -0.9948 \;\; y(t-1) - 0.0112 \sin\big(y(t-2)\big) + 0.0004u(t-1)\, y(t-1)$$

$$-0.0004\, u(t-2)\, y(t-2) \qquad (5.1)$$

MSE= 1.6486E-05   and Fitness= 0.0237

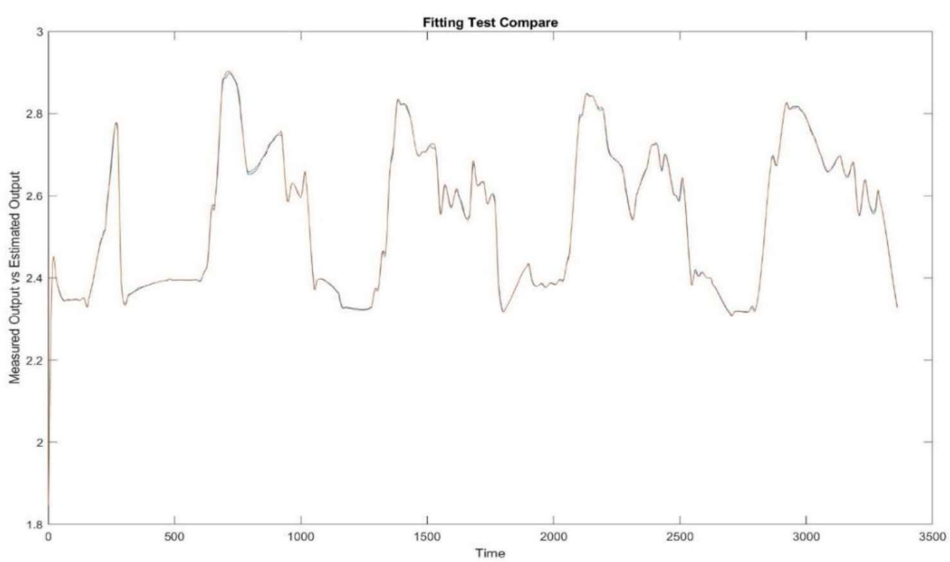


Figure 5.3: Comparison of Measured and Estimated Output for Model No.1

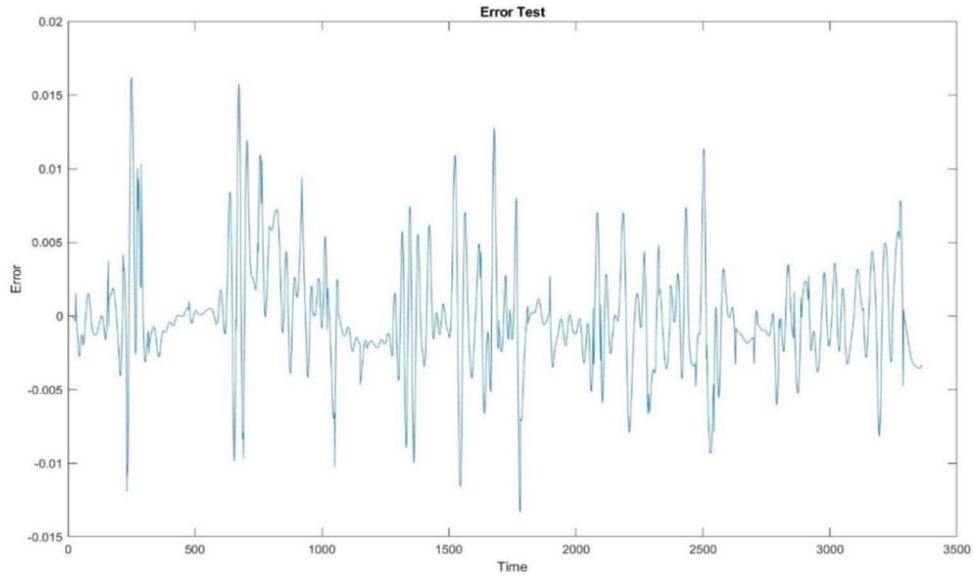


Figure 5.4: Error Test for Model No.1

Model Structure No.2

$$\hat{y}\,(t) = 0.1174\,\frac{y(t-1)}{\exp\,(y(t-1)} + 0.9835y(t-2) + 0.0019\,u(t-1))$$

$$-0.0018\,u(t-2) \tag{5.2}$$

MSE= 6.5921E-05 and Fitness= 0.0475

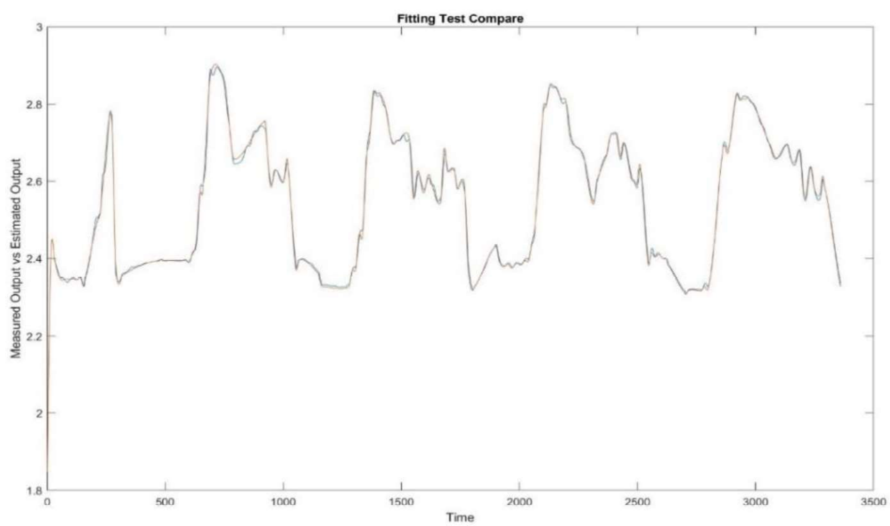


Figure 5.5: Comparison of Measured and Estimated Output for Model No.2

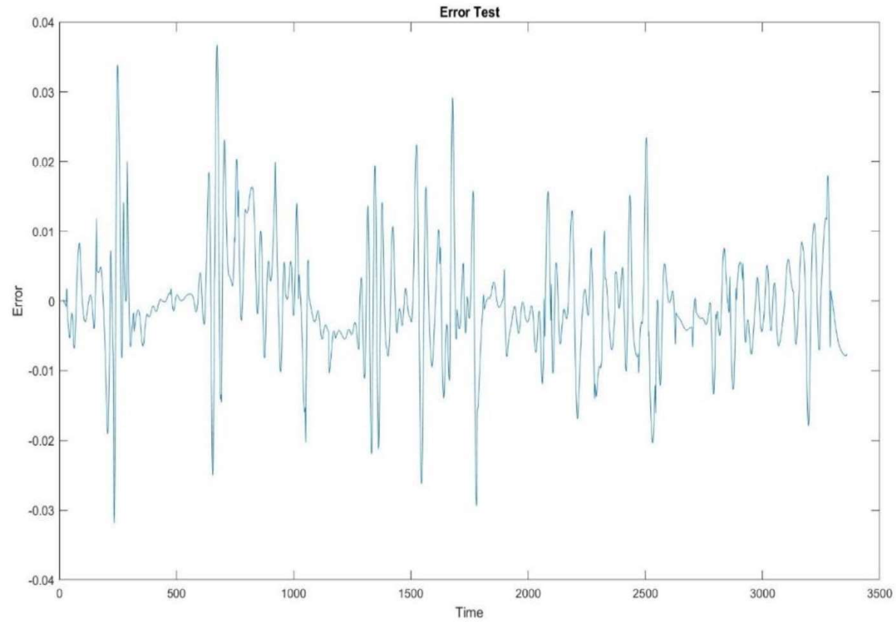


Figure 5.6: Error Test for Model No.2

Model Structure No.3

$$\hat{y}\,(t) = -\,2.1448\,sin(y(t-1)) + 0.0021u(t-1) * y(t-1) \quad (5.3)$$

MSE = 0.0273 and Fitness= 0.9656

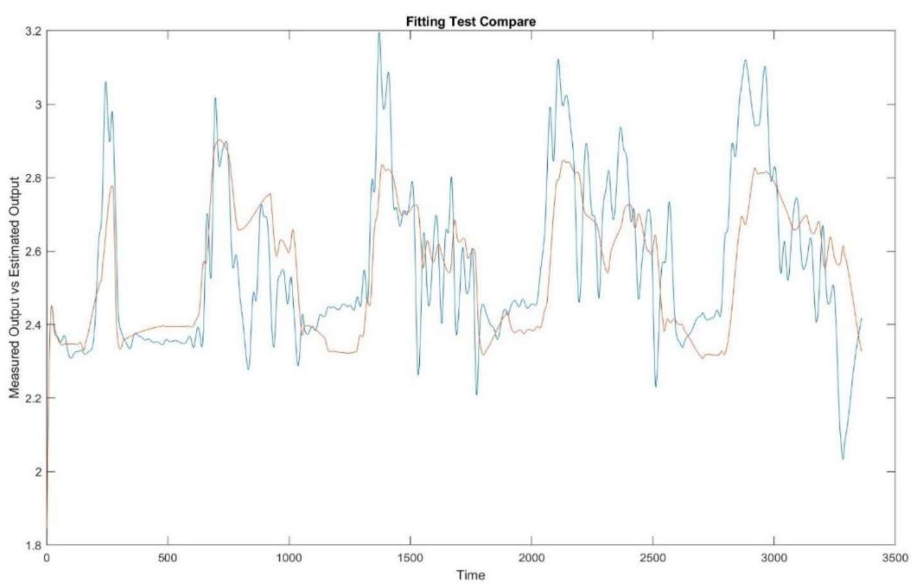


Figure 5.7: Comparison of Measured and Estimated Output for Model No.3

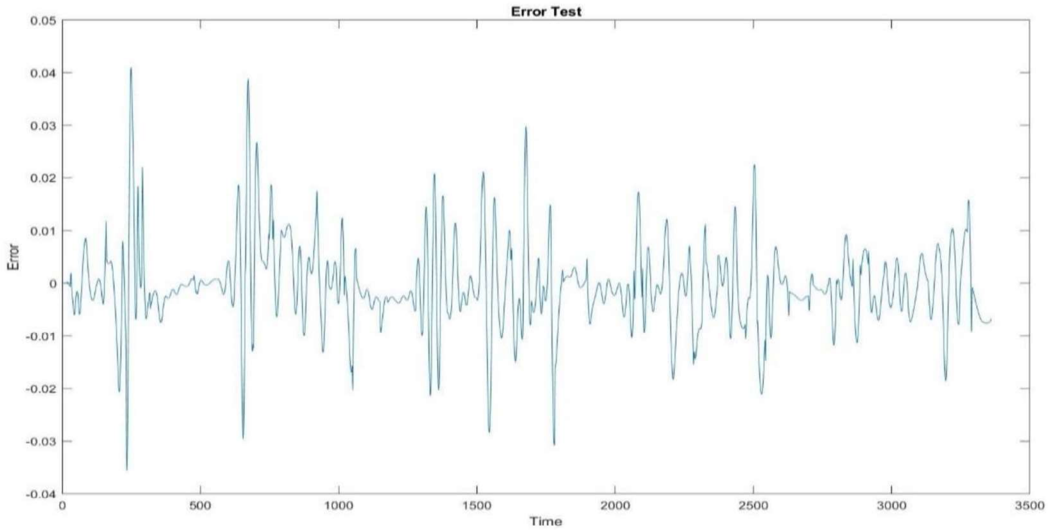


Figure 5.8: Error Test for Model No.3

Model Structure Vector No.4

$$\hat{y}\ (t) = \ -0.0019\ \ exp\big(y(t-1)\big) + 1.0044y(t-2) + 0.002\ u(t-1)\ -0.002\ u(t-2) \tag{5.4}$$

MSE = 6.7980E-05 and Fitness= 0.0482

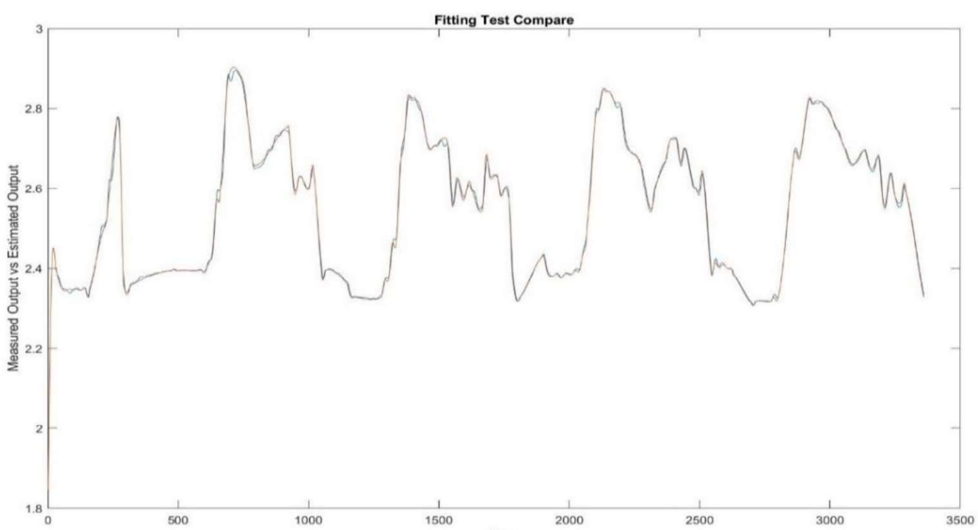


Figure 5.9: Comparison of Measured and Estimated Output for Model No.4

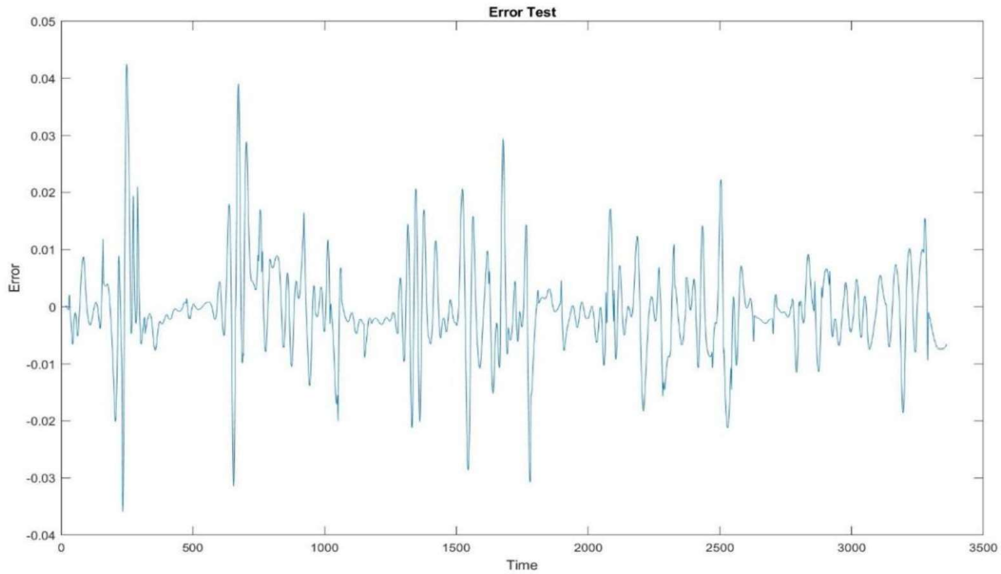


Figure 5.10: Error Test for Model No.4

Model Structure Vector No.5

$$\hat{y}\,(t) = -1.9981\,\frac{y(t-1)}{y(t-2)} - 0.0078\,u(t-1) + 0.0100\,u(t-2)) \qquad (5.5)$$

MSE = 0.0055 and Fitness= 0.4330

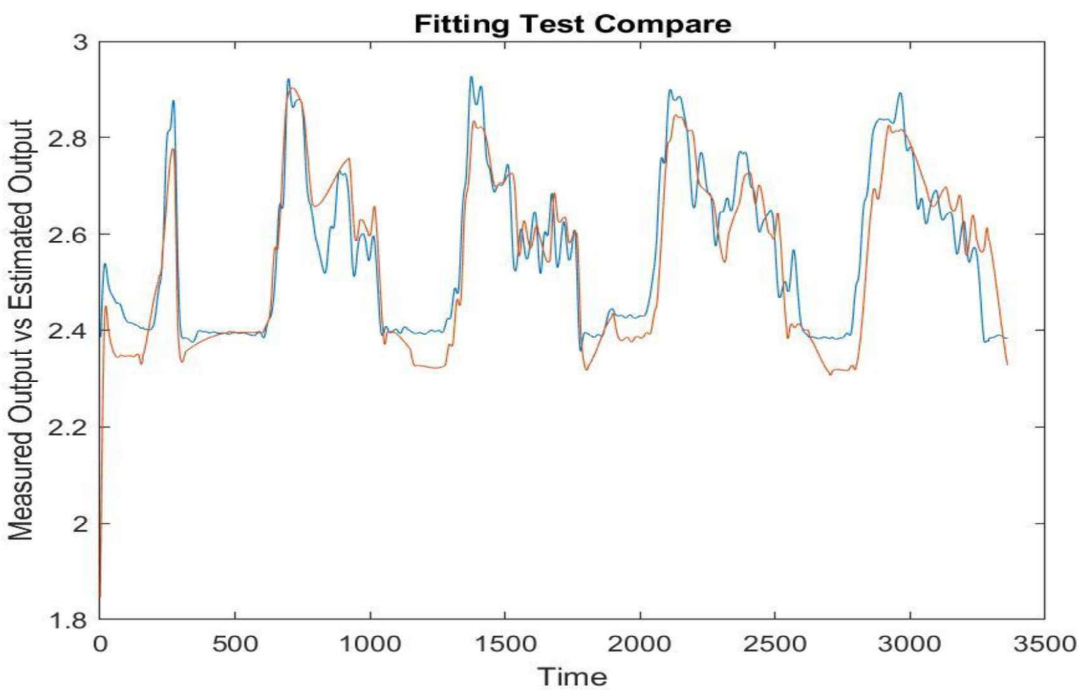


Figure 5.11: Comparison of Measured and Estimated Output for Model No.5

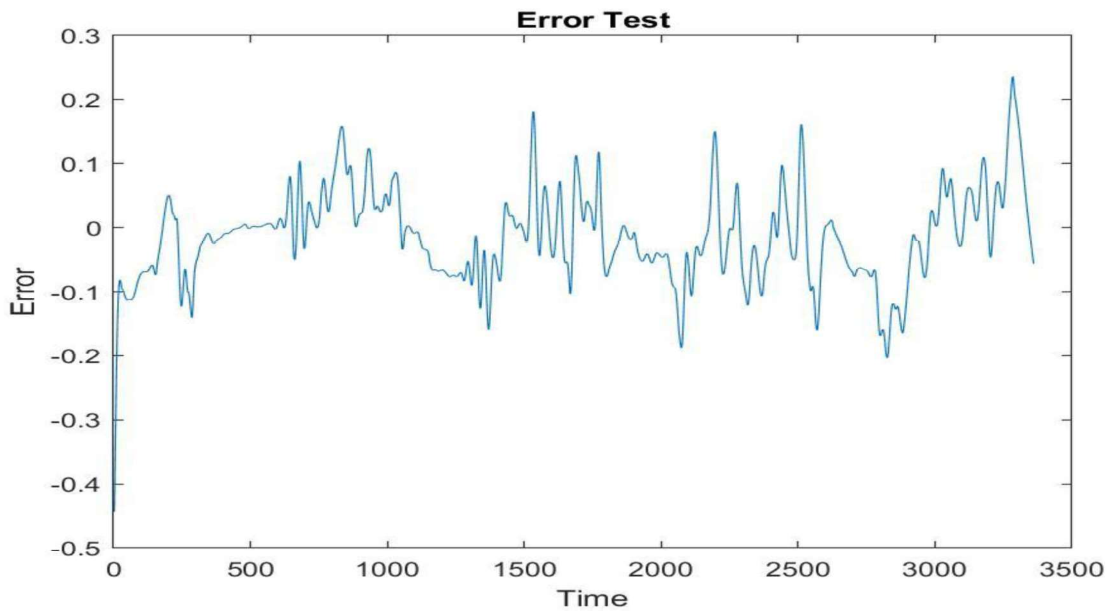


Figure 5.12: Error Test for Model No.5

Model Structure Vector No.6

$$\hat{y}(t) = 0.9940 \ \ y(t-1) + 0.0111\sin(y(t-2)) + 0.001\, u(t-1) \ -0.0009\, u(t-2) \tag{5.6}$$

MSE= 1.8863E-05 and Fitness= 0.0254

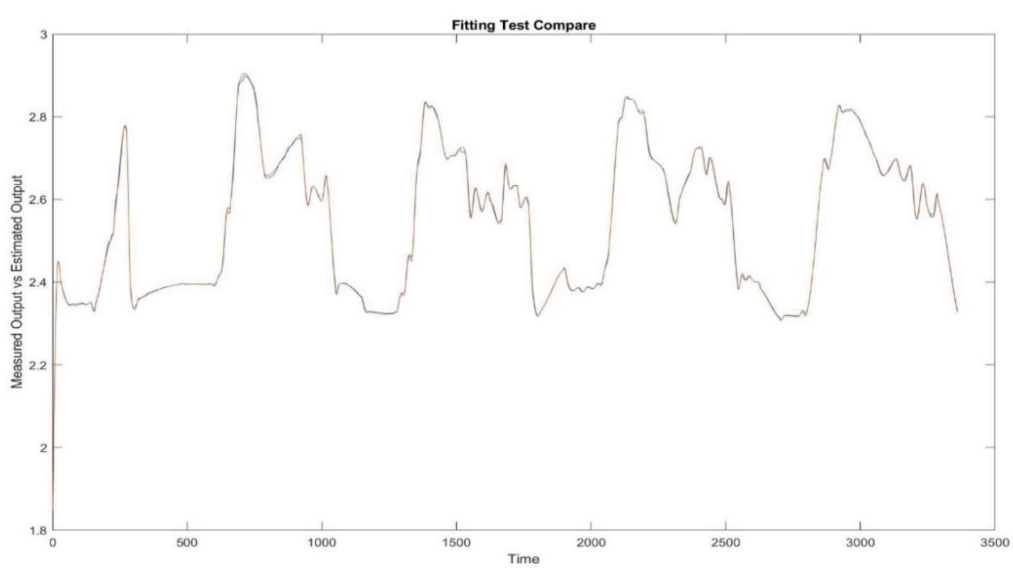


Figure 5.13: Comparison of Measured and Estimated Output for Model No.6

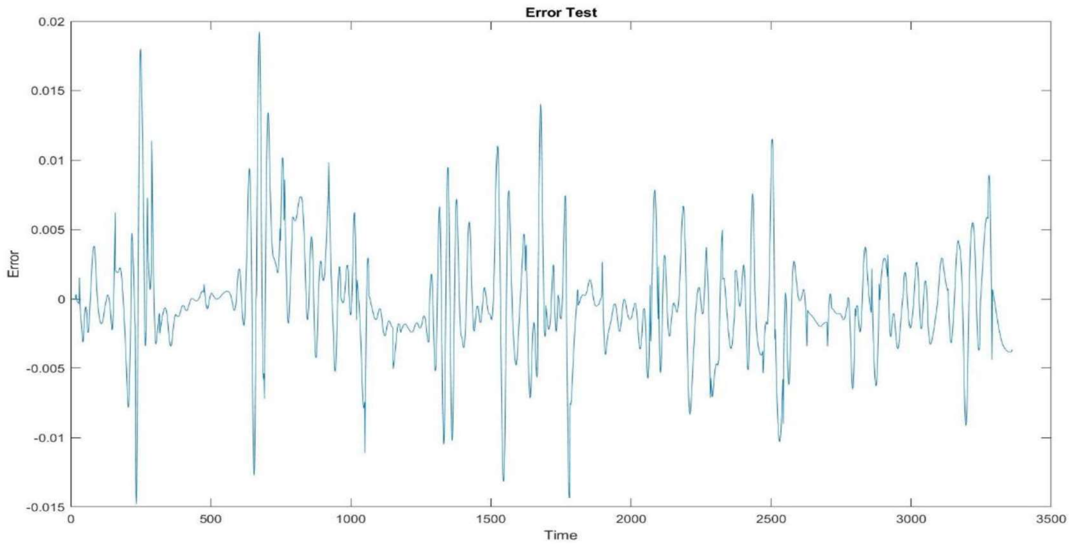


Figure 5.14: Error Test for Model No.6

Model Structure Vector No.7

$$\hat{y}\,(t) = 3.1221\,\frac{y(t-1)}{y(t-2)} - 1.1948\sin\left(y(t-2)\right) - 0.00001\,u(t-1) + 0.0007\,u(t-2) \tag{5.7}$$

MSE = 6.6802E-05 and Fitness= 0.0478

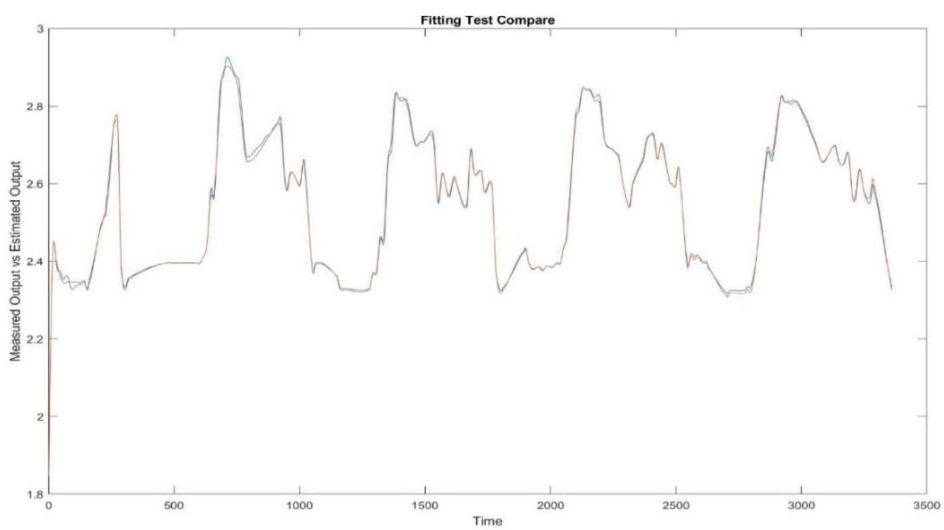


Figure 5.15: Comparison of Measured and Estimated Output for Model No.7

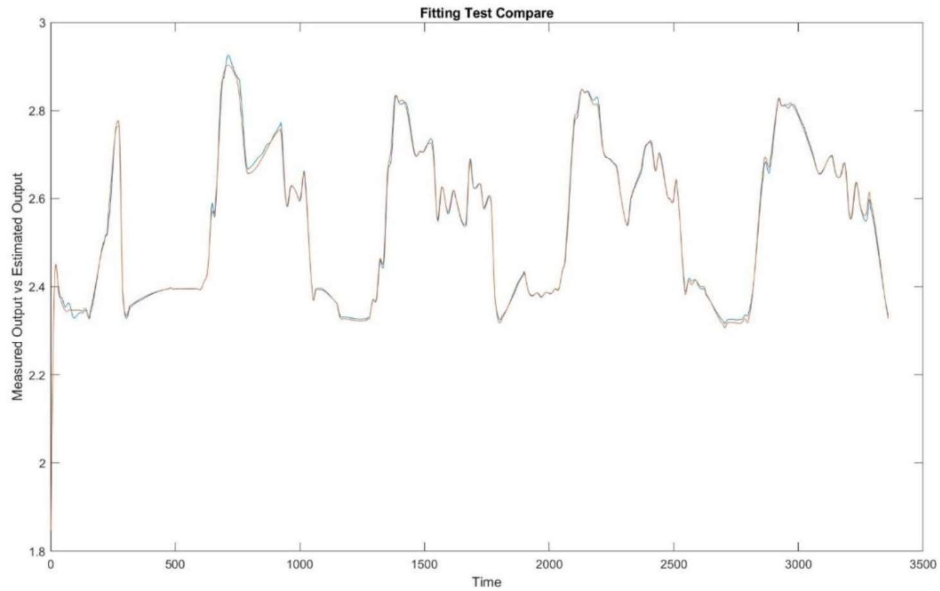


Figure 5.16: Error Test for Model No.7

Model Structure Vector No.8

$$\hat{y}\,(t) = 0.9958y(t-1) - 0.0057\sin\big(y(t-2)\big) + 0.0003u(t-1)$$
$$-0.0001\,u(t-2)y(t-2) \tag{5.8}$$

MSE = 2.0522E-05 and Fitness= 0.0265

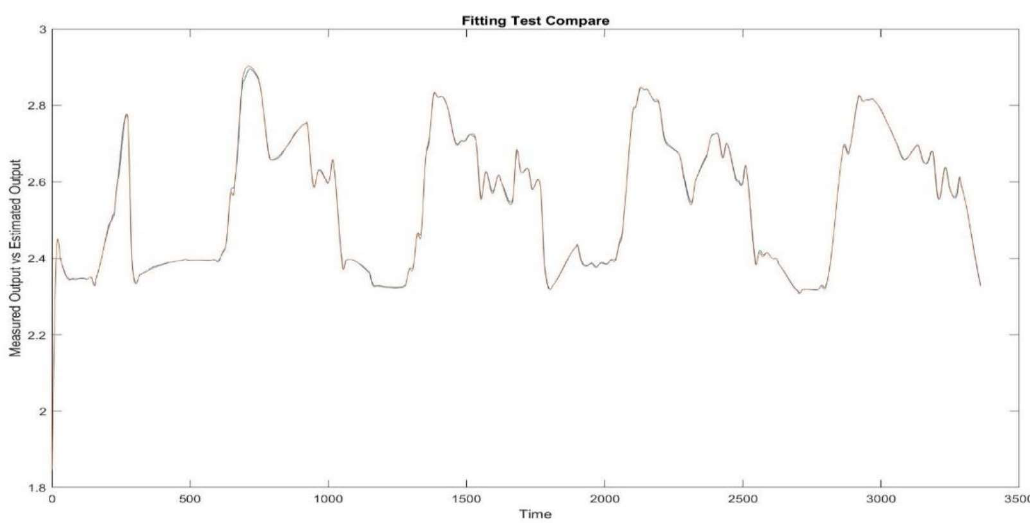


Figure 5.17: Comparison of Measured and Estimated Output for Model No.8

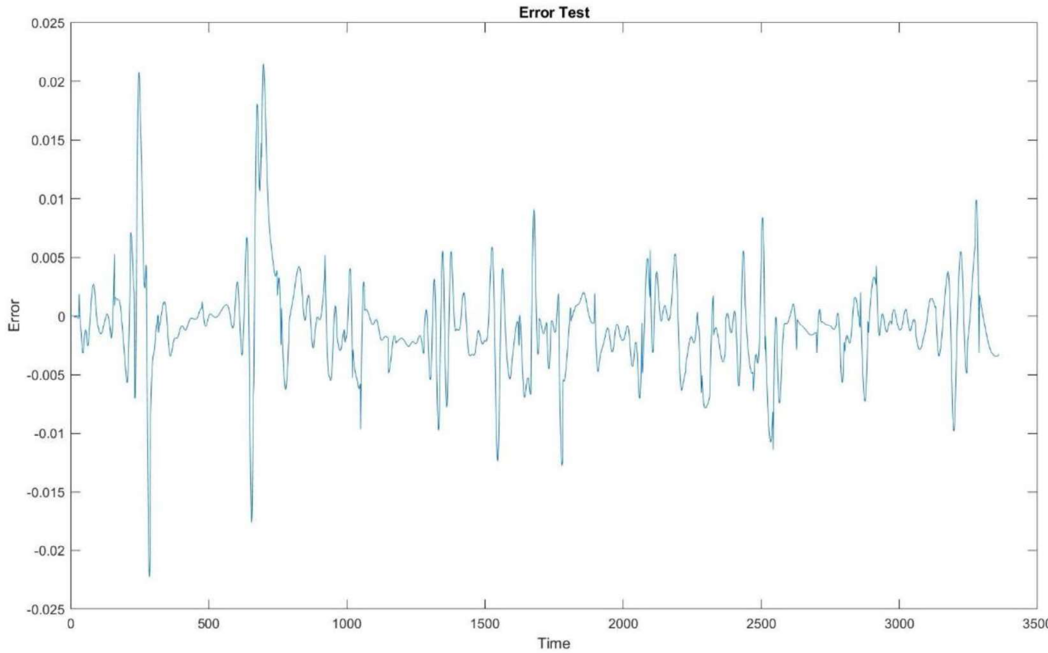


Figure 5.18: Error Test for Model No.8

5.2.2 Applications Summary

The eight applications are tested with the same historical input-output data and following the same sequence. Table 5.2 summarize the obtained results of mean square error and fitness of the tested models. The minimum MSE and fitness corresponds to the more accurate model. The results shows that case number three is having the worst accuracy with MSE 2.7360E-02 and Fitness 0.9656. The case number one is the best accuracy as MSE is 1.6486E-05 and fitness 0.0237. Different model structures have different results that is depending on the assumed model structure.

Table 5.2: Summary of The Model Identification Results

Model No.	MSE	Fitness
1	1.6486E-05	0.0237
2	6.5921E-05	0.0475
3	2.7360E-02	0.9656
4	6.7980E-05	0.0482
5	5.5000E-03	0.4430
6	1.8863E-05	0.0254
7	6.6802E-05	0.0478
8	2.0522E-05	0.0265

5.3 First Case Study: Evolutionary Algorithm for System Identification Approach

On this second part of the simulation, the evolutionary algorithm for the system identification is applied to search on wide ranges of the possible non-linear model structures and tested trough the same MSE and Fitness criteria. The same historical input-output data is used.

Applying the Real Coded Genetic Algorithm for NARX System Identification

Applied the genetic algorithm for system identification approach of the wide non-linear regressors that makes huge number of the proposal model structures for the optimization process of the best model criteria results.

First Trial

First approach model's optimization trial simulation of the minimization mean square error over the generation evolutions, Fitness Test compares the estimated output against the actual one of the found optimized model structure and the mean square error tests.

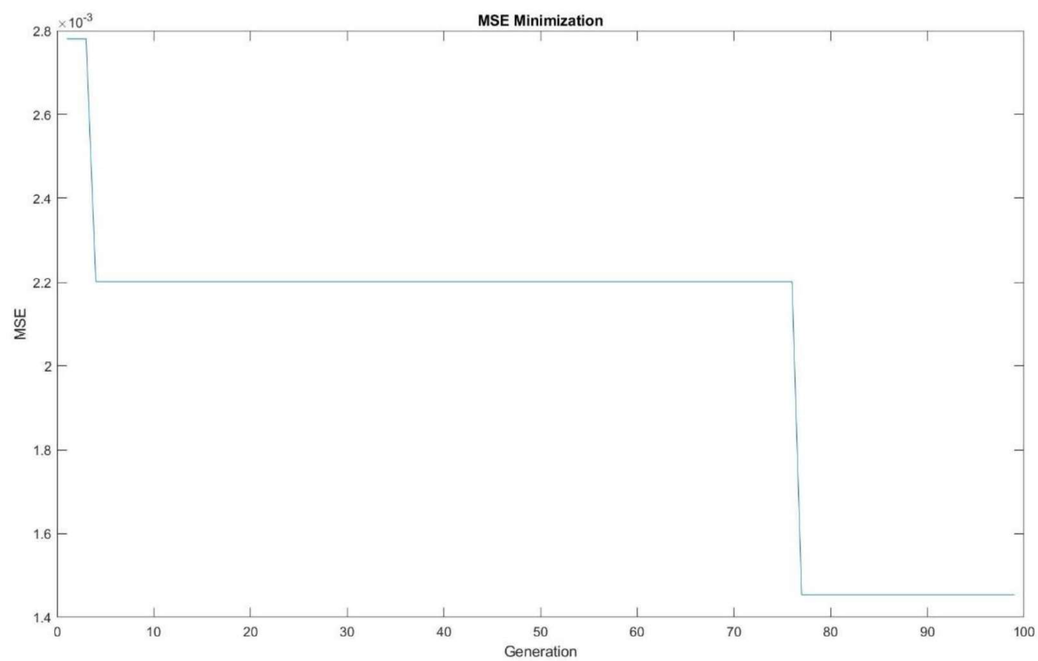


Figure 5.19: Mean Square Error Minimization for Trial No.1

First Optimized Found Model

The optimized model structure of the first trial considered the crossover process without focus on the mutation process. The same MSE and Fitness are used. The simulation is showing the first graph of the comparison between the measure and estimated outputs and the second graph as mean square error tests. The initialization population used 10 random selections. The crossover parameter is 0.9 that have high probability while the mutation is 0.005 which is almost negligible. The convergence shows the first improvement at the fifth iteration and got optimized with the best result at the seventy eighth iteration.

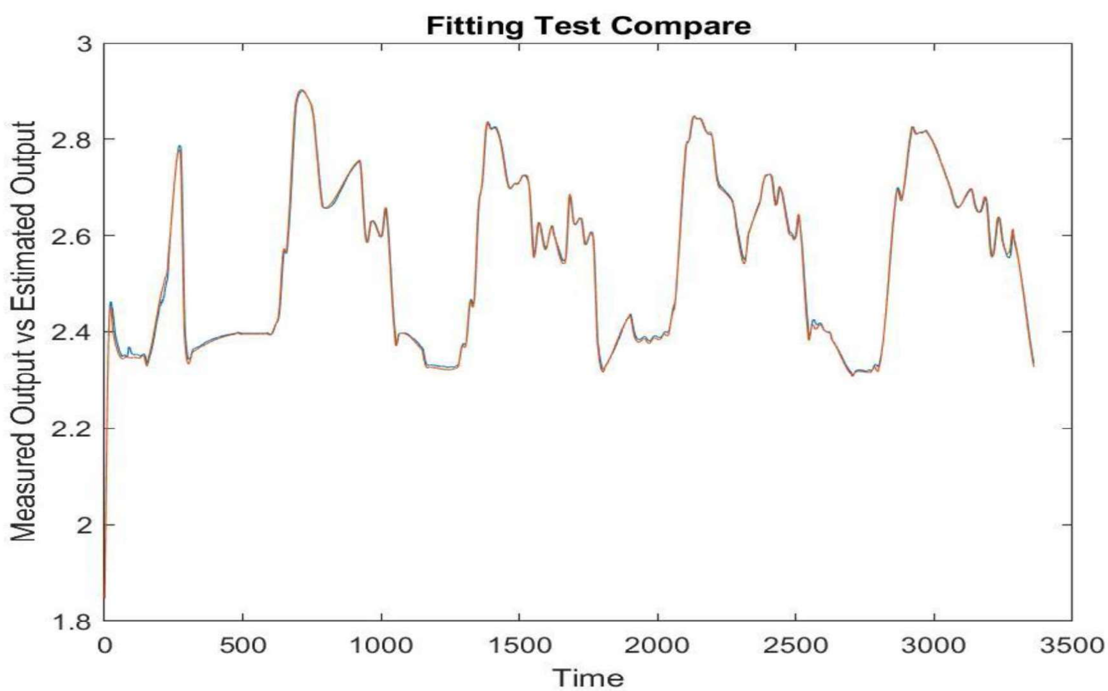


Figure 5.20: Fitting Test Compare for the Optimized Model No.1

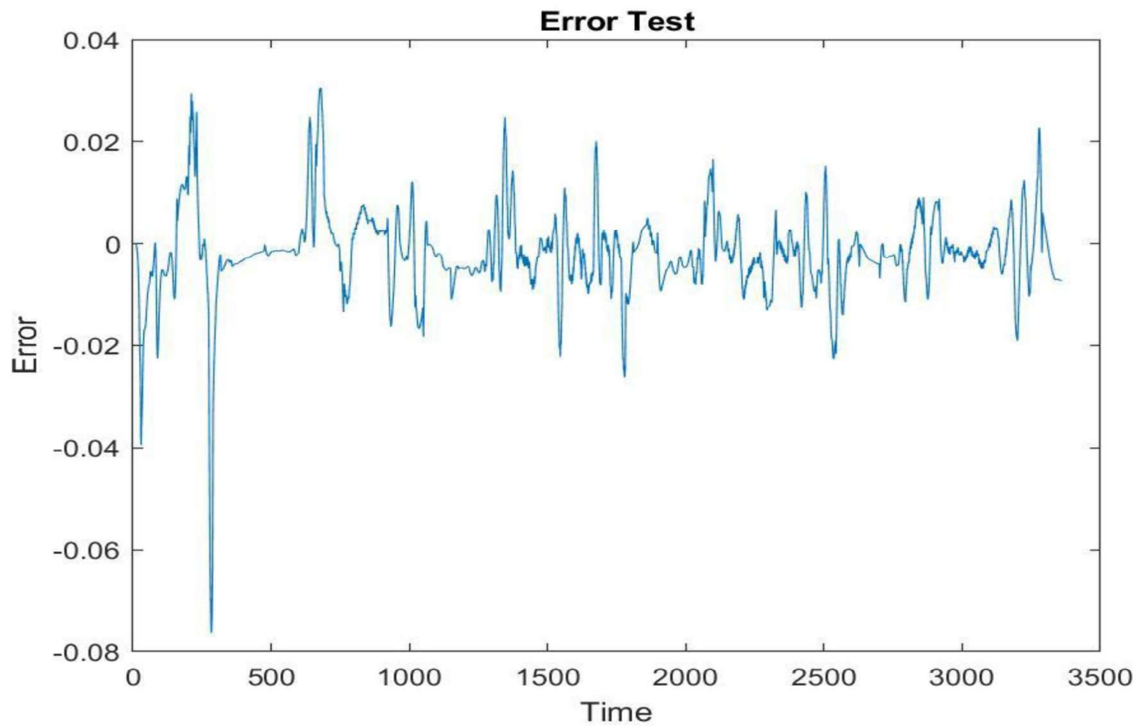


Figure 5.21: Mean Square Error for the Optimized Model No.1

Result:

The model and corresponding MSE and fitness function are given below.

$\hat{y}$(t) =0.9 exp (y(t-1)) - 0.057sin(y-2) +0.03 sin(u(t-1)) - (1/0. 01u(t-2))

-MSE = 7.9361E-05 and Fitness= 0.0521

Second Trial

Second approach models optimization trial simulation of the minimization mean square error over the generation evolutions, Fitness Test compares the estimated output against the actual one of the found final optimized model structure and the mean square error tests.

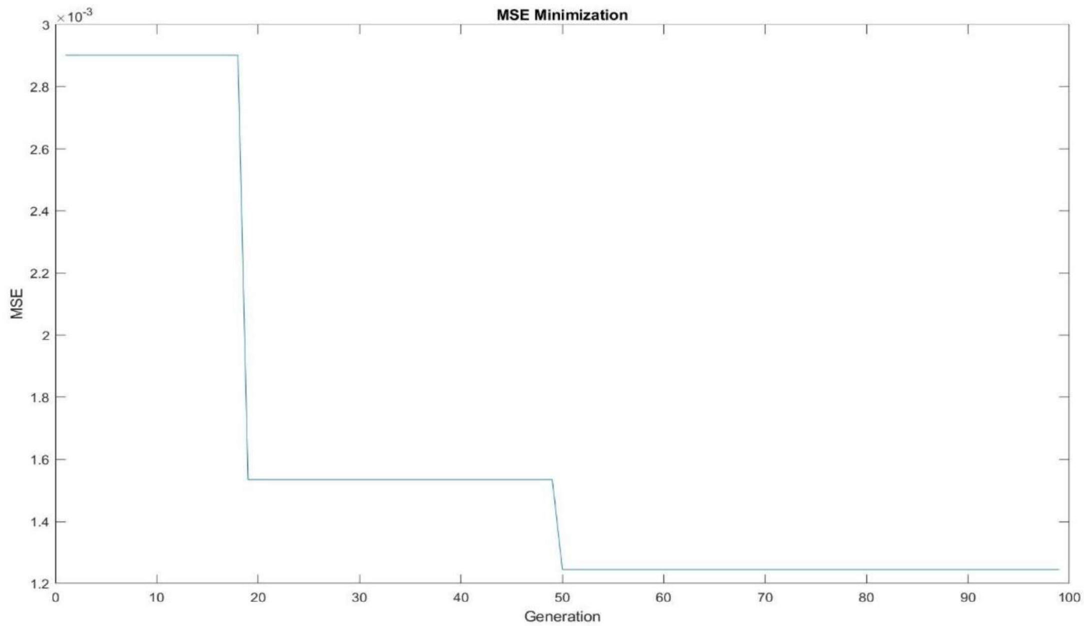


Figure 5.22: Mean Square Error Minimization for Trial No.2

Second Optimized Found Model

The Second optimized model structure tuned with the focus on the crossover and mutation process as well. The same MSE and Fitness are used. The simulation where is showing the first graph of the compression between the measure and estimated outputs and the second graph as mean square error tests. The initialization population used 10 random selections. The crossover parameter is 0.6 that have good probability while the mutation is 0.05. The conversion shows the first improvement at the twenty iteration and got optimized with the best result at the fifty iterations.

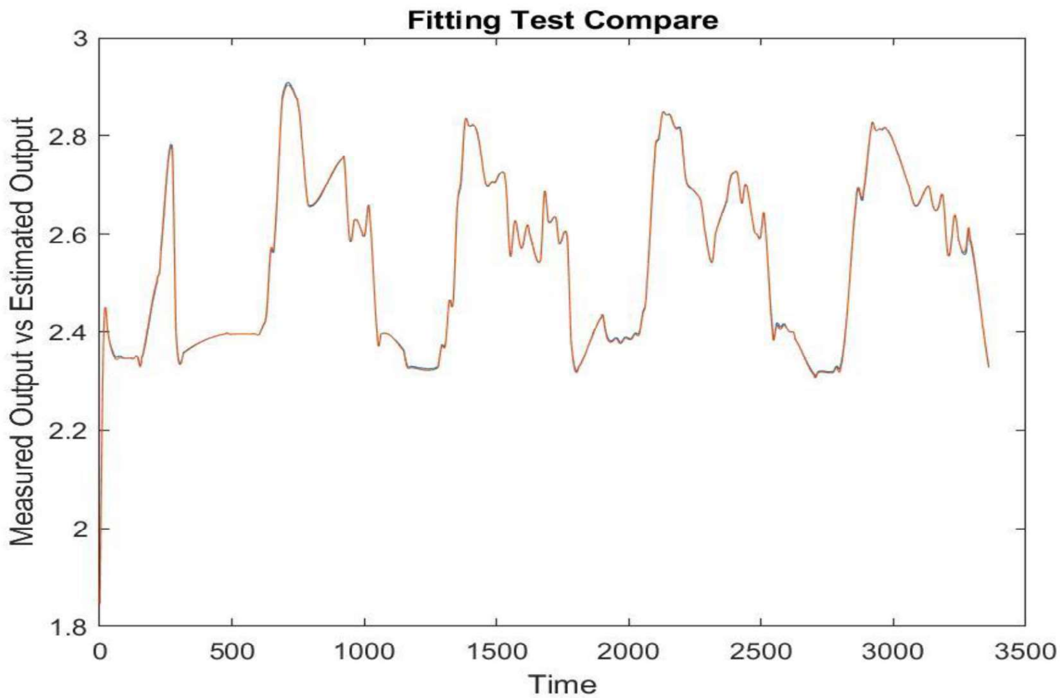


Figure 5.23: Fitting Test Compare for the Optimized Model No.2

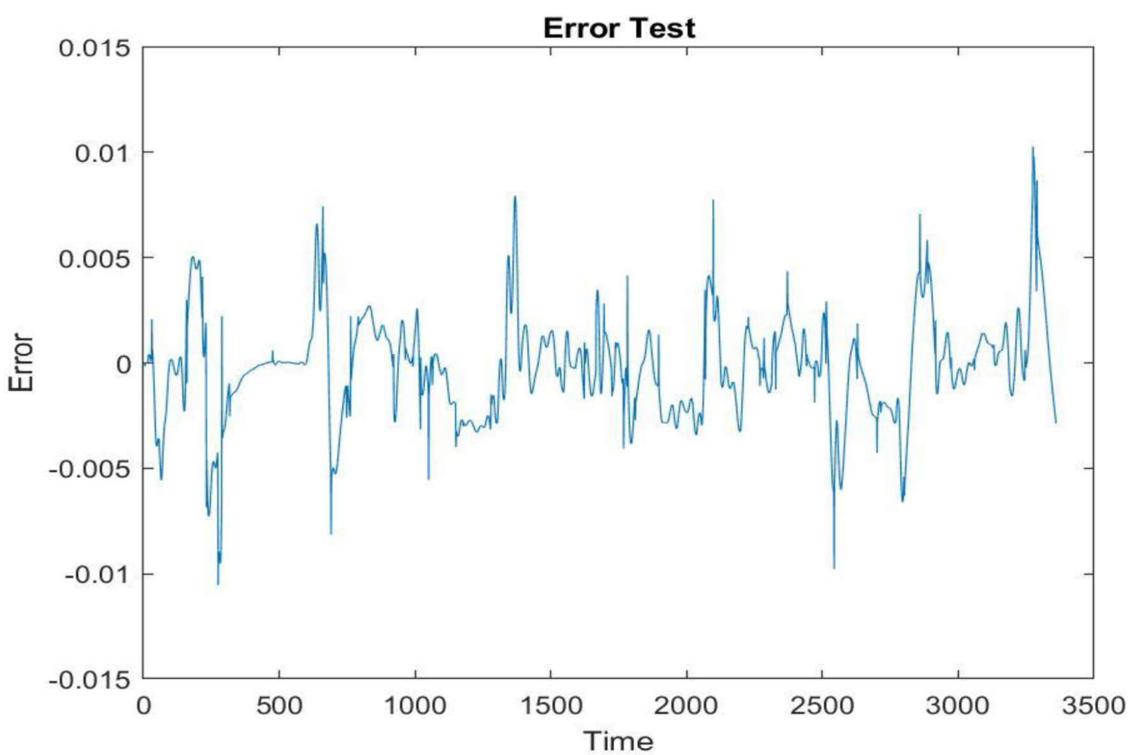


Figure 5.24: Mean Square Error for the Optimized Model No.2

Result:

The model and corresponding MSE and fitness function are given below.

$\hat{y}$= - 0.019 (y(t-1))^2+1.04 sin(y(t-2)) +0.020u(t-1) - ($\frac{\mathbf{50}}{\mathbf{u(t-2)}}$))

MSE = 9.3692E-06 Fitness= 0.0179

5.3.1 Fist Case Study: Approach Summary

The two optimization approach application models are tested with the same historical input-output data and following the same sequence of the first simulation part. Table 5.3 is summarizing the mean square error and fitness of the tested models. The minimum MSE and fitness is the best accurate model and vice versa results.

Table 5.3: Applications Results

Trial No.	MSE	Fitness
1	7.9361E-05	0.0521
2	9.3692E-06	0.0179

The results shows that trial one is having the worst accuracy of MSE 7.9361E-05, Fitness 0.0521 and trial number two is the best accuracy as MSE 9.3692E-06 and fitness 0.0179. This result shows that the second trial gives a model that is better than all the eight model structures presented in Section 5.2. Moreover, in comparison between the different setup crossover and mutation probability parameters between the two optimization trials, the more tuned trial on the second one shows better speed in the solution convergence.

5.4 Second Case Study: Classical Applications Simulation

On this part of the simulation, used two nonlinear parametric NARX model structure simulations for the second case study with the supportive simulation figures, mean squares errors and the fitness function results. In the following we apply the recursive least squares method to obtain models of the system assuming two different model structures proposed. For each model, the mean square error of the tested model and the fitness function number as a comparison criterion between the different nonlinear models.

The historical input-output data is generated using a nonlinear model with additive output noise for the dynamic system below.

$$y(k) = -0.2\, y(k-1) - 1.6 \sin\big(y(k-2)\big) + 0.8\, u(k-1) y(k-1) \qquad (5.11)$$

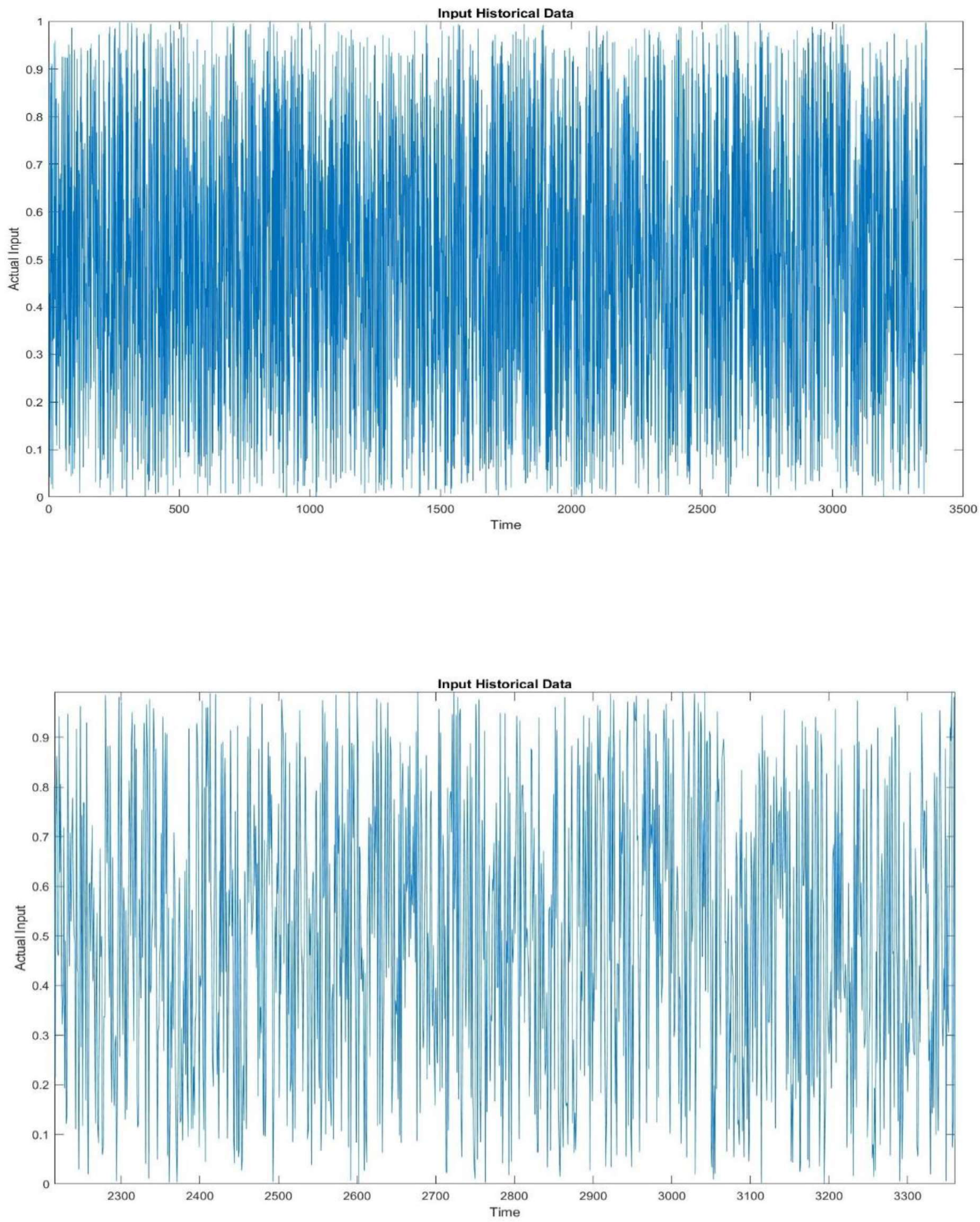


Figure 5.25: Historical Input Data for Dynamic System Case Study 2

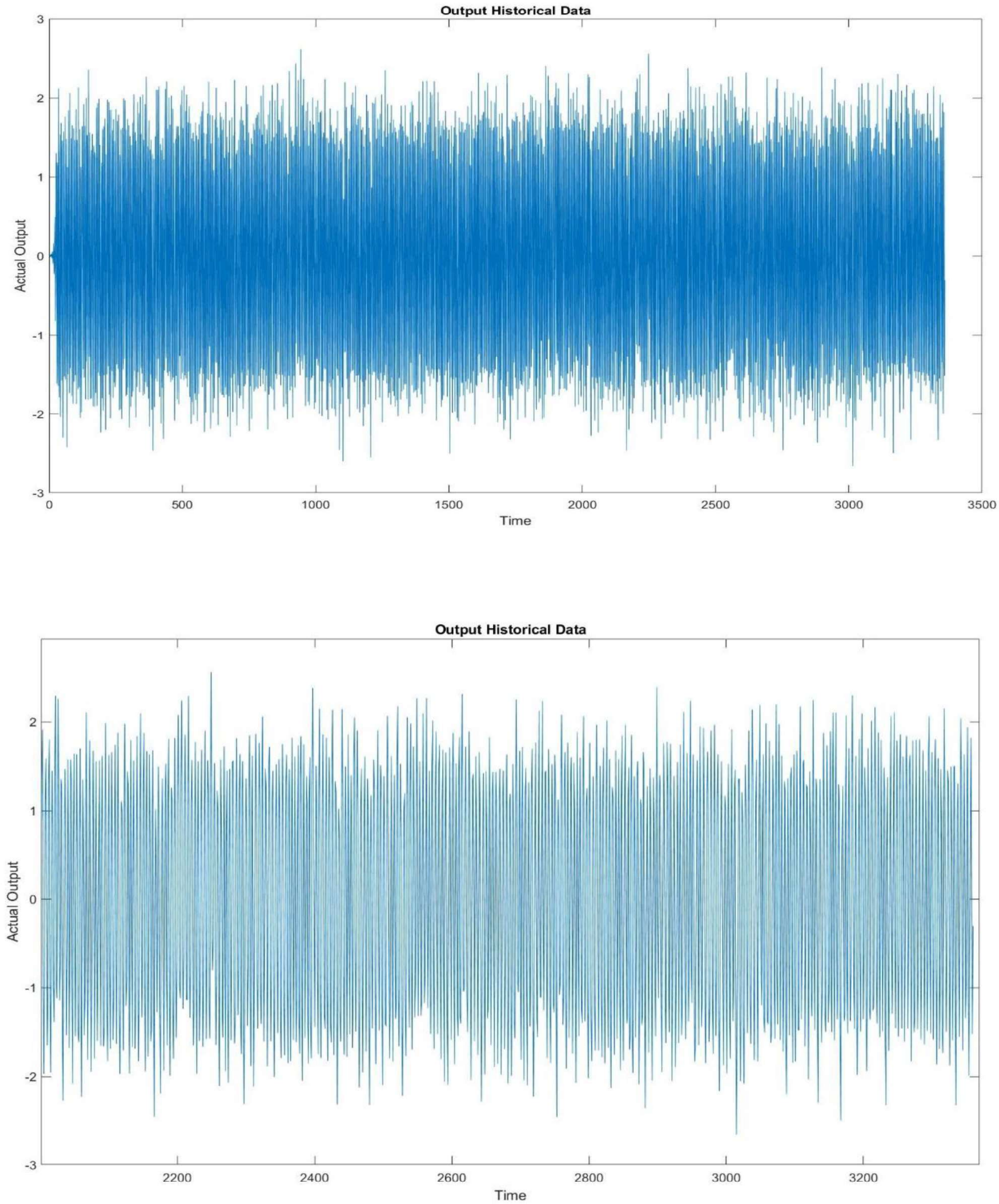


Figure 5.26: Historical Output Data for Dynamic System Case Study 2

5.4.1 Proposed Applications of Methodology

Model 1:

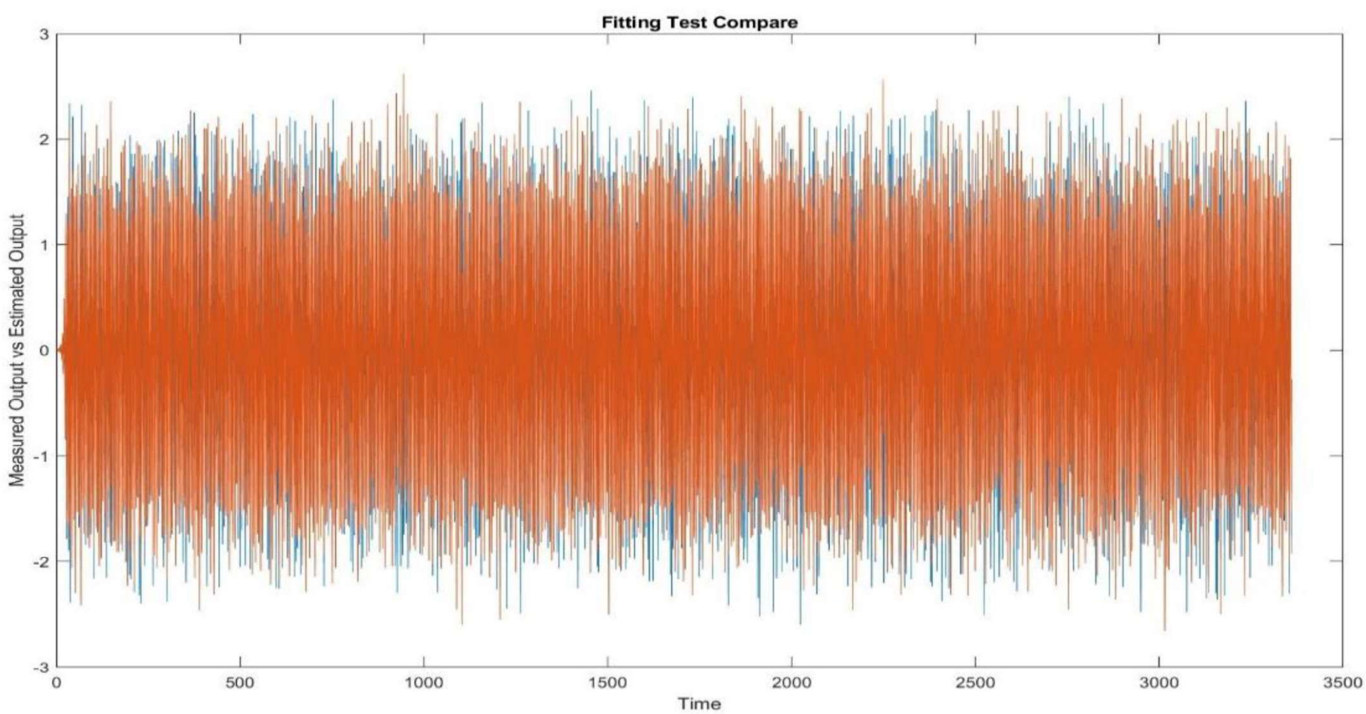
$$\hat{y}\,(\boldsymbol{k}) = \ -\boldsymbol{a_1}\ \boldsymbol{sin}(\boldsymbol{y}(\boldsymbol{k}-\mathbf{1})) - \boldsymbol{a_2}\ \boldsymbol{y}(\boldsymbol{k}-\mathbf{2}) + \boldsymbol{b_1}\ \boldsymbol{exp}\big(\boldsymbol{u}(\boldsymbol{k}-\mathbf{1})\big) + \boldsymbol{b_2}\ /\boldsymbol{sin}(\boldsymbol{u}(\boldsymbol{k}-\mathbf{2}))$$

The result is shown below.

MSE= 0.2427 and Fitness= 0.3831

The obtained parameters

$$[\ a_1\ \ a_2\ \ \ b_1\ b_2\ ] = [\ -0.6837\quad 0.8923\quad 0.0011\quad 0.0000]$$



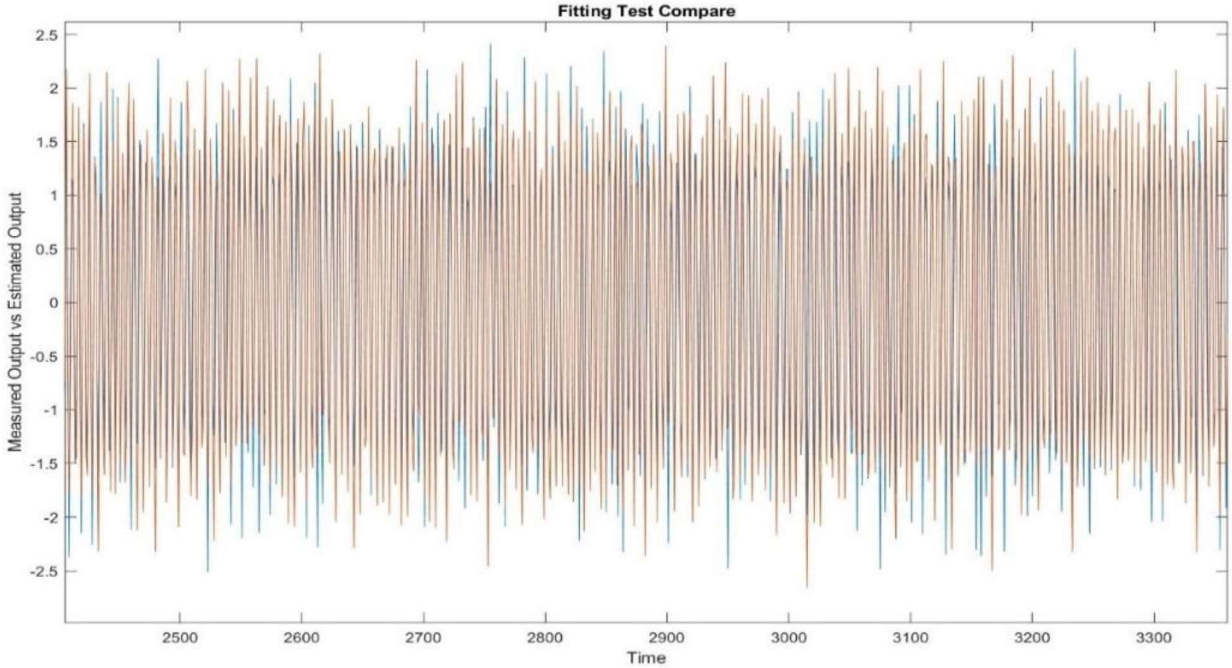


Figure 5.27: Second Case Study, Fitting Test Error for The Optimized Model No.1

The graph shows the general and zoomed simulation of comparison between the measured and estimated outputs.

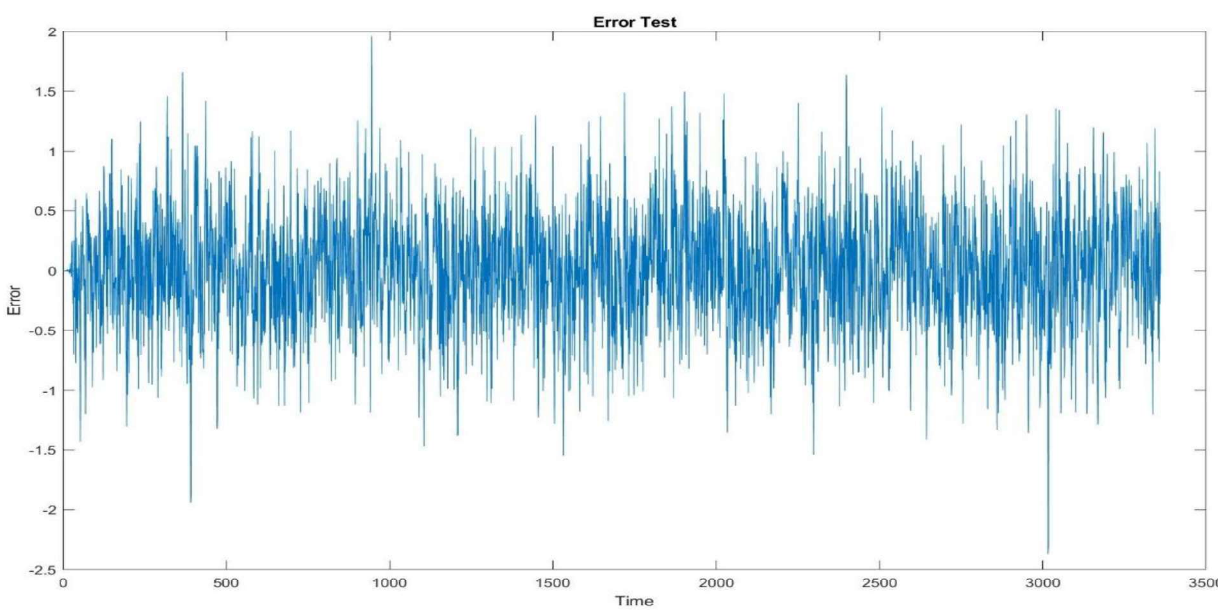


Figure 5.28: Second Case Study, Error for the Optimized Model No.1

The graph shows the changes of the mean square error over the time.

Model 2:

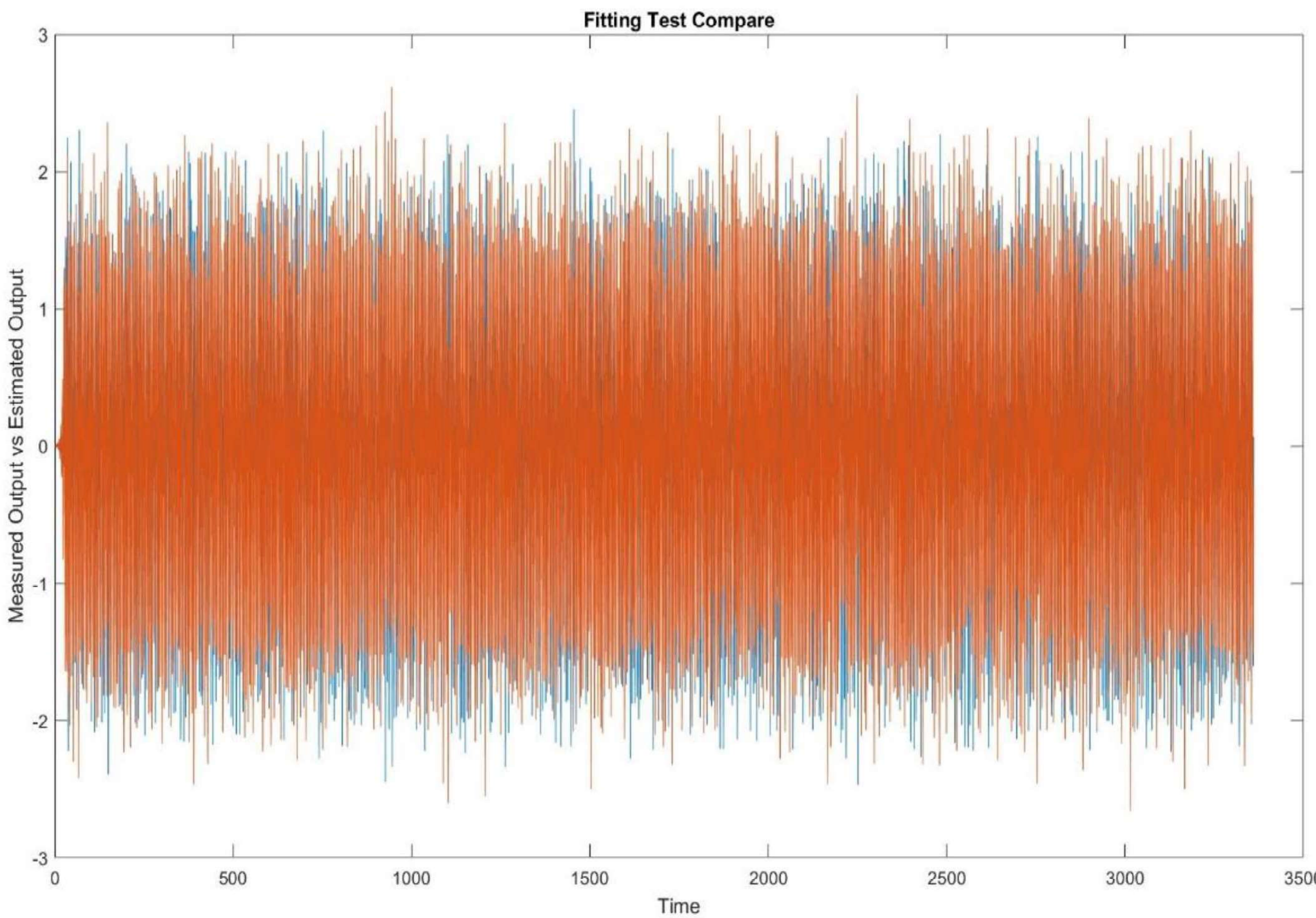
$$\hat{y}(k) = -a_1\, exp(y(k-1)) - a_2\, y(k-2) + b_1\, u(k-1) + b_2\, u(k-2) \qquad (5.13)$$

The result is shown below.

MSE= 0.3327 and Fitness= 0.4486

The obtained parameters

$$[\, a_1 \;\; a_2 \;\; b_1 \; b_2 \,] = [-0.1987 \quad 0.9097 \quad -0.3436 \quad -0.3689]$$



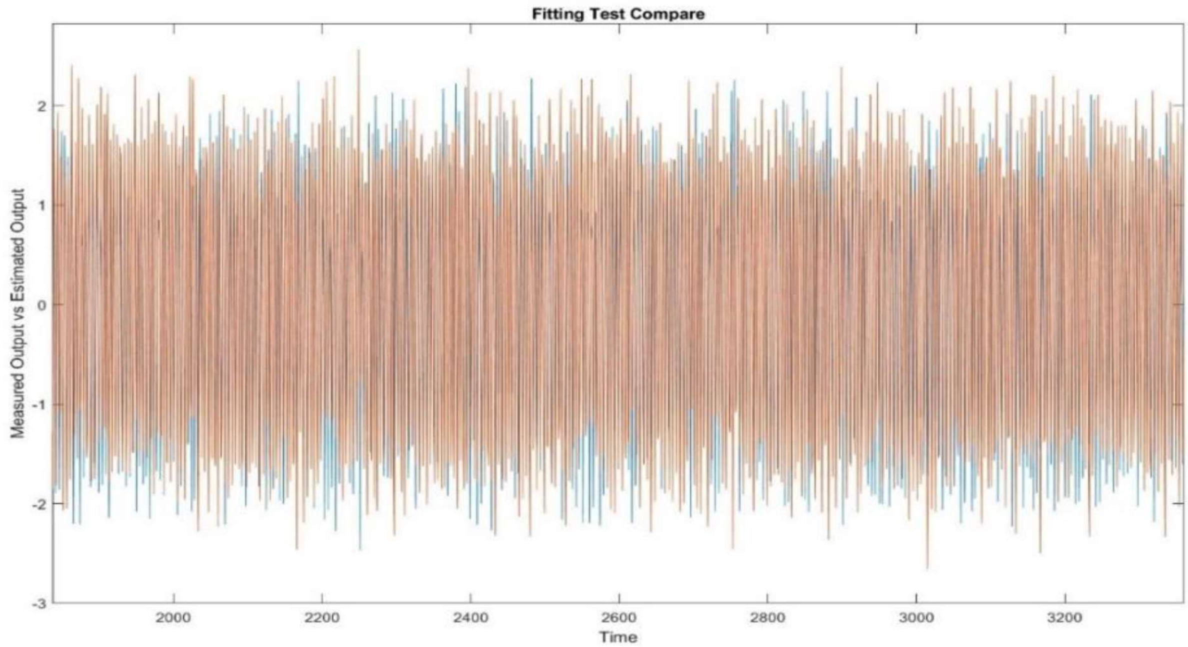


Figure 5.29: Second Case Study, Fitting Test Error for the Optimized Model No.2

The graph shows the general and zoomed simulation of comparison between the measured and estimated outputs.

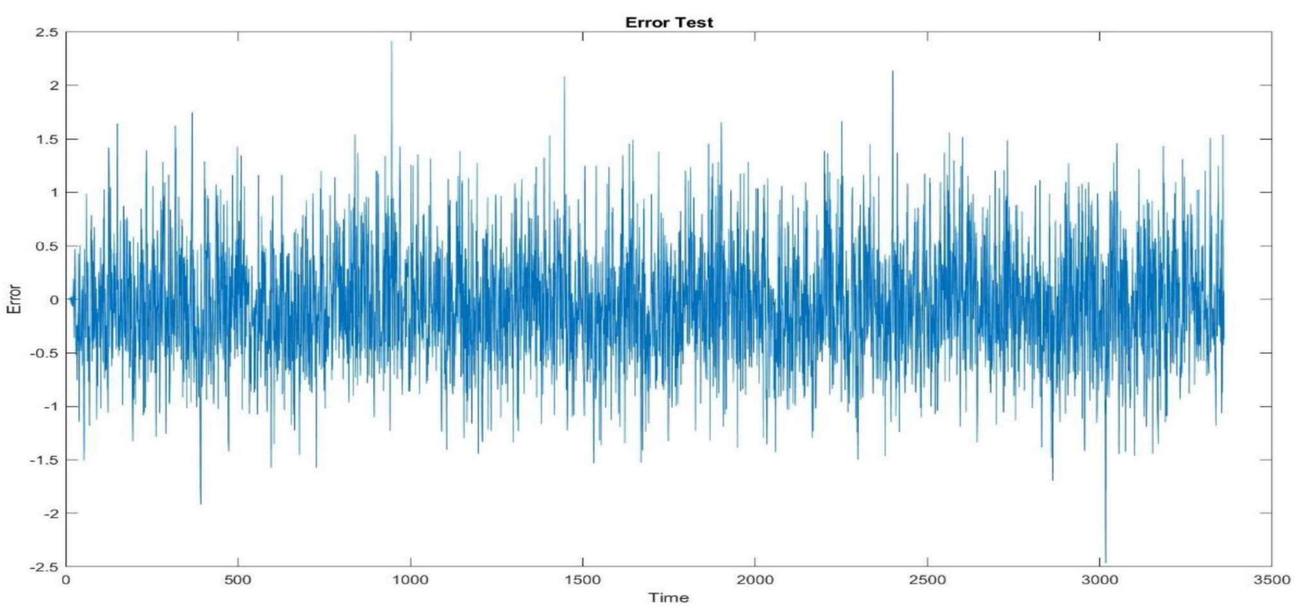


Figure 5.30: Second Case Study Error for the Optimized Model No.2

The graph shows the changes of the error over time.

5.4.2 Application Summary

The two optimization approach application models are tested with the same historical input-output data and following the same sequence. Table 5.4 is summarizing the mean square error and fitness of the tested models. The minimum MSE and fitness is the best accurate model and vice versa results.

Table 5.4: Second Case Classical Method Applications Results

Model No.	MSE	Fitness
1	0.2427	0.3831
2	0.3327	0.4486

The results show that first model have better accuracy of MSE 0.2427, Fitness 0.3831 better than the second one.

5.5 Second Case Study: Evolutionary Algorithm for System Identification Approach

On this part of the simulation, the evolutionary algorithm for the system identification is applied to search on wide ranges of the possible non-linear model structures and tested trough the same MSE and Fitness criteria. The historical input-output data is used of the second case study application.

Applying the Real Coded Genetic Algorithm for NARX System Identification

The obtained Genetic Optimized Model:

$$\hat{y}\,(k) = -a_1\,(y(k-1)) - a_2\,sin(y(k-2)) + b_1\,u(k-1) + b_2\,exp(u(k-2))$$

MSE= 0.1355 and Fitness= 0.2863

The obtained parameters

$$[\,a_1\;\; a_2\;\;\; b_1\; b_2\,] = [\;-0.5014\quad 1.5409\quad 0.0128\quad -0.0042]$$

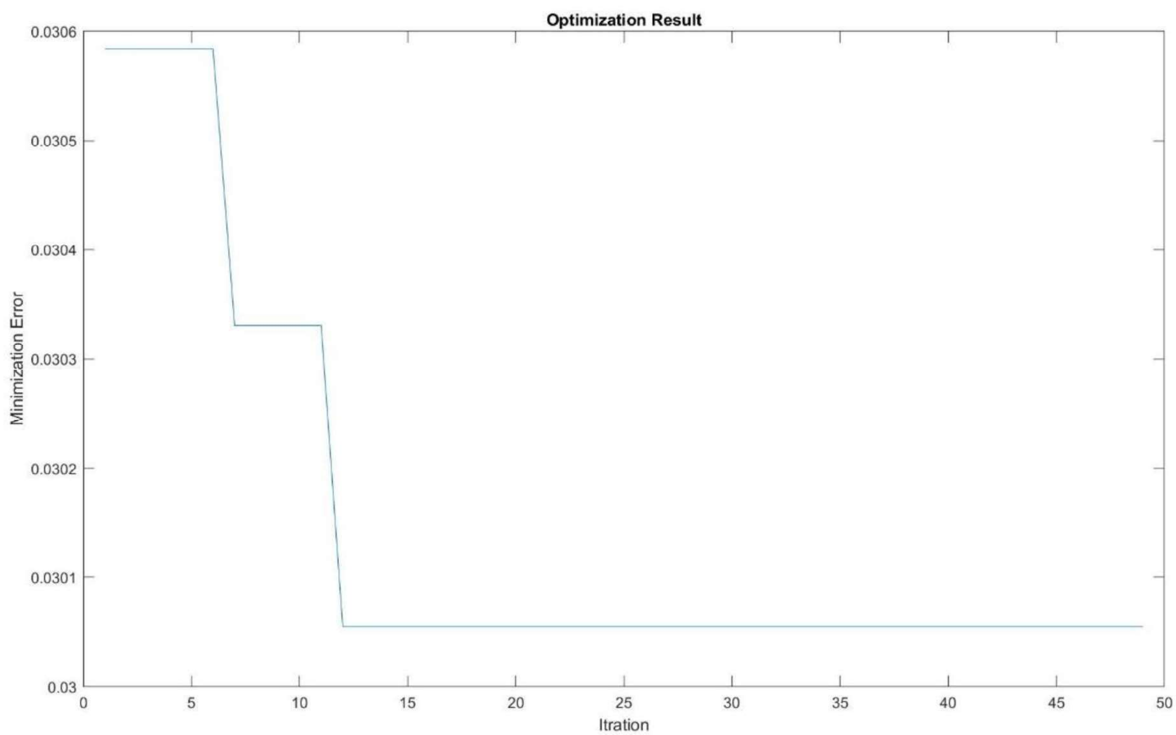
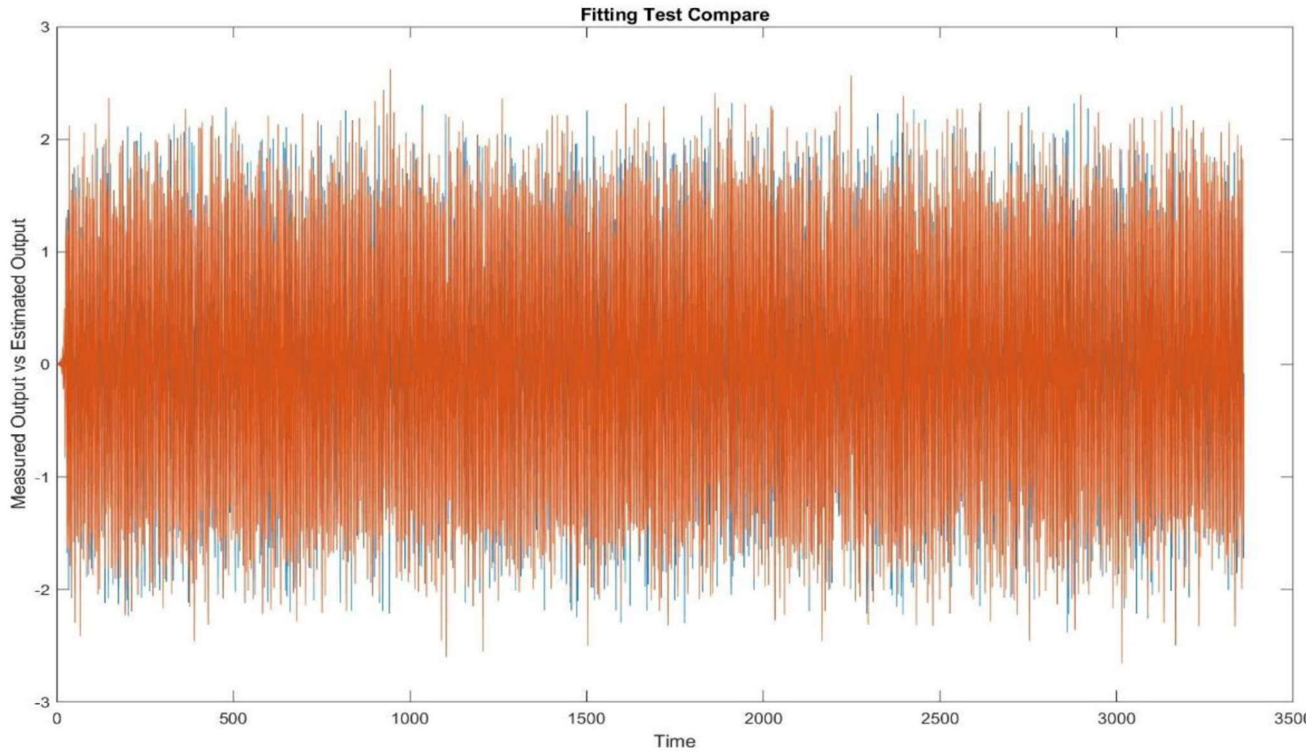


Figure 5.31: Mean Square Error Minimization for Second Case Study

The solution shows the first improvement at the seventh iteration and got the optimized result at the fifteenth iteration.



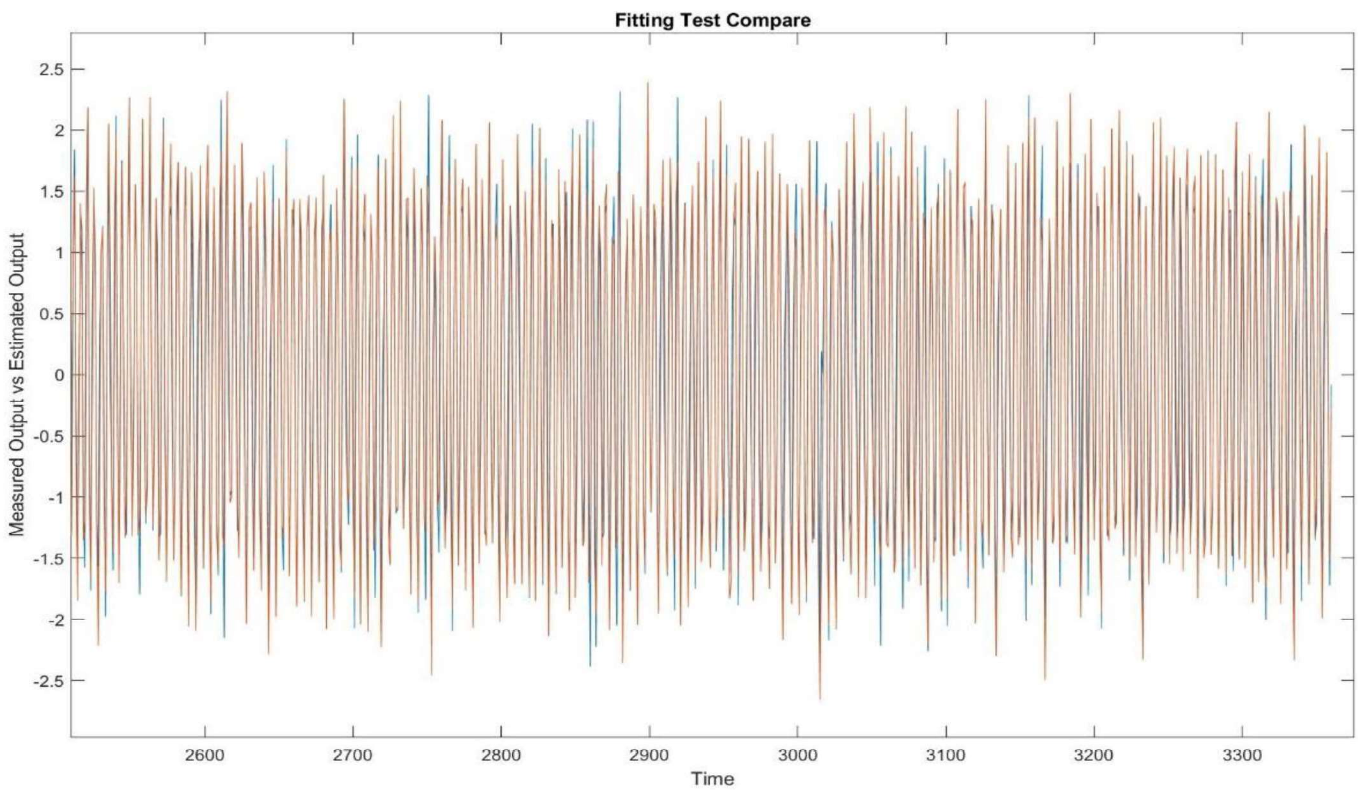


Figure 5.32: Fitting Test Compare for the Optimized Model Case Study 2

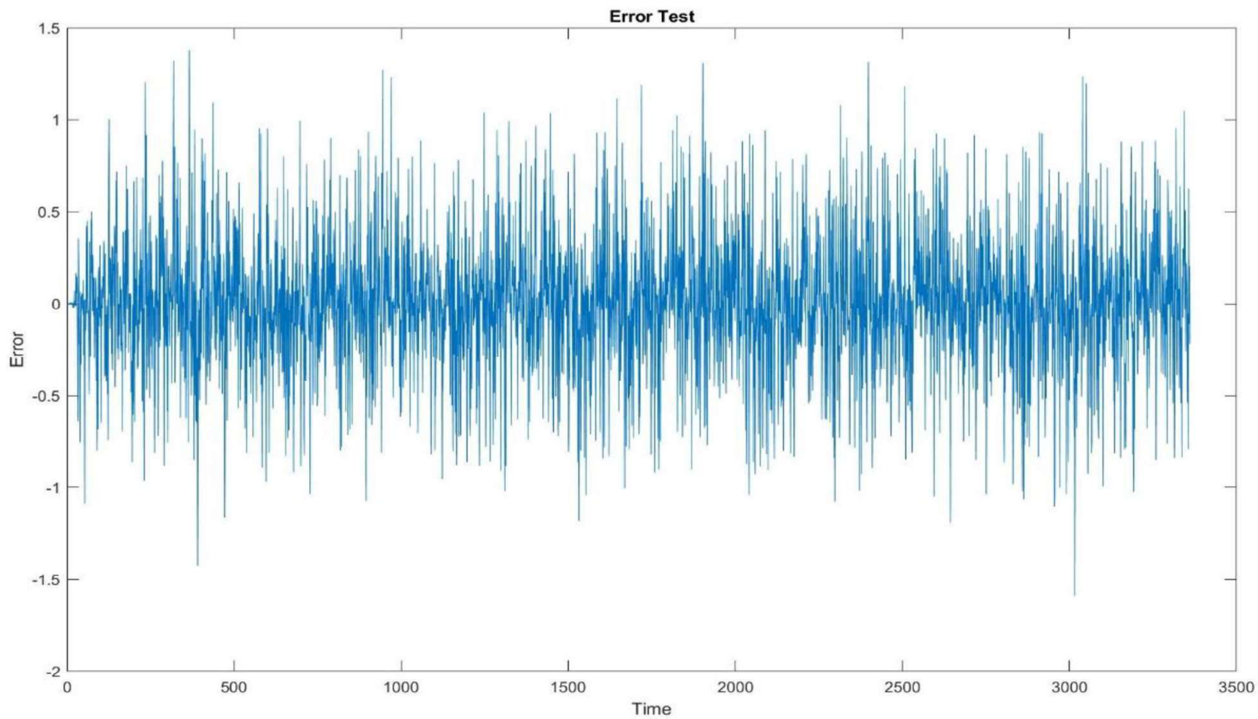


Figure 5.33: Mean Square Error for the Optimized Model Case Study 2

5.5.1 Second Case Study: Approach Summary

The results shows that the optimized model obtained by the genetic algorithm is the best in accuracy and have the minimum mean square error of 0.1355 over the two classical tested models.

5.6 Artificial Nonparametric Method Simulation

Here, we selected the NARX neural network model of the large number of forms and sizes to be trained on different trials and algorithms with the turbojet historical data. The data is divided into 70% for training, 15% for validation and 15% for

Testing. The first Levenberg-Marquardt algorithm requires more memory with less training time, and it is stopping automatically when there is no improvement on the mean square error. The second Bayesian Regularization algorithm consumes more training time but can absorb more generalization and noisy historical training data. The training time is stopping automatically when the adaptive weight is minimum. The third Scaled Conjugate Gradient algorithm requires less memory and stopping when there is no improving on the mean squared error on the validation period.

5.7 First Case: Artificial Neural Network Algorithm Trainings and Simulation

In this example, we use the turbojet historical data presented in Section 5.2.1. And applied the neural network fitting tool for different neurons trial with three trainings algorithms.

Table 5.5: NARX-ANN Performance in Levenberg-Marquardt Training Algorithm

Training Algorithm	Number of neurons in the hidden layer	Training MSE	Validation MSE	Testing MSE
Levenberg-Marquardt	50	4.00861 E-3	4.38177 E-3	4.15019 E-3
	100	4.03658 E-3	4.76044 E-3	4.47626 E-3
	250	3.62162 E-3	4.77040 E-3	4.59519 E-3

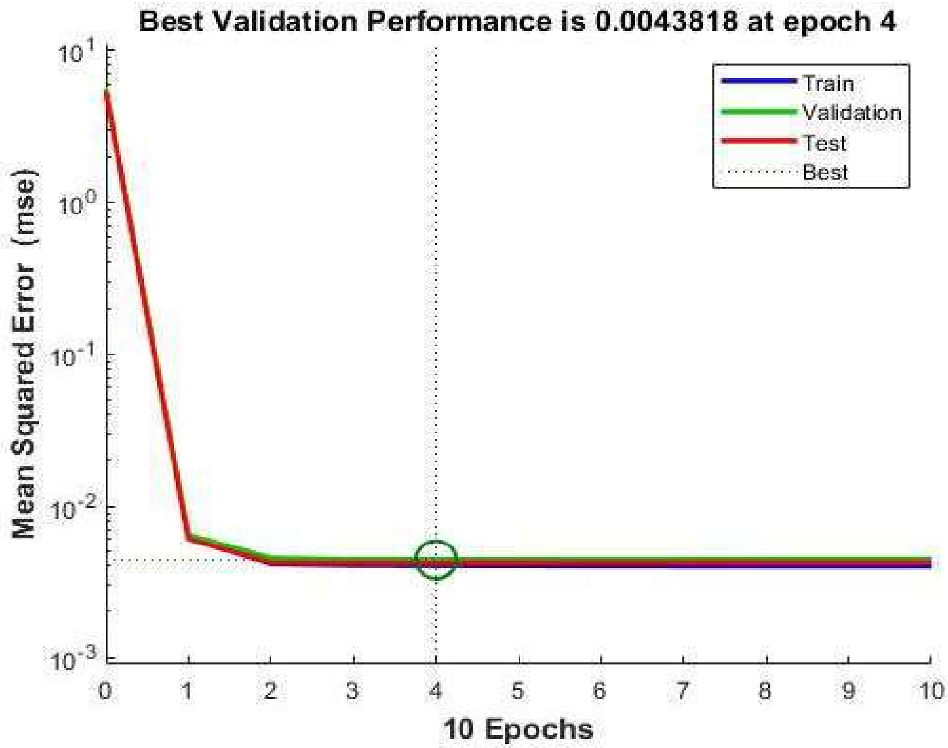


Figure 5.34: 50 Neurons Trial, Best Validation Performance

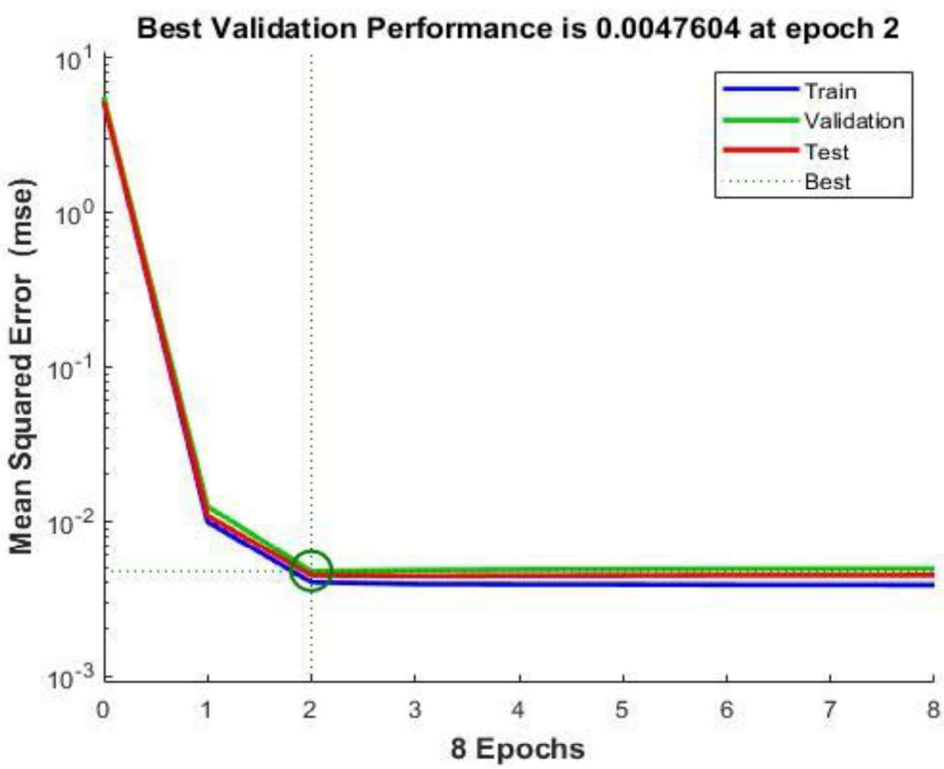


Figure 5.35: 100 Neurons Trial, Best Validation Performance

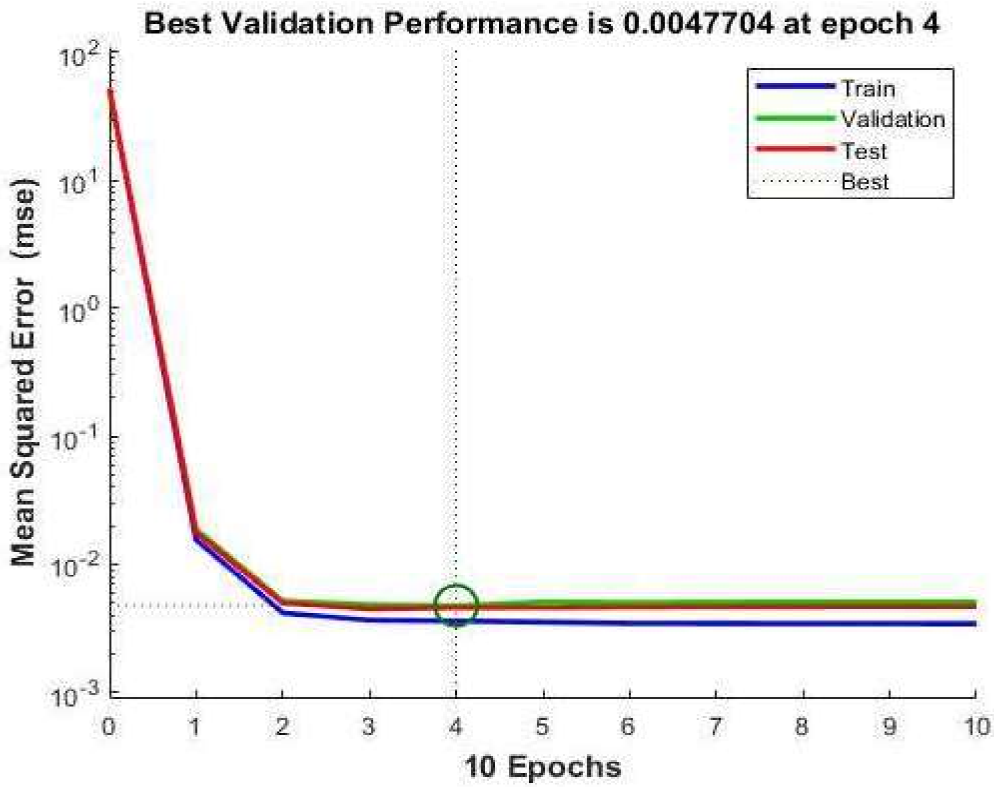


Figure 5.36: 250 Neurons Trial, Best Validation Performance

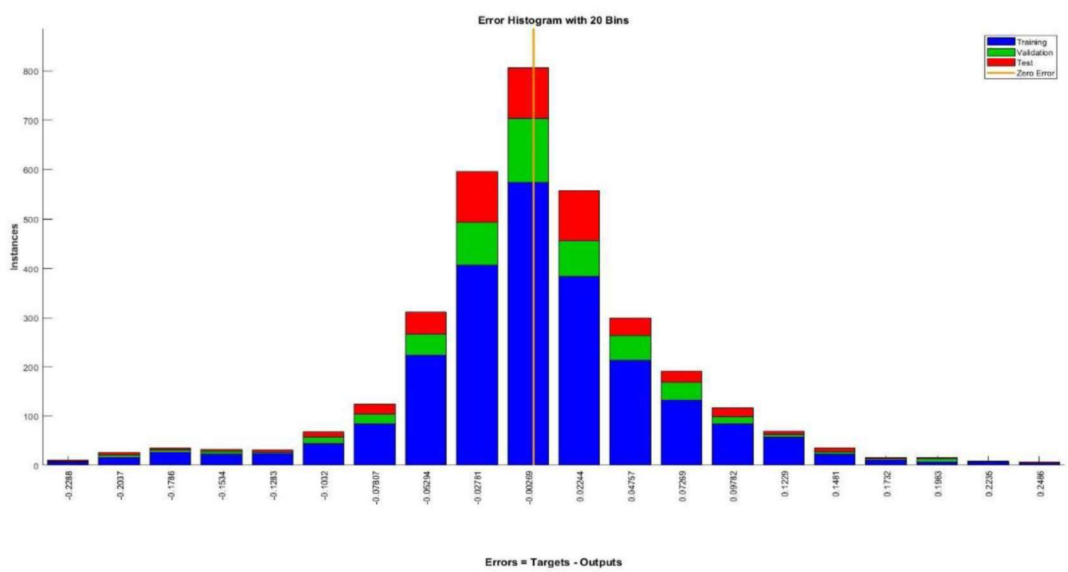


Figure 5.37: 50 Neurons Trial, Error Histogram

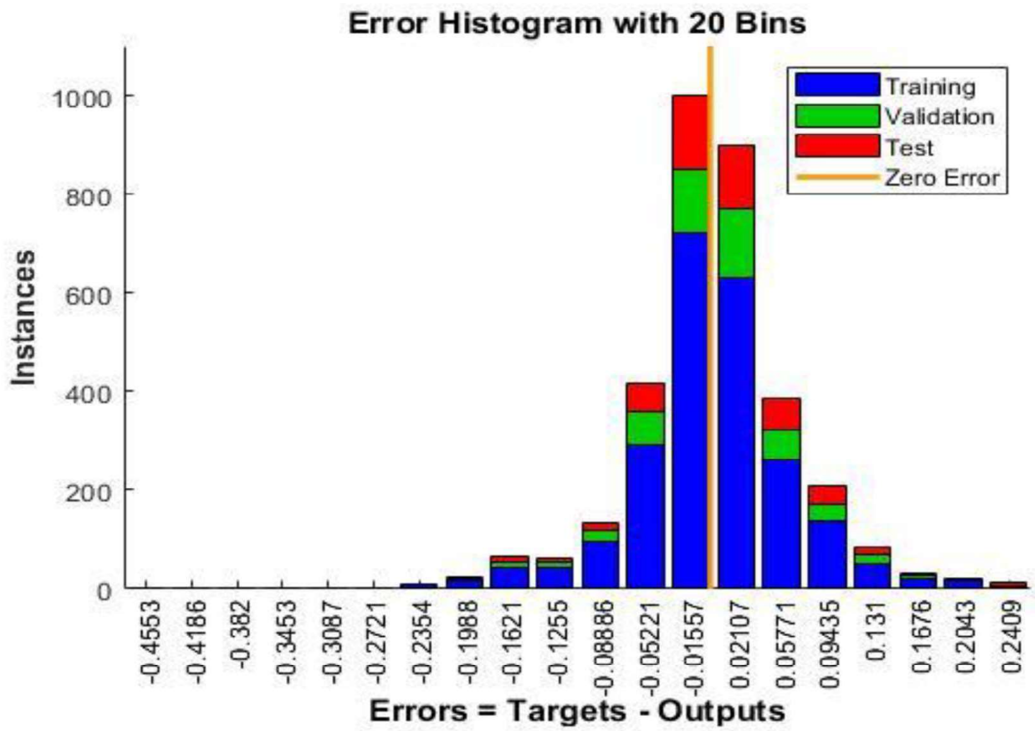


Figure 5.38: 100 Neurons Trial, Error Histogram

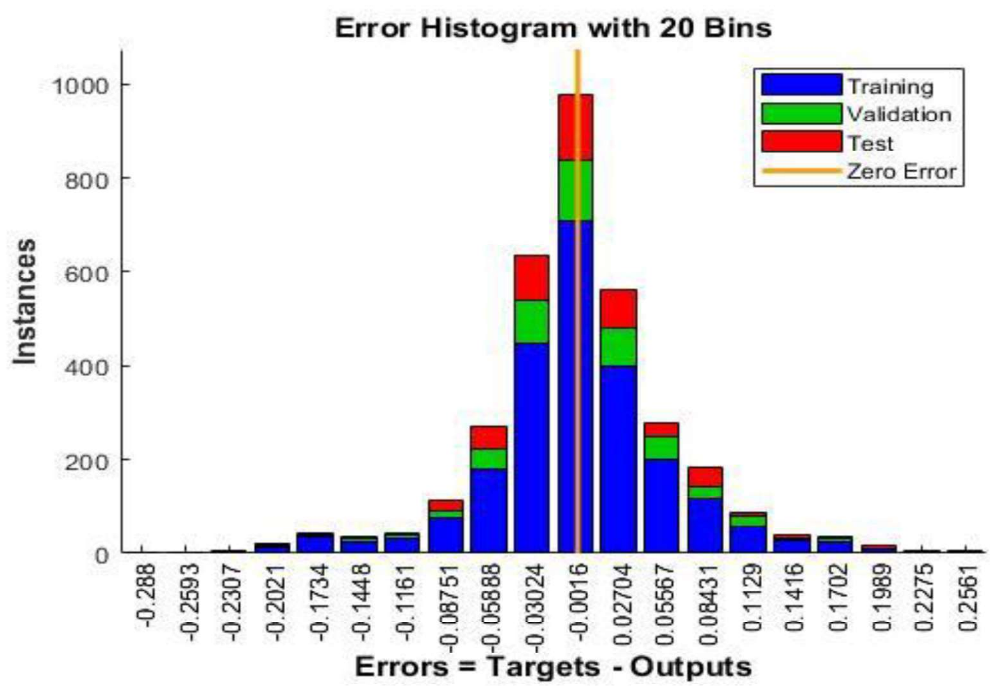


Figure 5.39: 250 Neurons Trial, Error Histogram

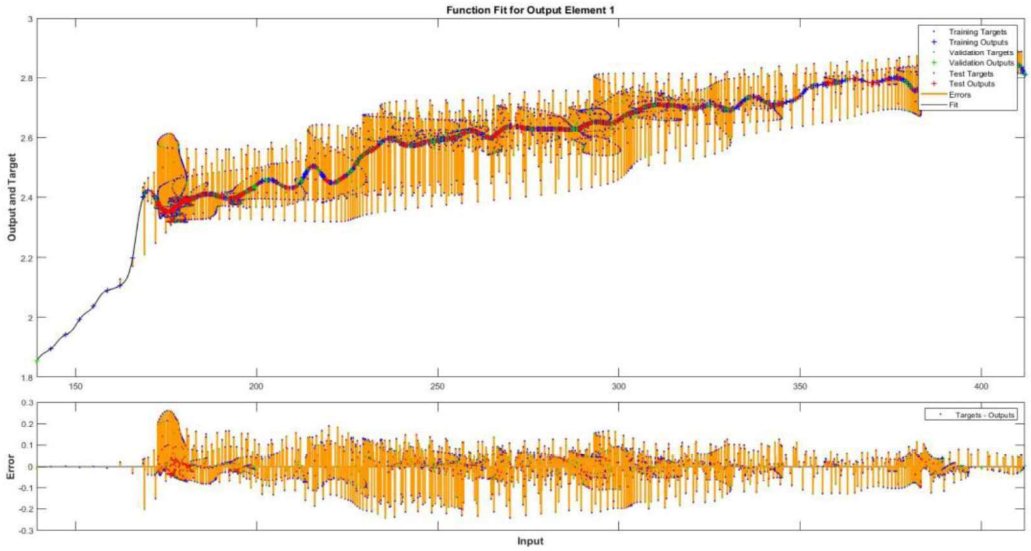


Figure 5.40: 50 Neurons Trial, Output Fit Plot

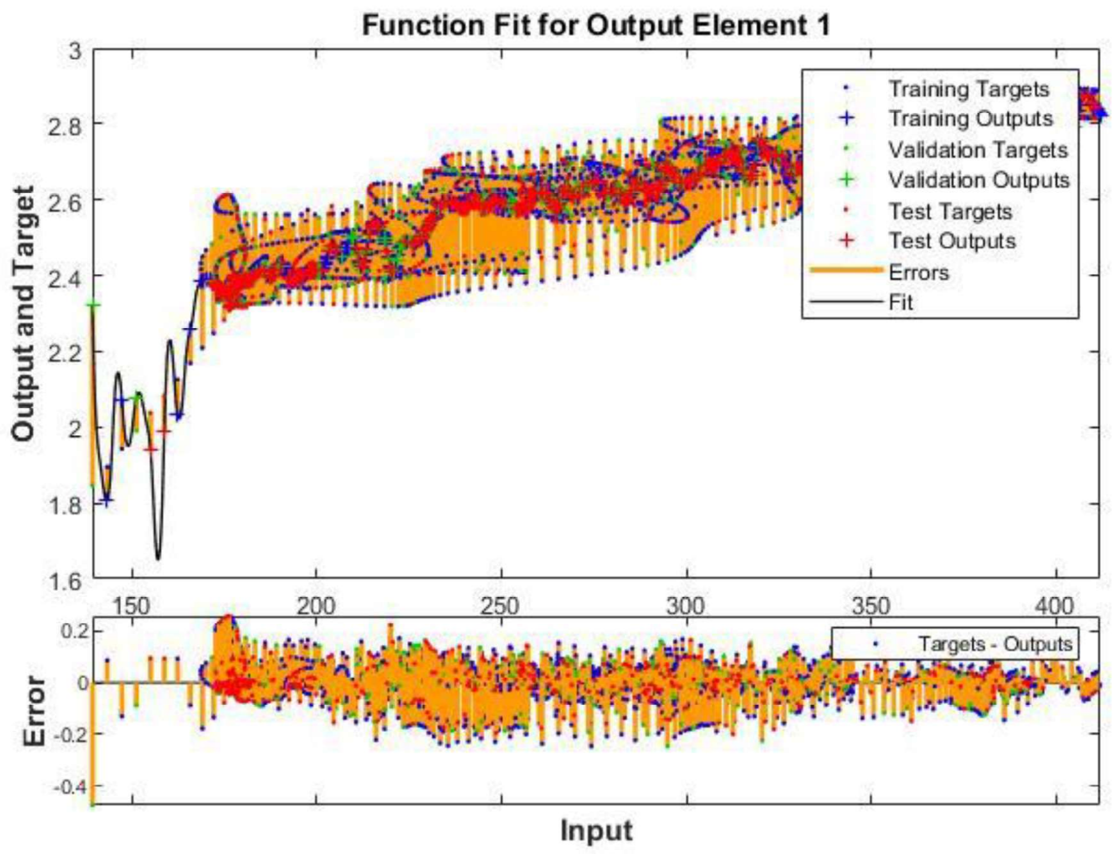


Figure 5.41: 100 Neurons Trial, Output Fit Plot

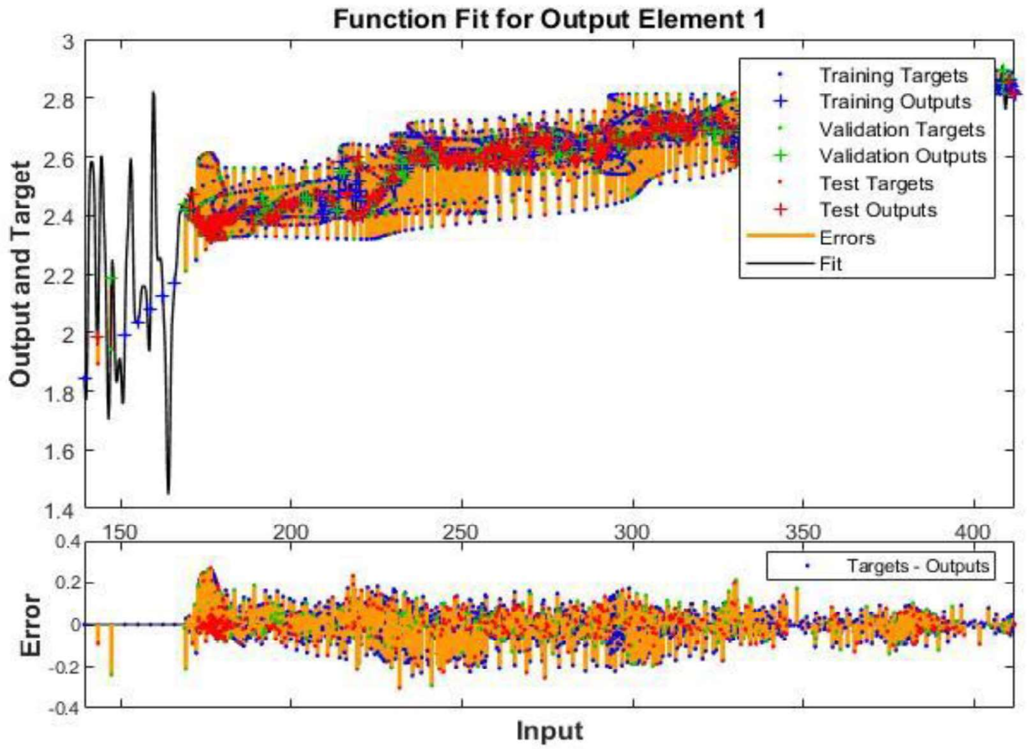


Figure 5.42: 250 Neurons Trial, Output Fit Plot

The figures show the best validation performance in MSE and includes the train and test performance during the epochs numbers. Each epochs represent one cycle of the full training and learning data. The number of epochs is a multiple machine learning trials for better convergence and generalization. The error histogram shows the errors for the training, validation, and tests with 20 intervals on the error range which represent the number of samples from data with each specific error number. The output fit plot shows the targeted output graph against the actual output for the training, validation, and test. And plotted the error range and changes during the input changes over time. The repeating figures represent the training trial on different number of neurons simultaneously as 50,100 and 250 neurons.

Table 5.6: NARX-ANN Performance in Bayesian-Regularization Training Algorithm

Training Algorithm	Number of neurons in the hidden layer	Training MSE	Validation MSE	Testing MSE
Bayesian-Regularization	50	4.13793 E-3	4.22371 E-3	4.07158 E-3
	100	3.97344 E-3	3.64617 E-3	4.57103 E-3
	250	3.75895 E-3	4.37121 E-3	4.64978 E-3

Table 5.7: NARX-ANN Performance in Scaled Conjugate Gradient Training Algorithm

Training Algorithm	Number of neurons in the hidden layer	Training MSE	Validation MSE	Testing MSE
Scaled Conjugate Gradient	50	4.156756 E-3	4.65900 E-3	5.14821 E-3
	100	4.29820 E-3	7.57437 E-3	6.65723 E-3
	250	5.31118 E-3	8.77837 E-3	9.31256 E-3

5.7.1 Performance Summary

Different algorithms trainings and number of neurons sizes were tried starting with a small number of neurons followed by higher numbers. The tables summarize the results on the obtained mean square error for all the trials and covering the mean square errors for the training, validation, and testing. The best training MSE is shown in the first training algorithm of the 250 neurons as (3.62162 E-3) while on the third training algorithm of the 250 neurons shows the worst as (5.31118 E-3). The validation MSE shows the best on the second training algorithm of 100 neurons as (3.64617 E-3) while the worst shown on the third training algorithm of 250 neurons as (8.77837 E-3). The Testing MSE shows the best on the second training algorithm of 50 neurons as (4.07158 E-3) while the worst shown on the third training algorithm of 250 neurons as (9.31256 E-3). Overall, the second training algorithm representing the best performance result in the neurons range from 50 to 100 over the validation and testing data.

5.8 Second Case: Artificial Neural Network Algorithm Trainings and Simulation

In this example, we use the proposed classical example historical data presented in Section 5.4. And applied the neural network fitting tool for different neurons trial with three trainings algorithms.

Table 5.8: NARX-ANN Performance in Levenberg-Marquardt Training Algorithm

Training Algorithm	Number of neurons in the hidden layer	Training MSE	Validation MSE	Testing MSE
Levenberg-Marquardt	50	1.61794	1.66219	1.78430
	100	1.61824	1.68076	1.64462
	250	1.56954	1.84836	1.91592

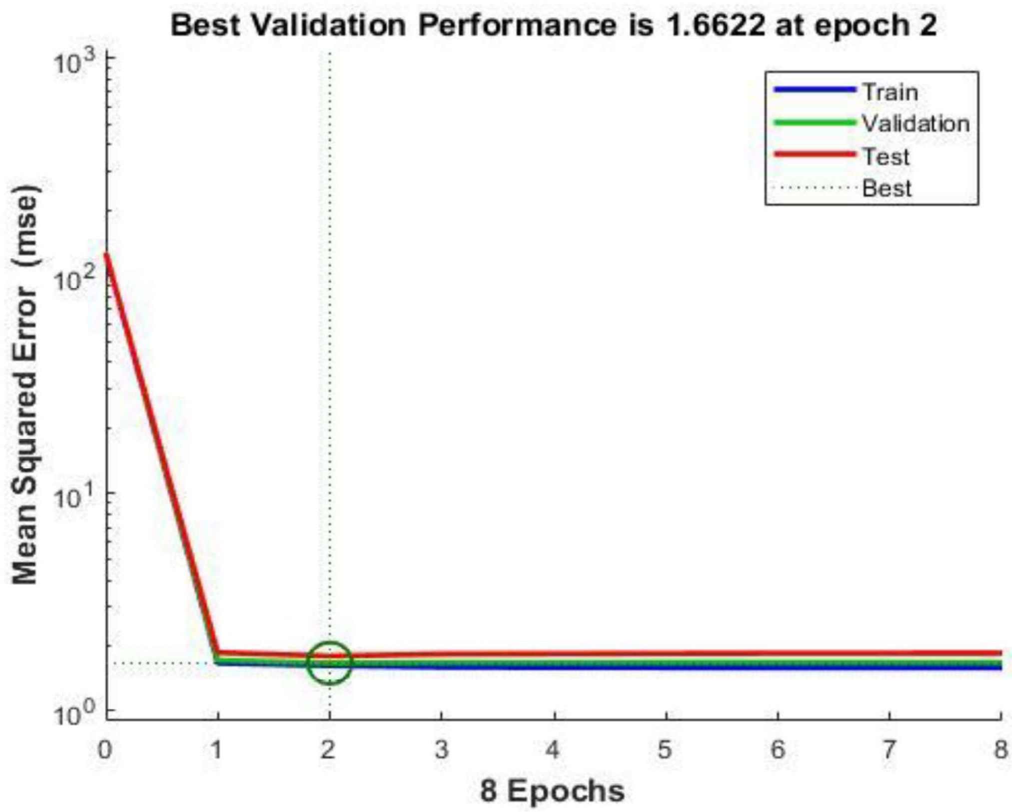


Figure 5.43: 50 Neurons Trial, Best Validation Performance

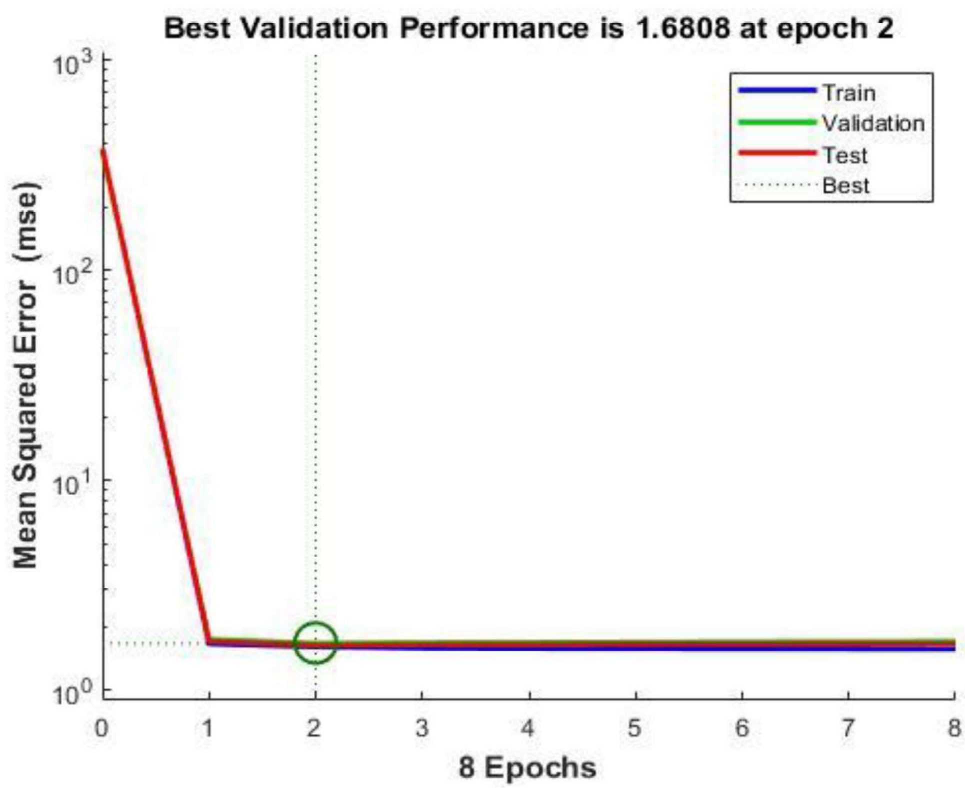


Figure 5.44: 100 Neurons Trial, Best Validation Performance

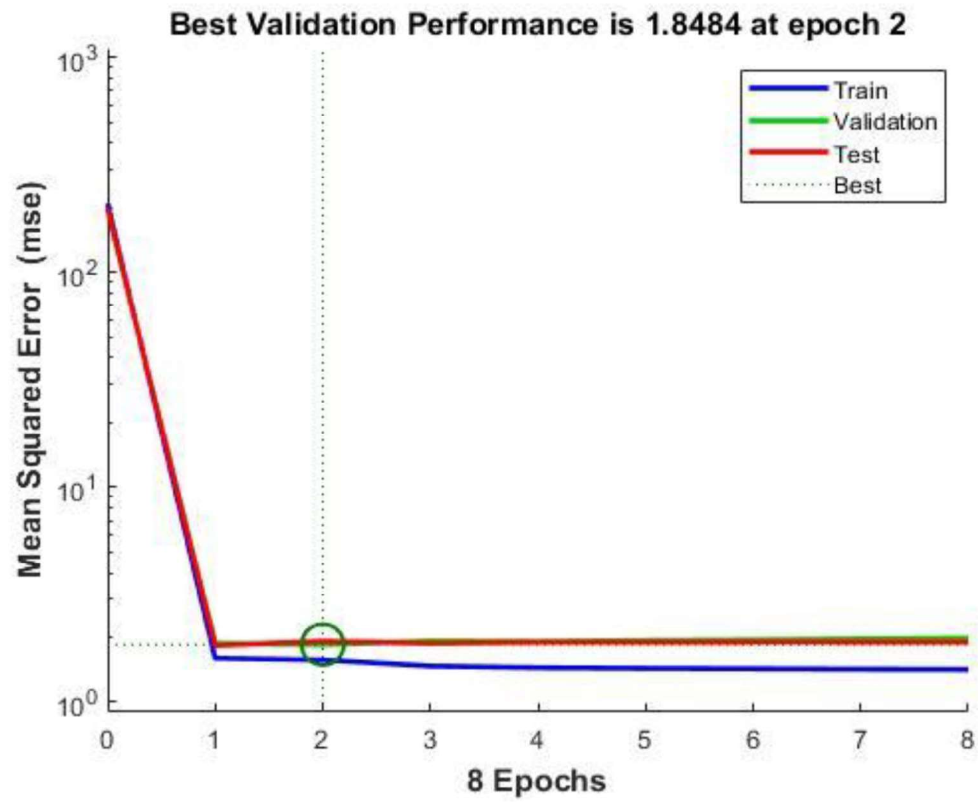


Figure 5.45: 250 Neurons Trial, Best Validation Performance

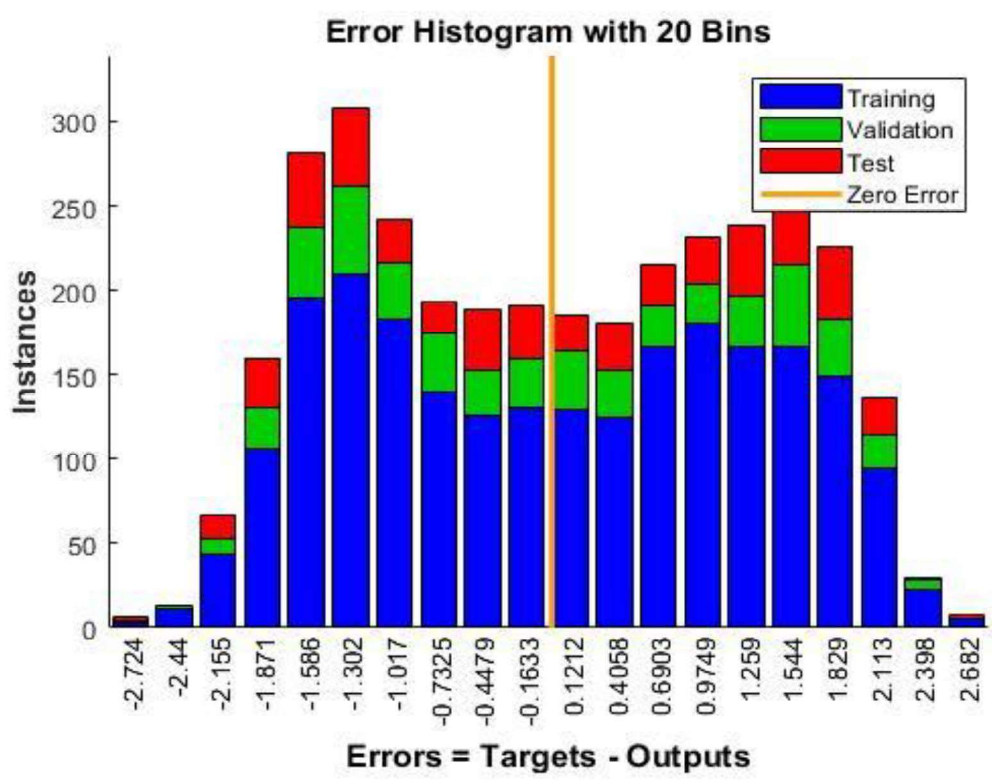


Figure 5.46: 50 Neurons Trial, Error Histogram

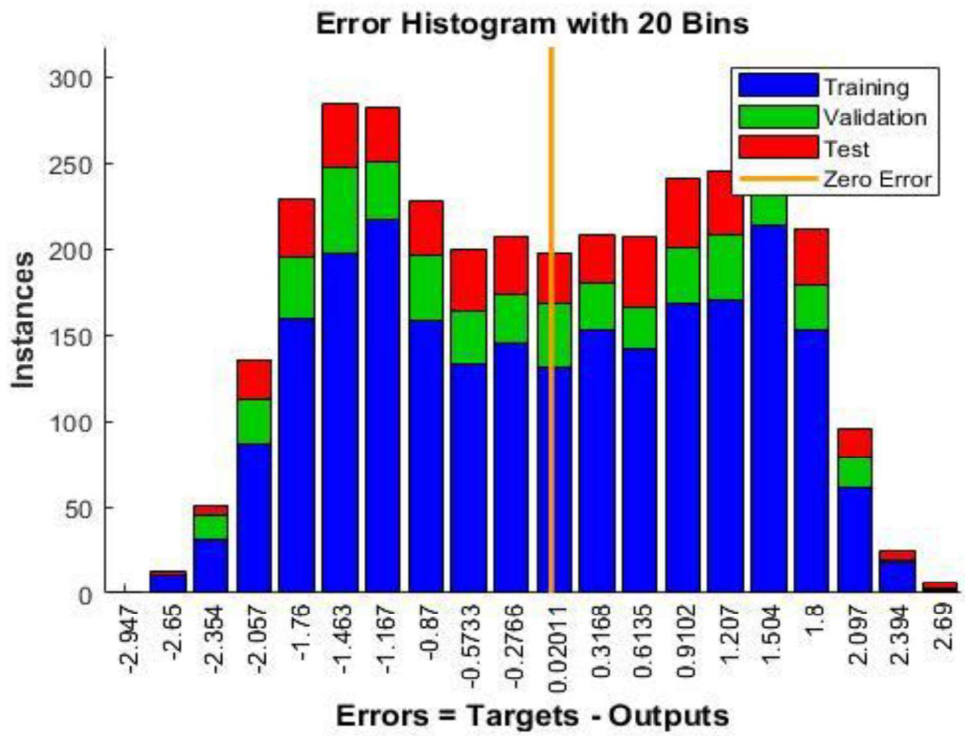


Figure 5.47: 100 Neurons Trial, Error Histogram

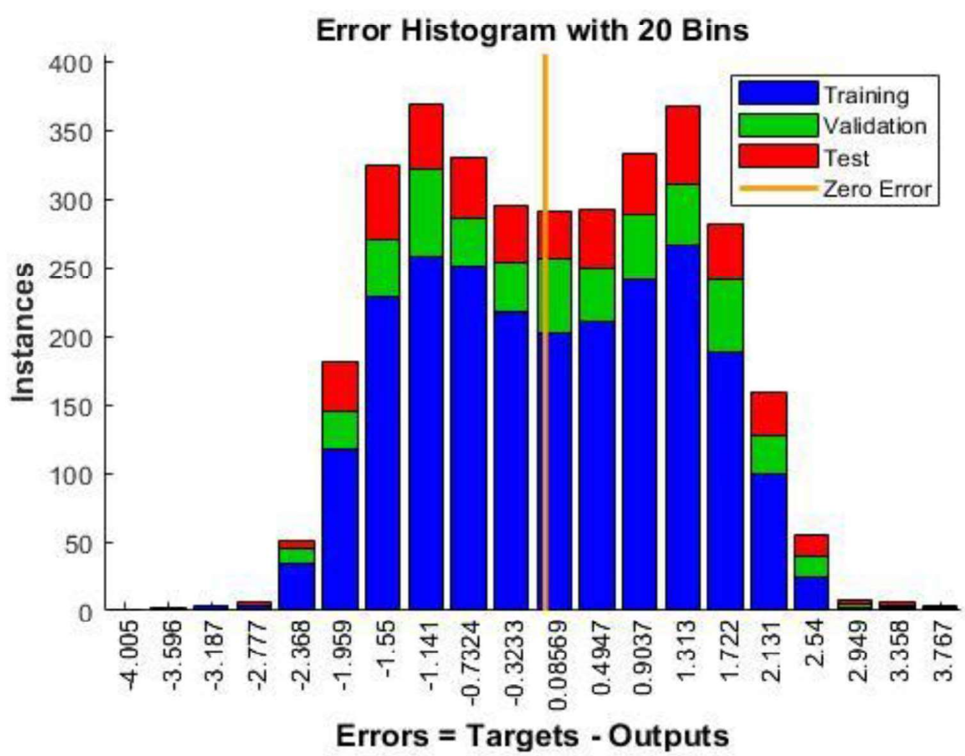


Figure 5.48: 250 Neurons Trial, Error Histogram

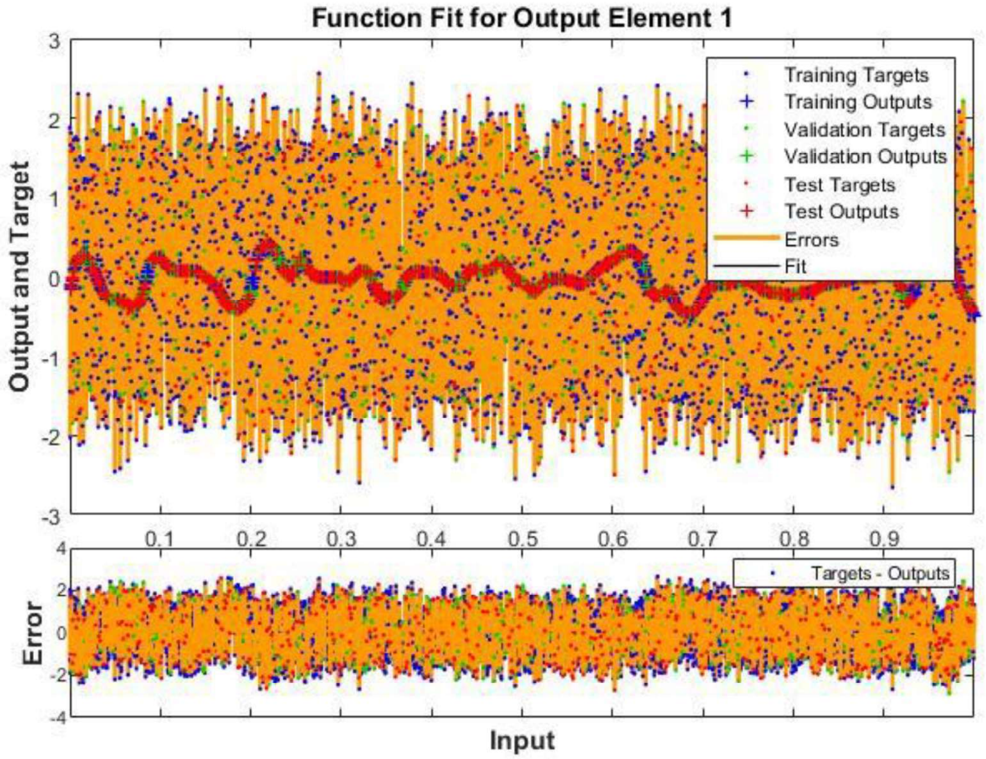


Figure 5.49: 50 Neurons Trial, Output Fit Plot

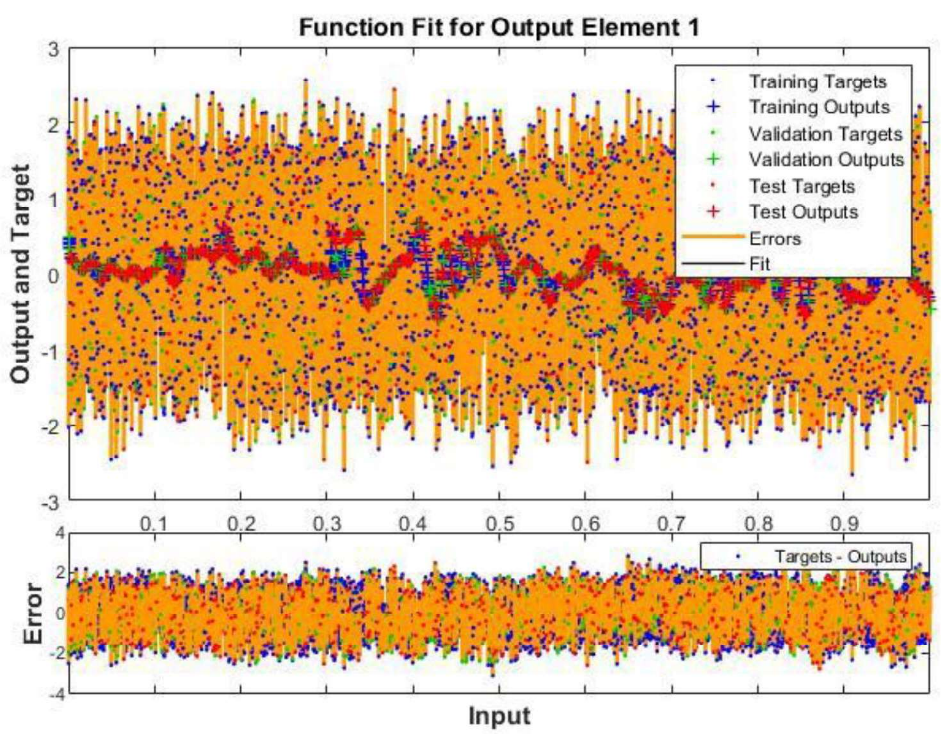


Figure 5.50: 100 Neurons Trial, Output Fit Plot

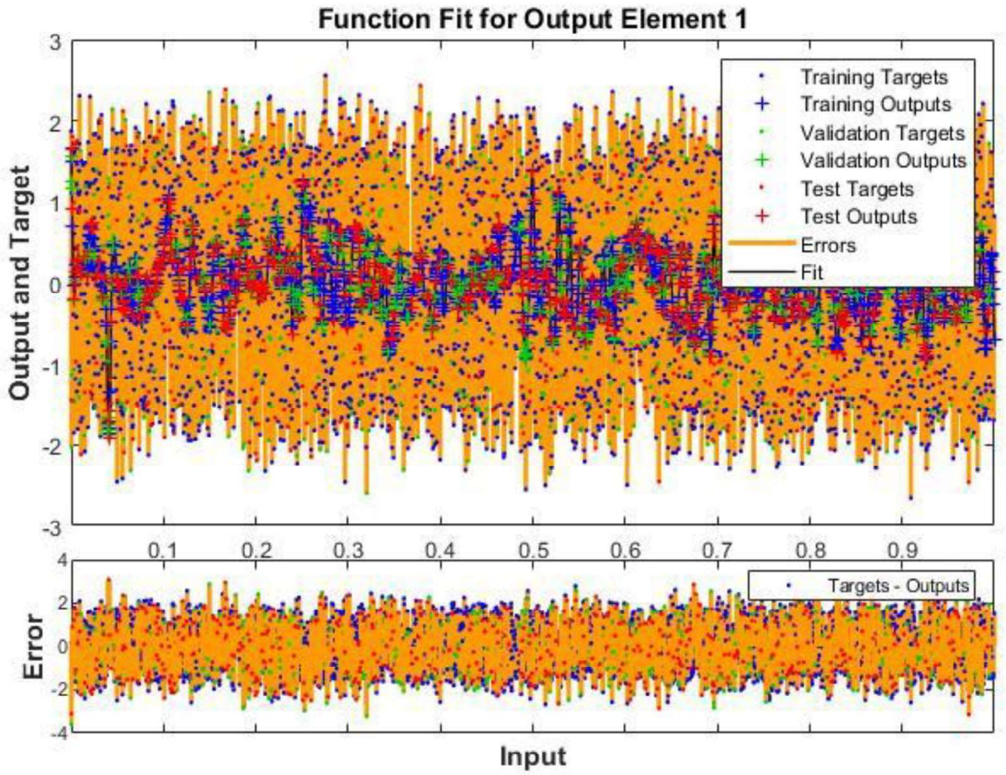


Figure 5.51: 250 Neurons Trial, Output Fit Plot

The figures are simulating the same steps and training algorithms for the second case study which described earlier on the first case.

Table 5.9: NARX-ANN Performance in Bayesian-Regularization Training Algorithm

Training Algorithm	Number of neurons in the hidden layer	Training MSE	Validation MSE	Testing MSE
Bayesian-Regularization	50	1.59970	0.00	1.71666
	100	1.55762	0.00	1.70362
	250	1.40826	0.00	2.02297

Table 5.10: NARX-ANN Performance in Scaled Conjugate Gradient Training Algorithm

Training Algorithm	Number of neurons in the hidden layer	Training MSE	Validation MSE	Testing MSE
Scaled Conjugate Gradient	50	1.62210	1.74649	1.70385
	100	1.58314	1.77509	1.66594
	250	1.62857	1.97388	1.96601

5.8.1 Performance Summary

The same algorism trainings and number of neurons sizes were tried starting with small number of neurons followed by higher numbers for second case similarly as the first case study steps. The tables summarize the results on the obtained mean squared error for all the trials and covering the mean square errors for the training, validation, and testing. The best training MSE is shown in the second training algorithm of the 250 neurons as (1.40826) while on the third training algorithm of the 250 neurons shows the worst as (1.62857). The validation MSE shows the best on the second training algorithm of 50,100 and 250 neurons as (0.00) while the worst shown on the third training algorithm of 250 neurons as (1.97388). The Testing MSE shows the best on the first training algorithm of 100 neurons as (1.64462) while the worst shown on the second training algorithm of 250 neurons as (2.02297). Overall, the second training algorithm representing the best performance result in the neurons range from 50 to 100 over the training and validation data.

5.9 Evolutionary Algorithms for Identification of Hammerstein Models

In this section, we apply the genetic Algorithm to identify a Hammerstein model. The system being considered is assumed to consist of two blocks, a nonlinear static subsystem followed by a linear time-invariant subsystem. The system is shown in Figure 5.52

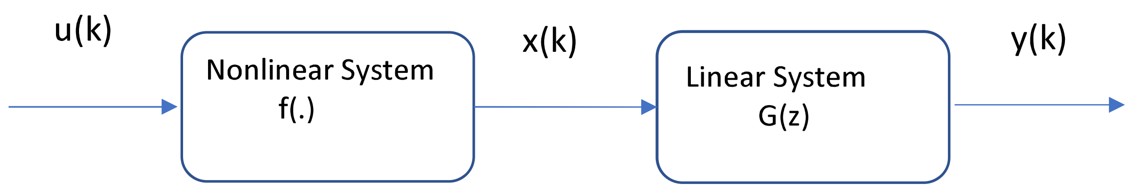


Figure 5.52 Hammerstein System

In this example the nonlinear block is defined by

$$x(k) = \alpha_1\, u(k) + \alpha_2\, u^2(k) + \alpha_3\, u^3(k) \qquad (5.14)$$

The linear block is defined as

$$y(k) = a_1\, y(k-1) + a_2\, y(k-2) + x(k-1) + b_2\, x(k-2) \qquad (5.14)$$

The true values of the parameters are

$$[\alpha_1 ; \alpha_2\ ;\ \alpha_3\ ;\ a_1 ; a_2\ ;\ b_2\,] = [\ \ 1.2,\ -0.56,\ 0.52,\ -1,\ 0.18,\ -0.3\ ]$$

First Trial:

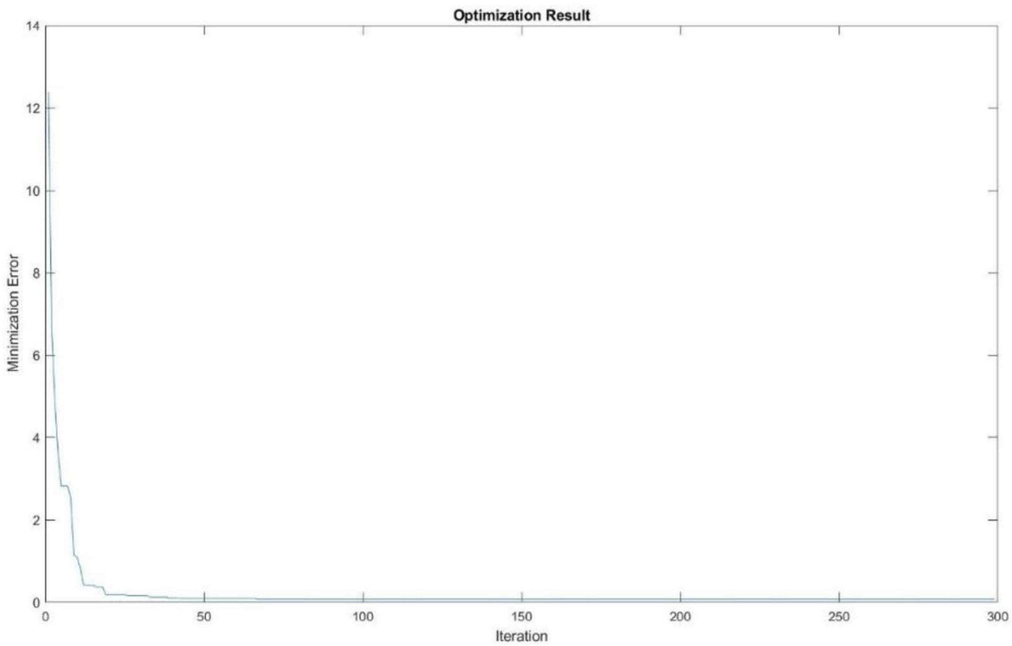


Figure 5.53: Mean Square Error Minimization for First Hammerstein Solution Trial

MSE= 0.0710
Parameters= [0.6349 -0.5258 0.5510 -1.2320 0.3581 -0.4940]

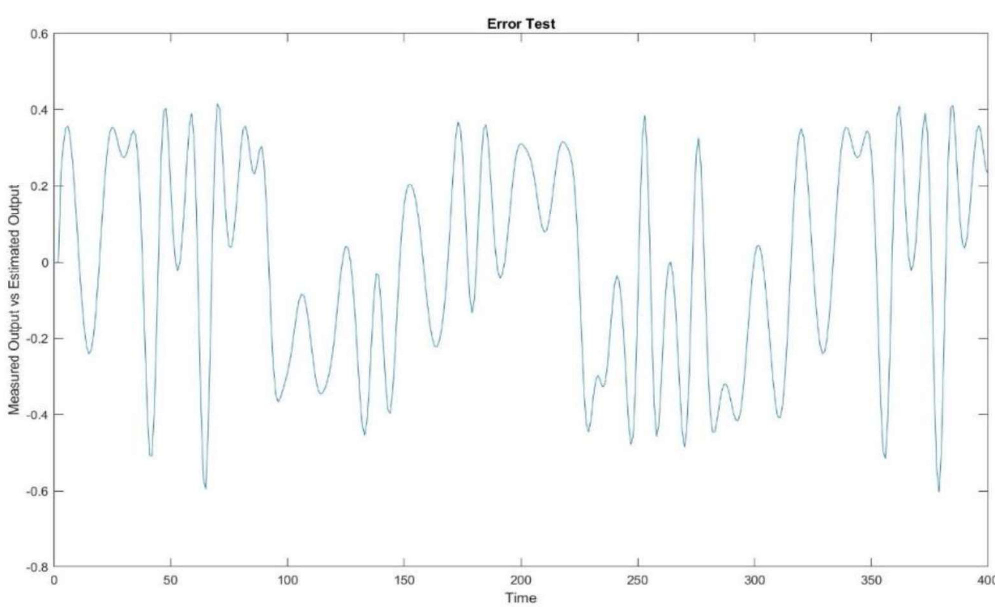


Figure 5.54: Measured Compared with Estimated Output for First Hammerstein Solution Trial

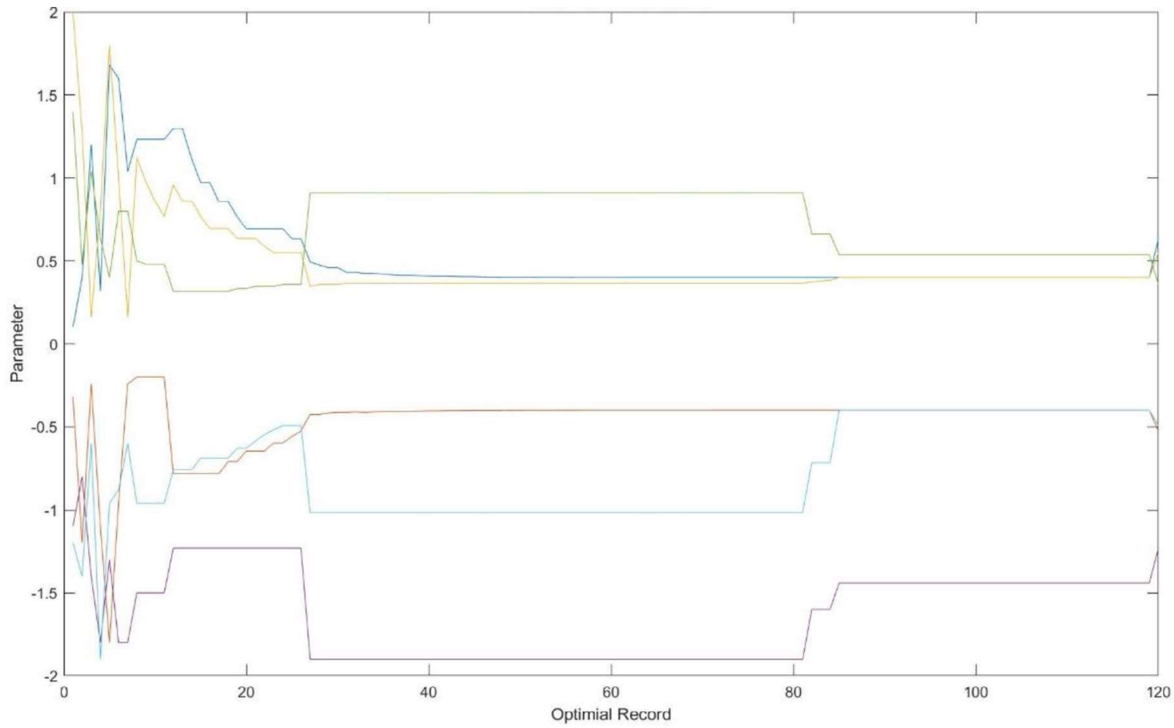


Figure 5.55: Parameters Convergence for First Hammerstein Solution Trial

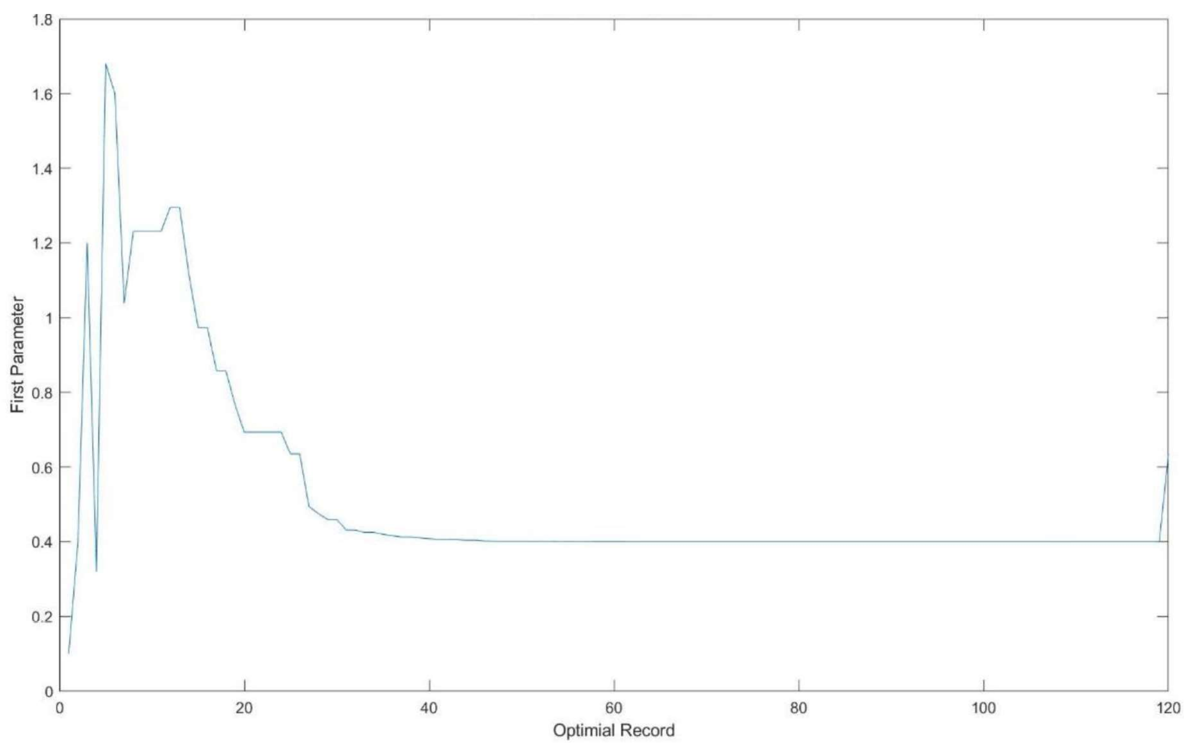


Figure 5.56: First Parameter Convergence for First Hammerstein Solution Trial

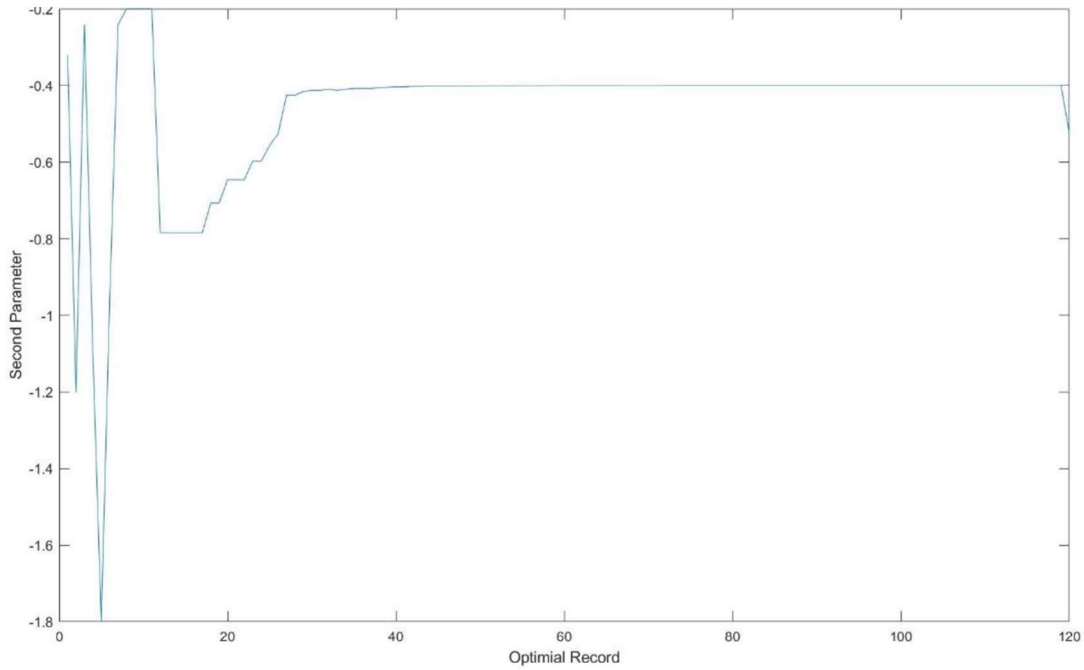


Figure 5.57: Second Parameter Convergence for First Hammerstein Solution Trial

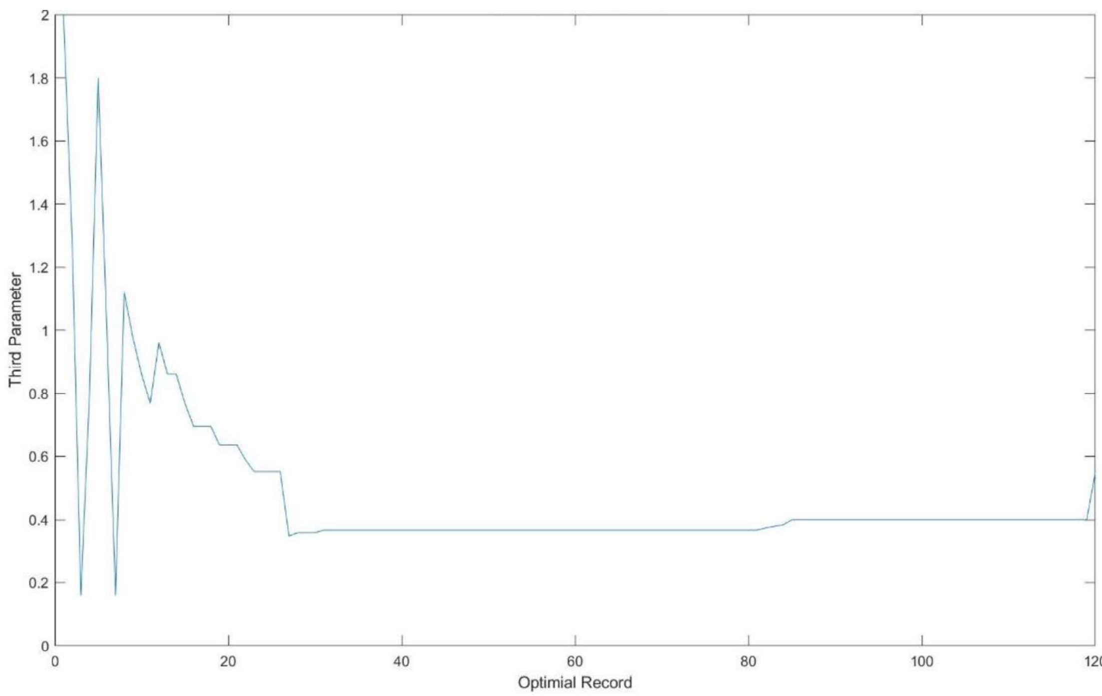


Figure 5.58: Third Parameter Convergence for First Hammerstein Solution Trial

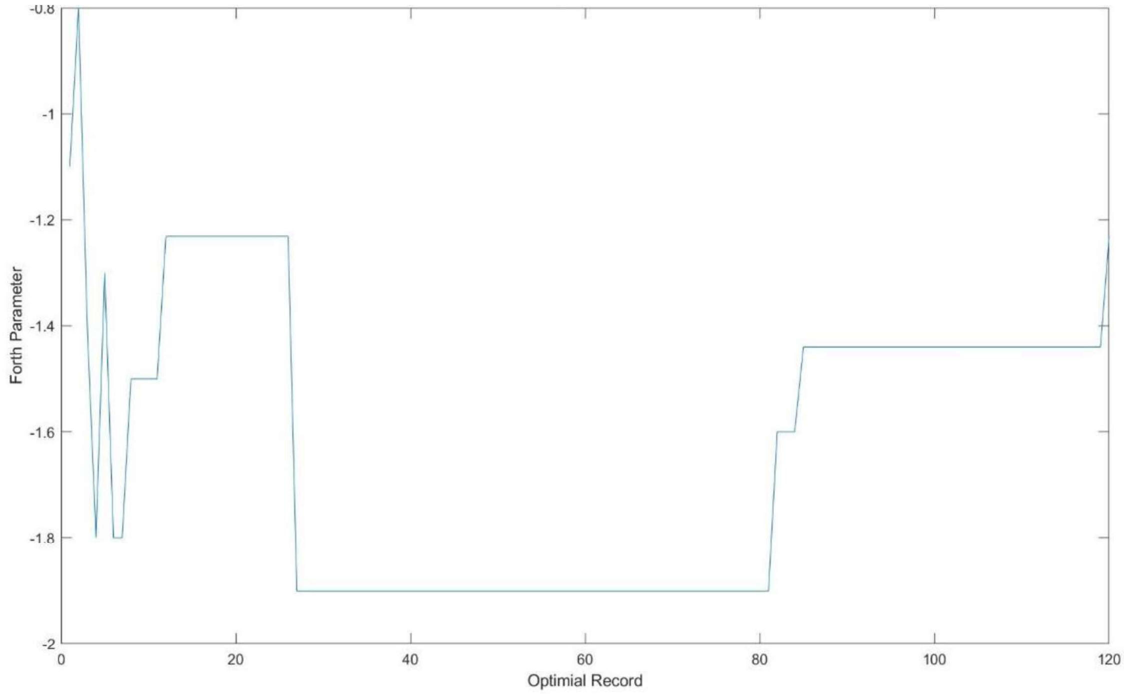


Figure 5.59: Forth Parameter Convergence for First Hammerstein Solution Trial

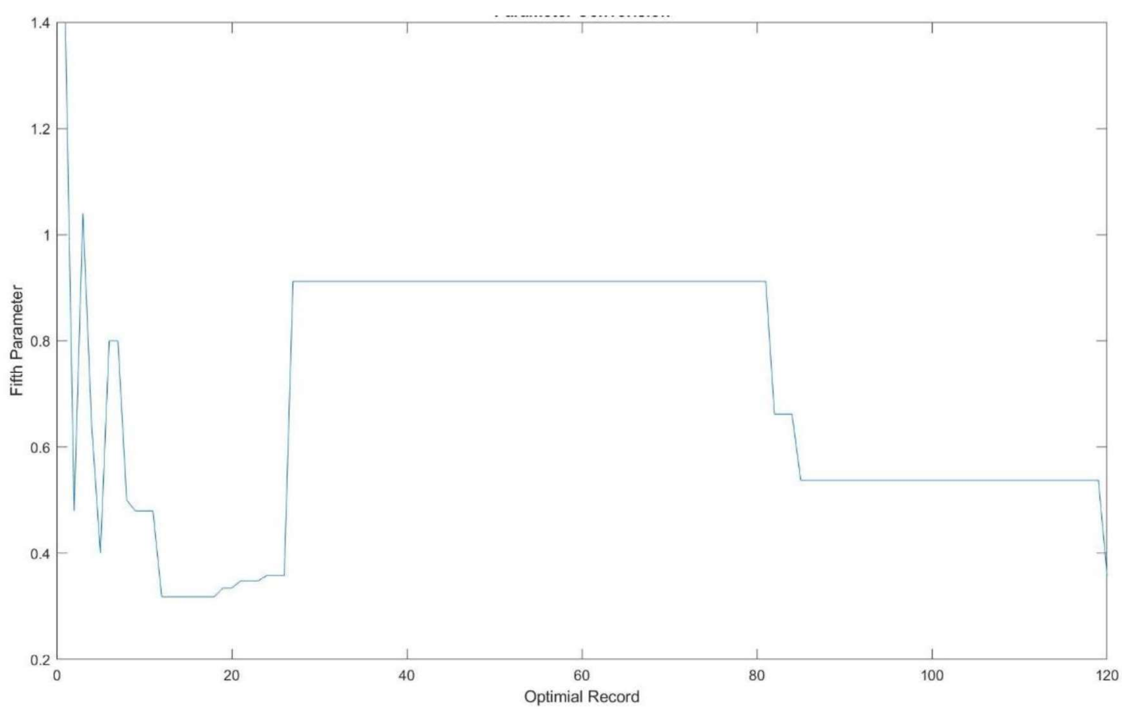


Figure 5.60: Fifth Parameter Convergence for First Hammerstein Solution Trial

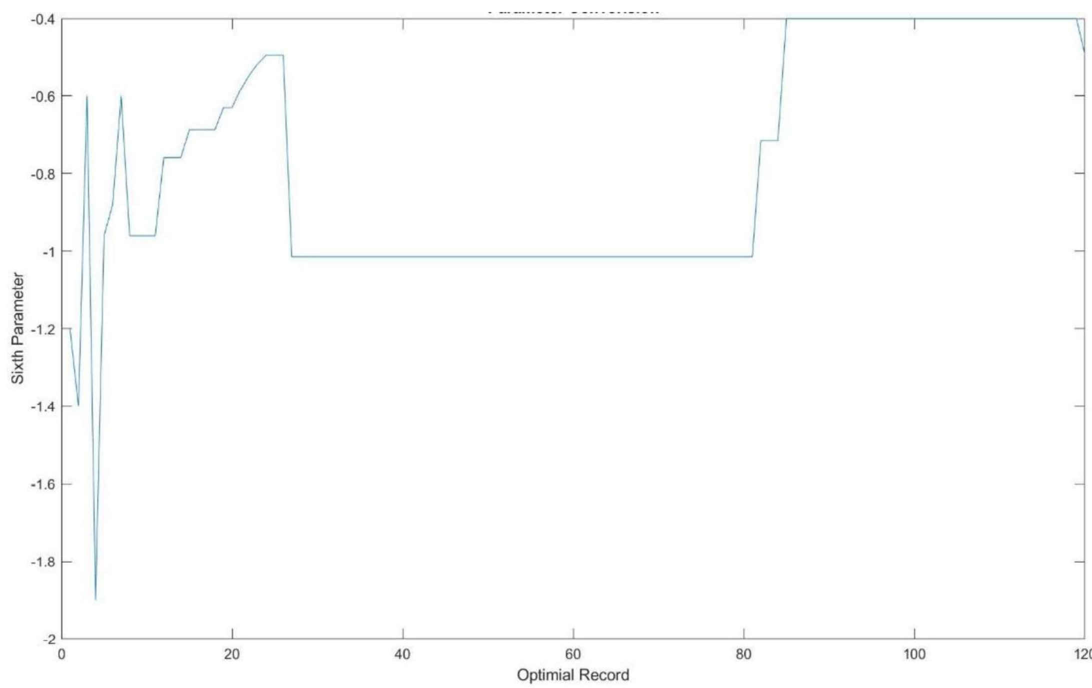


Figure 5.61: Sixth Parameter Convergence for First Hammerstein Solution Trial

Second Trial:

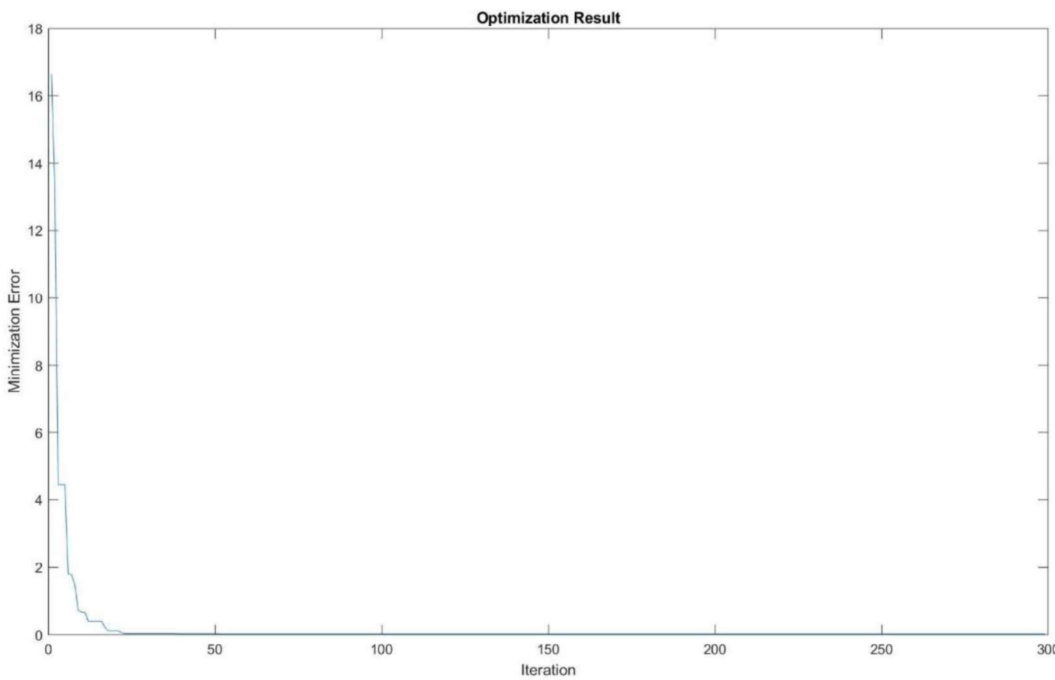


Figure 5.62: Mean Square Error Minimization for Second Hammerstein Solution Trial

MSE= 0.0316
Parameters= [0.9325 -0.5288 0.5049 -1.5000 0.5536 -0.7686]

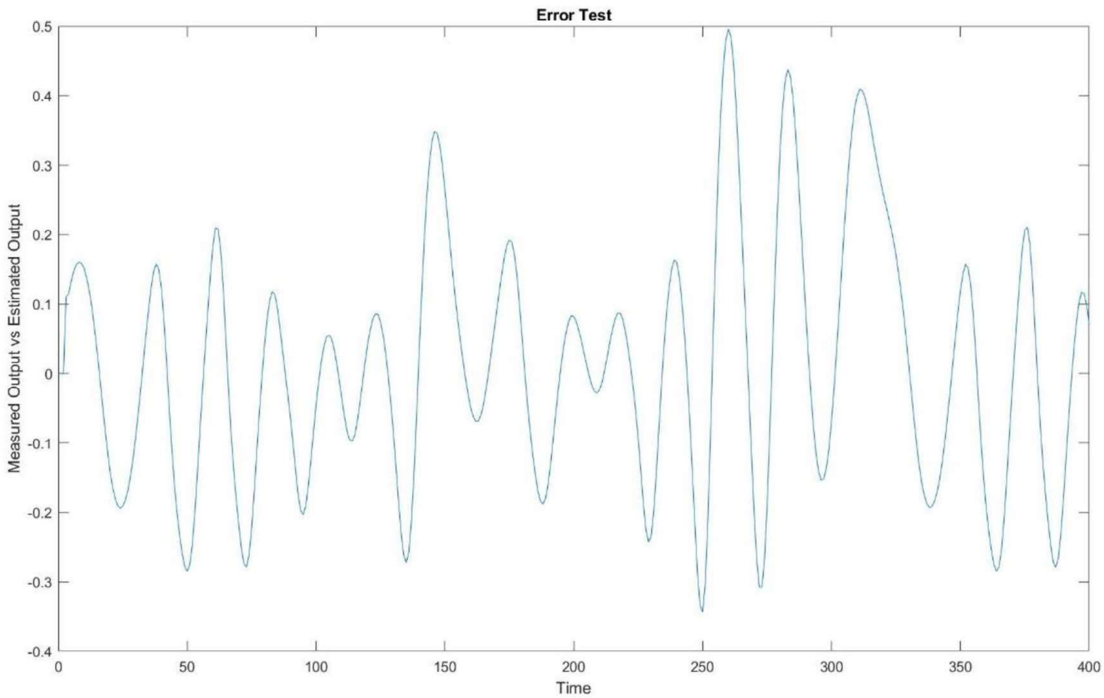


Figure 5.63: Measured Compared with Estimated Output for Second Hammerstein Solution Trial

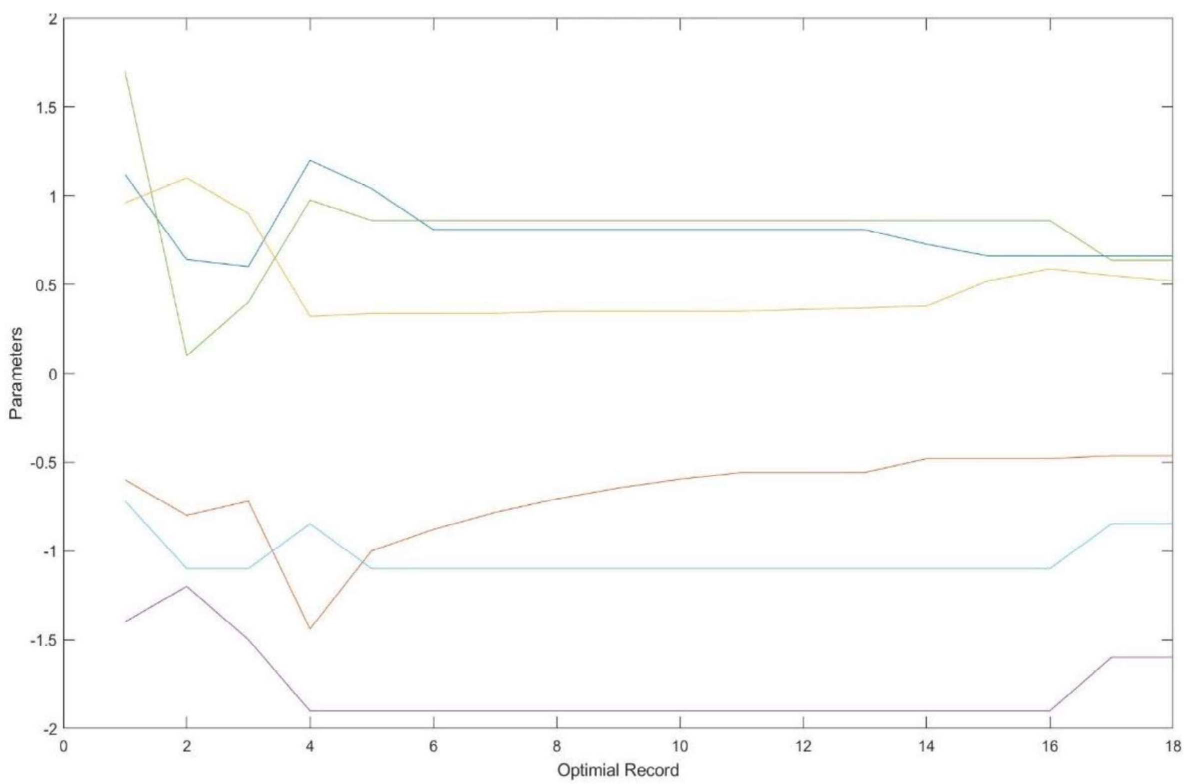


Figure 5.64: Parameters Convergence for Second Hammerstein Solution Trial

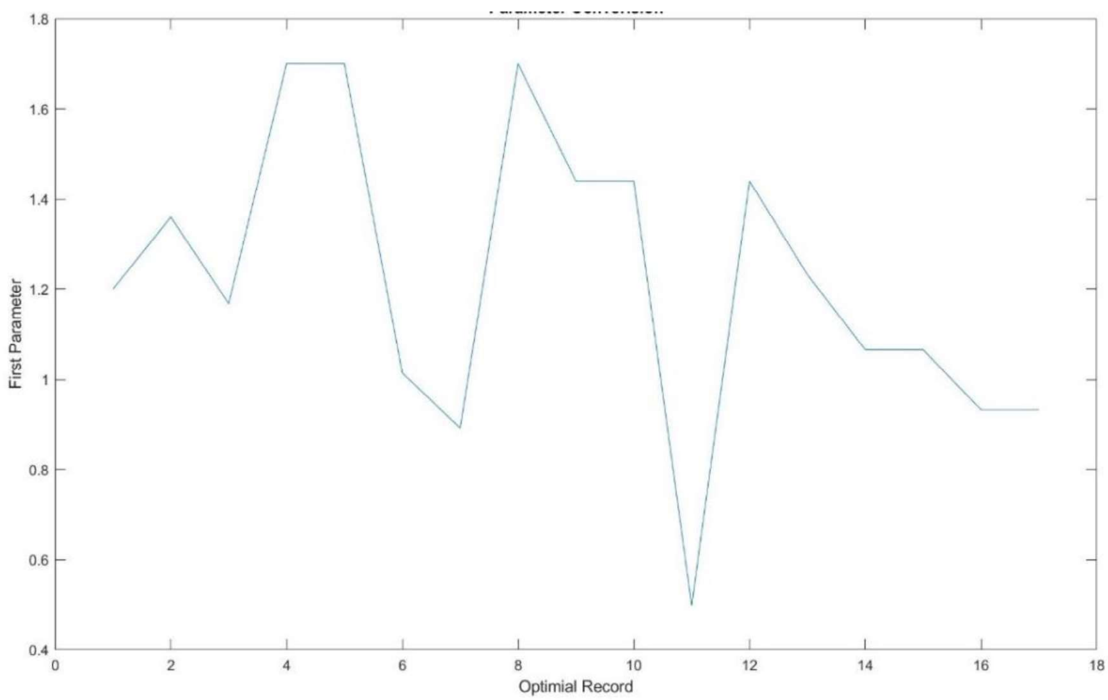


Figure 5.65: First Parameter Convergence for Second Hammerstein Solution Trial

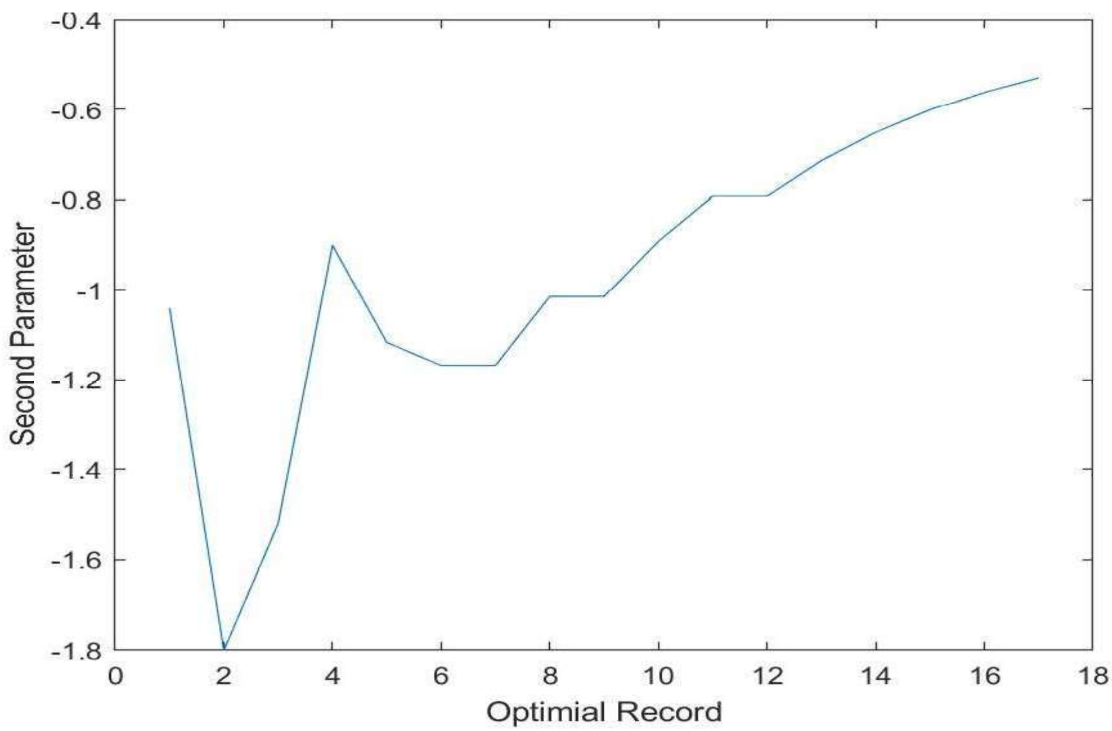


Figure 5.66: Second Parameter Convergence for Second Hammerstein Solution Trial

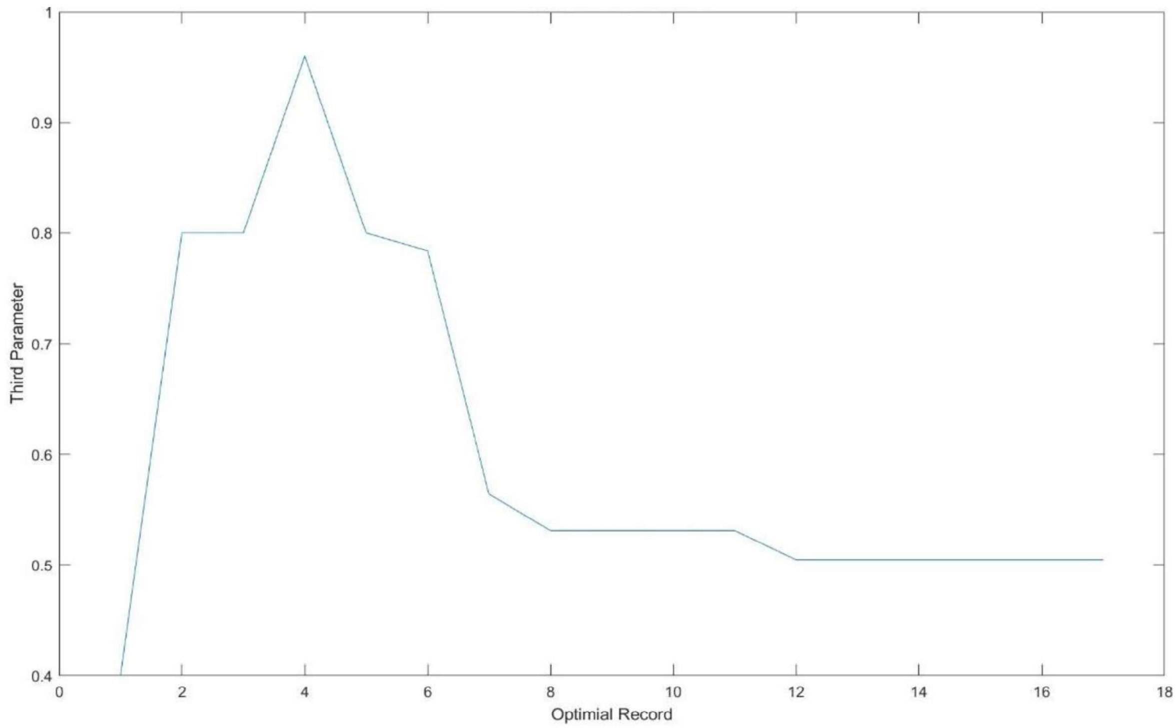


Figure 5.67: Third Parameter Convergence for Second Hammerstein Solution Trial

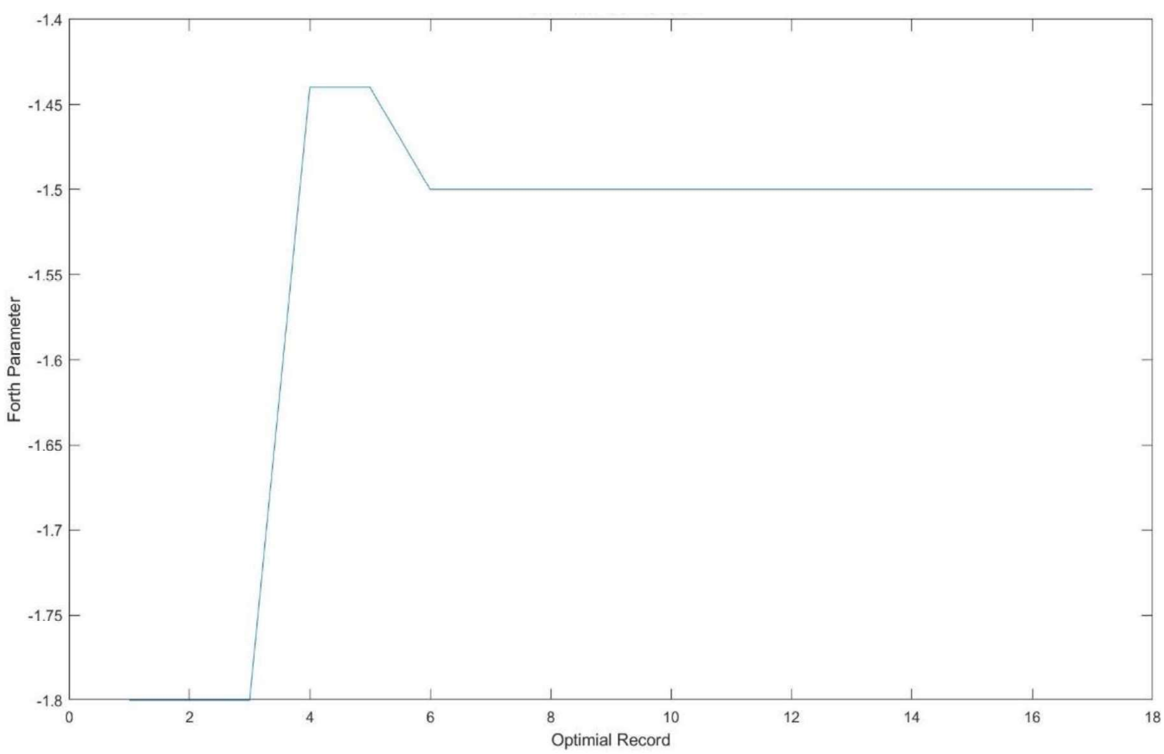


Figure 5.68: Forth Parameter Convergence for Second Hammerstein Solution Trial

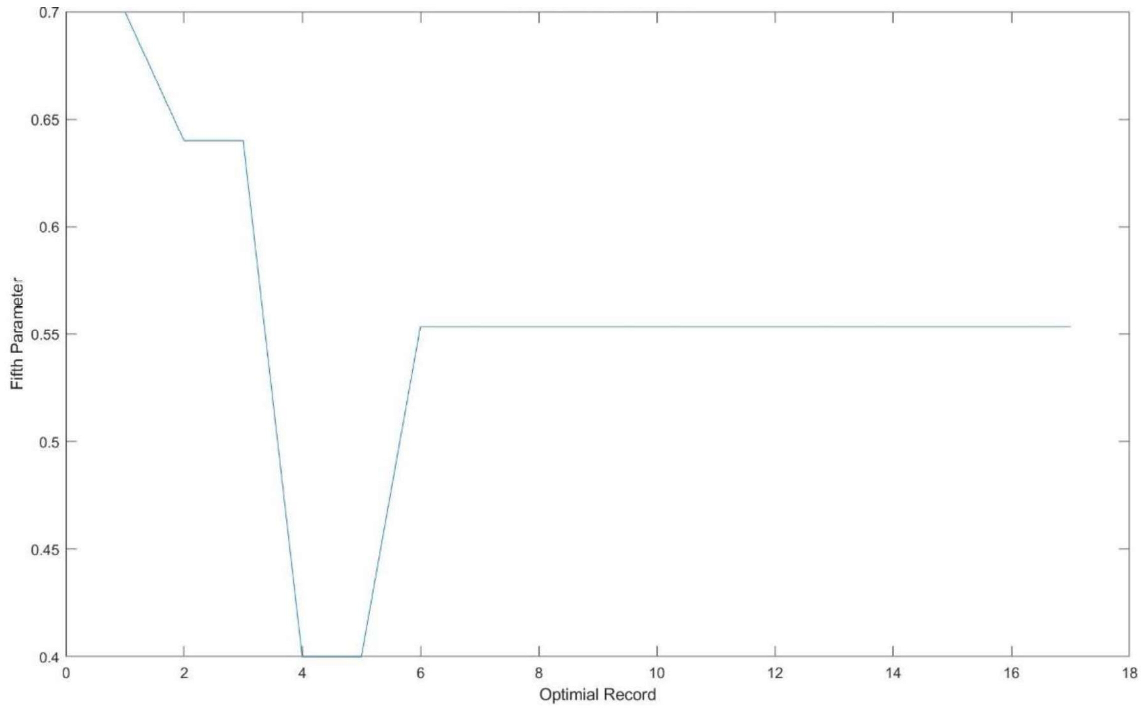


Figure 5.69: Fifth Parameter Convergence for Second Hammerstein Solution Trial

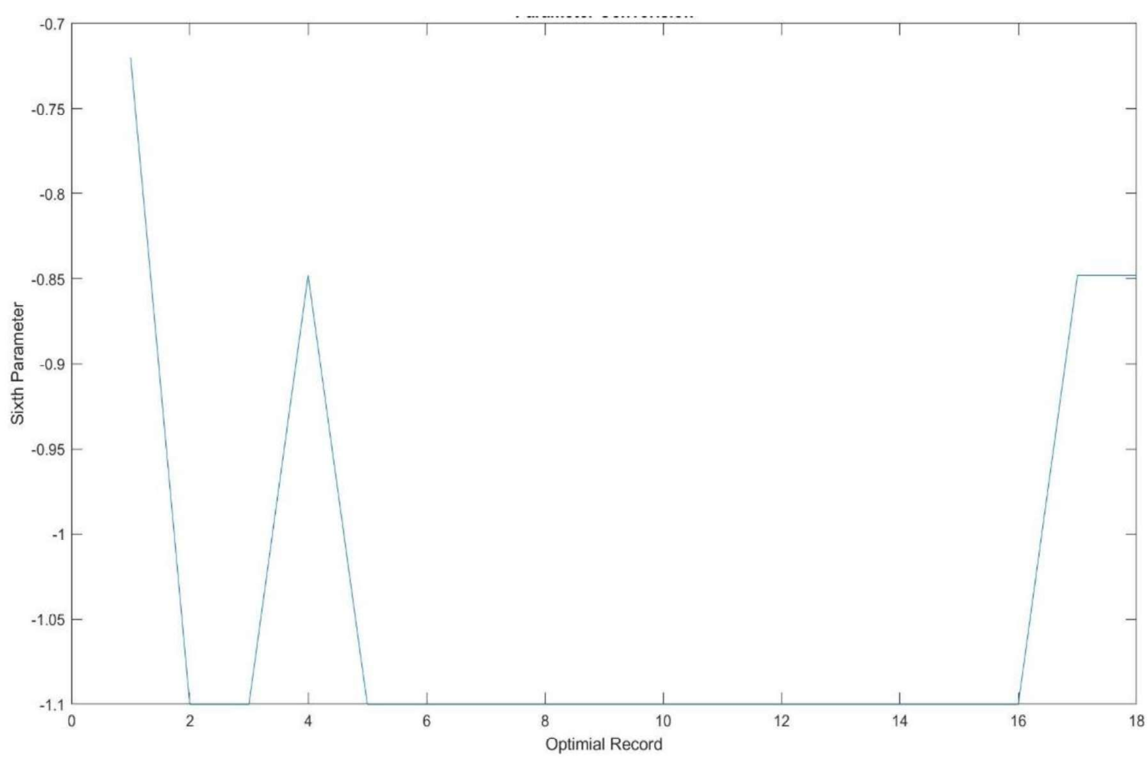


Figure 5.70: Sixth Parameter Convergence for Second Hammerstein Solution Trial

Third Trial

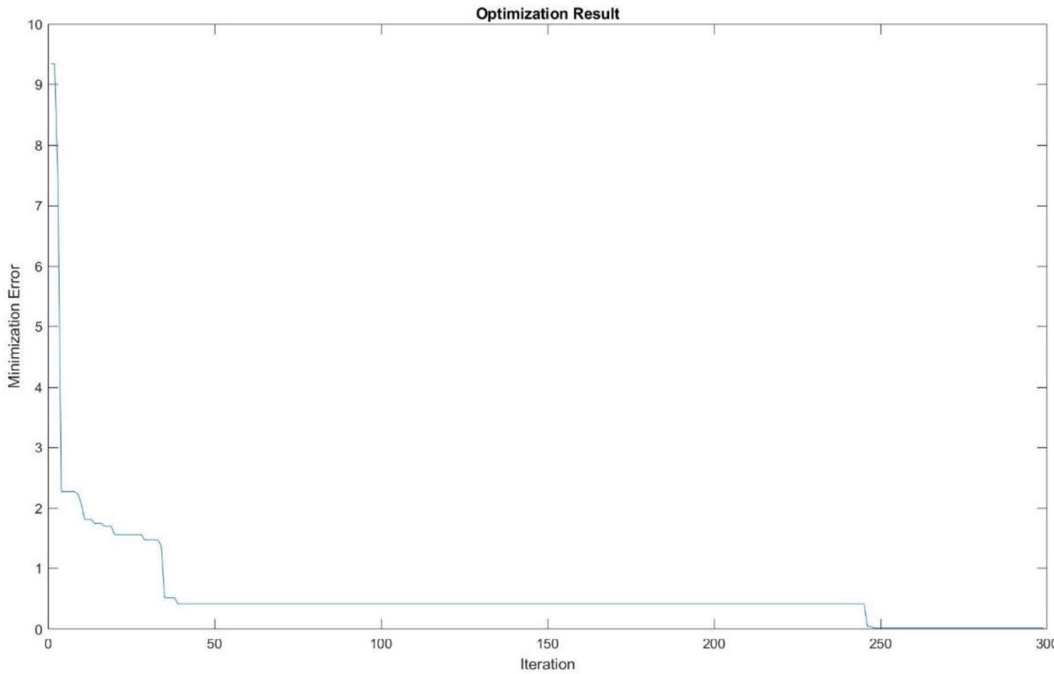


Figure 5.71: Mean Square Error Minimization for Third Hammerstein Solution Trial

MSE= 0.0207
Parameters= [0.6621 -0.4644 0.5203 -1.6000 0.6359 -0.8480]

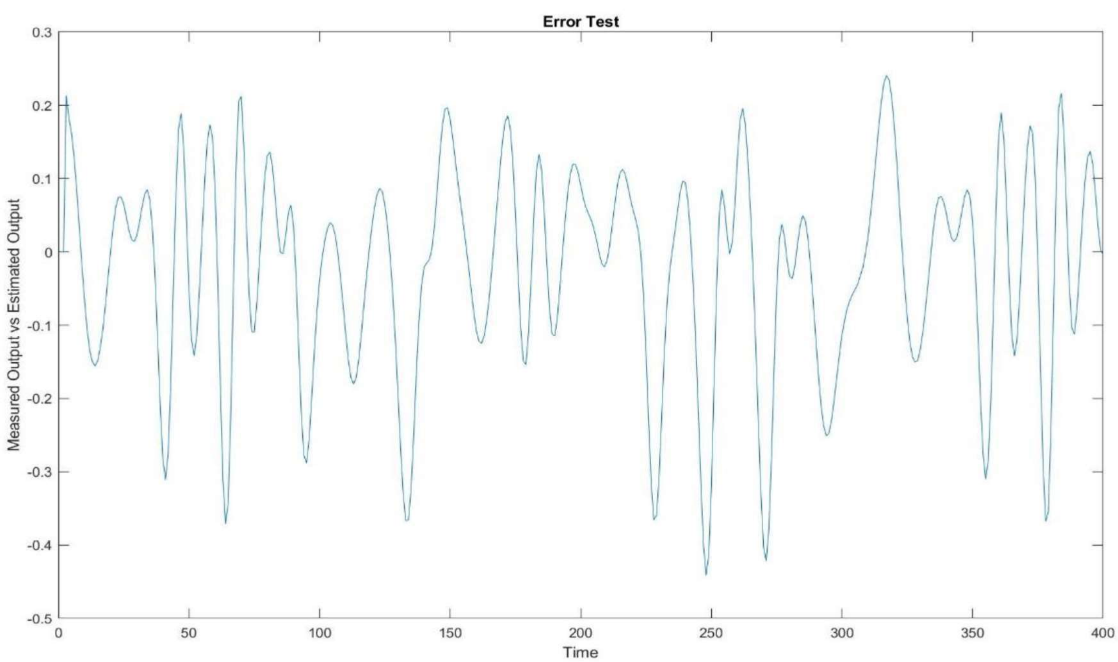


Figure 5.72: Measured Compared with Estimated Output for Third Hammerstein Solution Trial

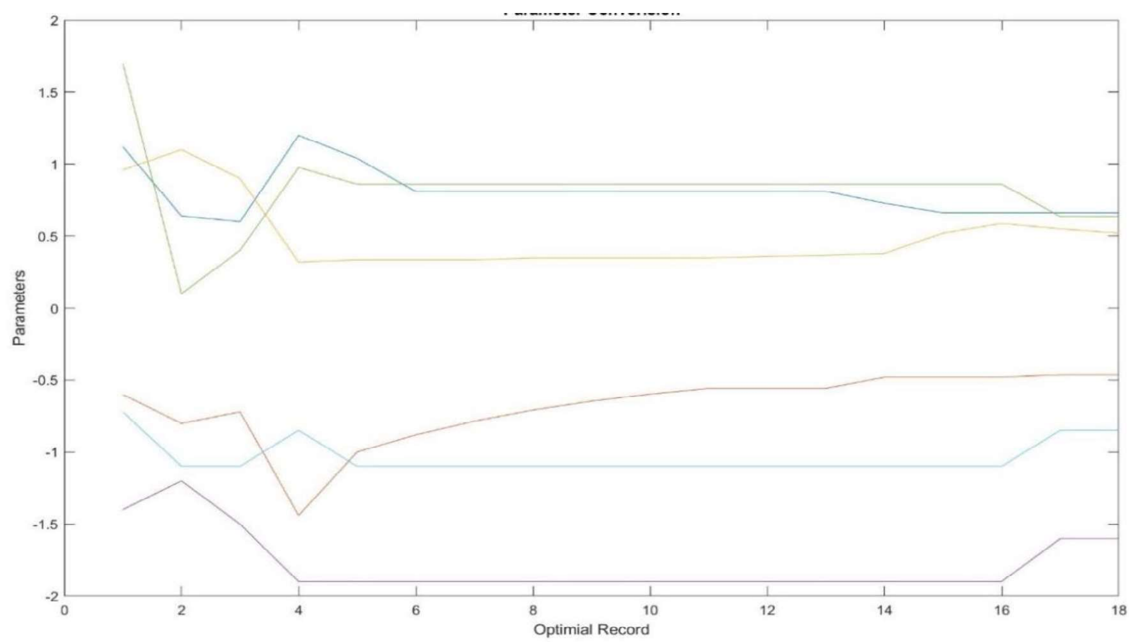


Figure 5.73: Parameters Convergence for Second Hammerstein Solution Trial

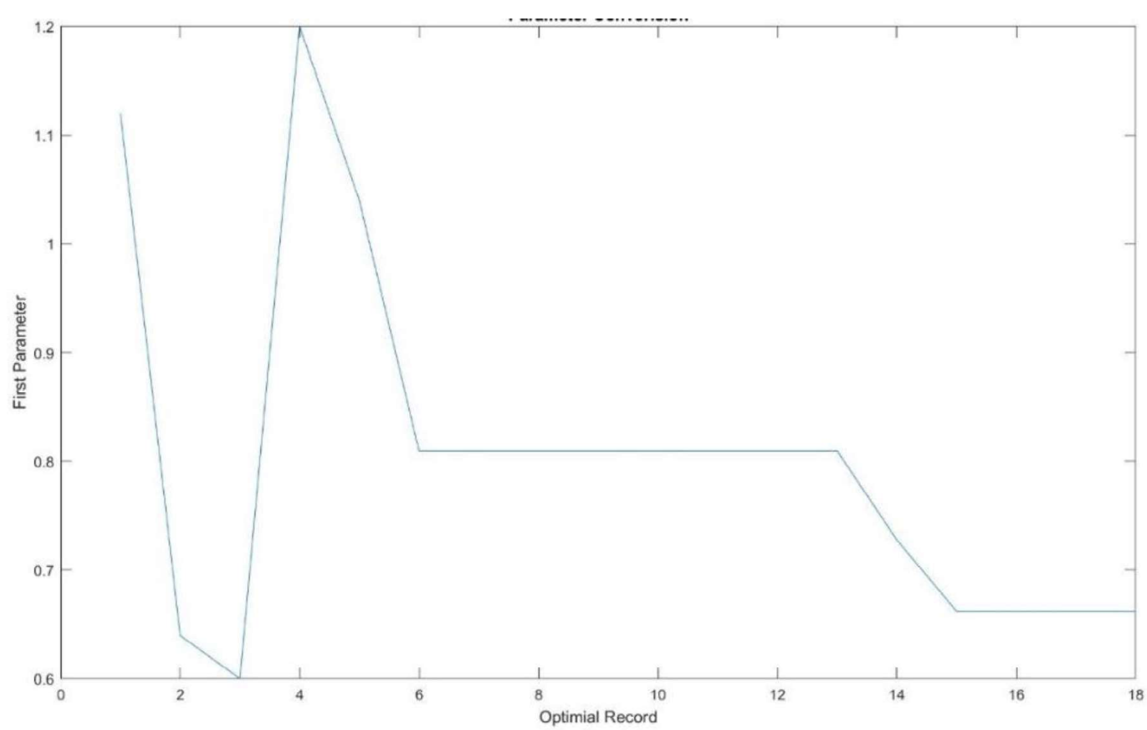


Figure 5.74: First Parameter Convergence for Third Hammerstein Solution Trial

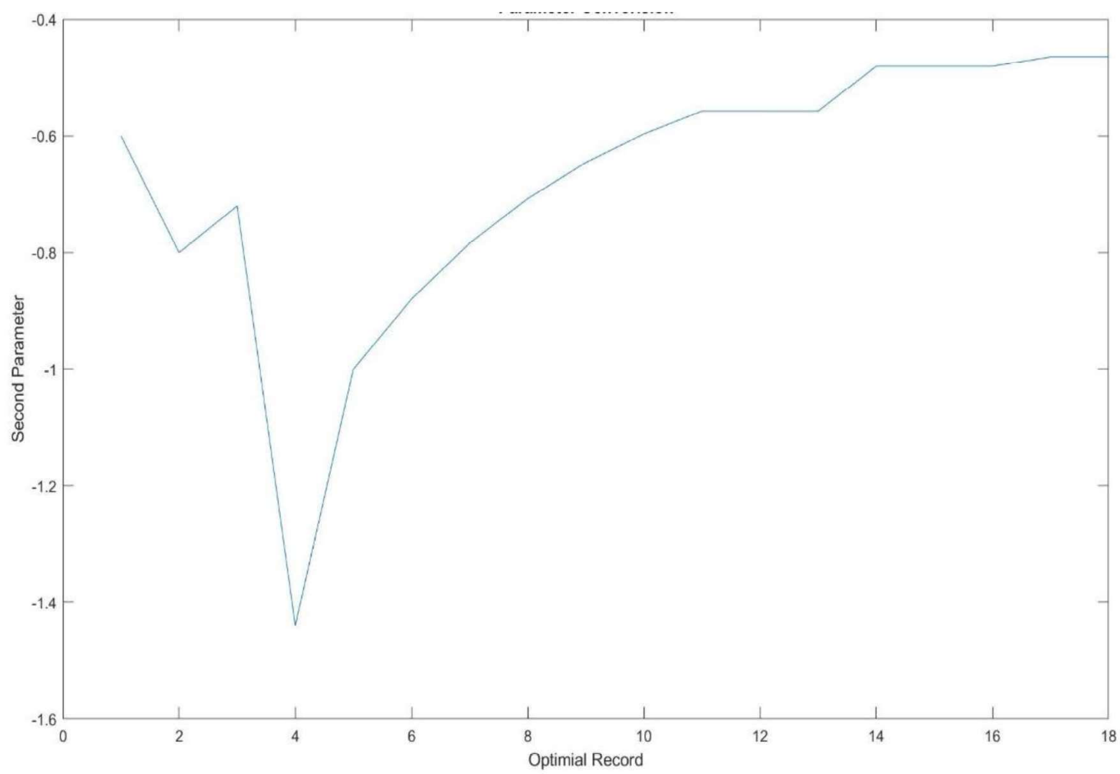


Figure 5.75: Second Parameter Convergence for Third Hammerstein Solution Trial

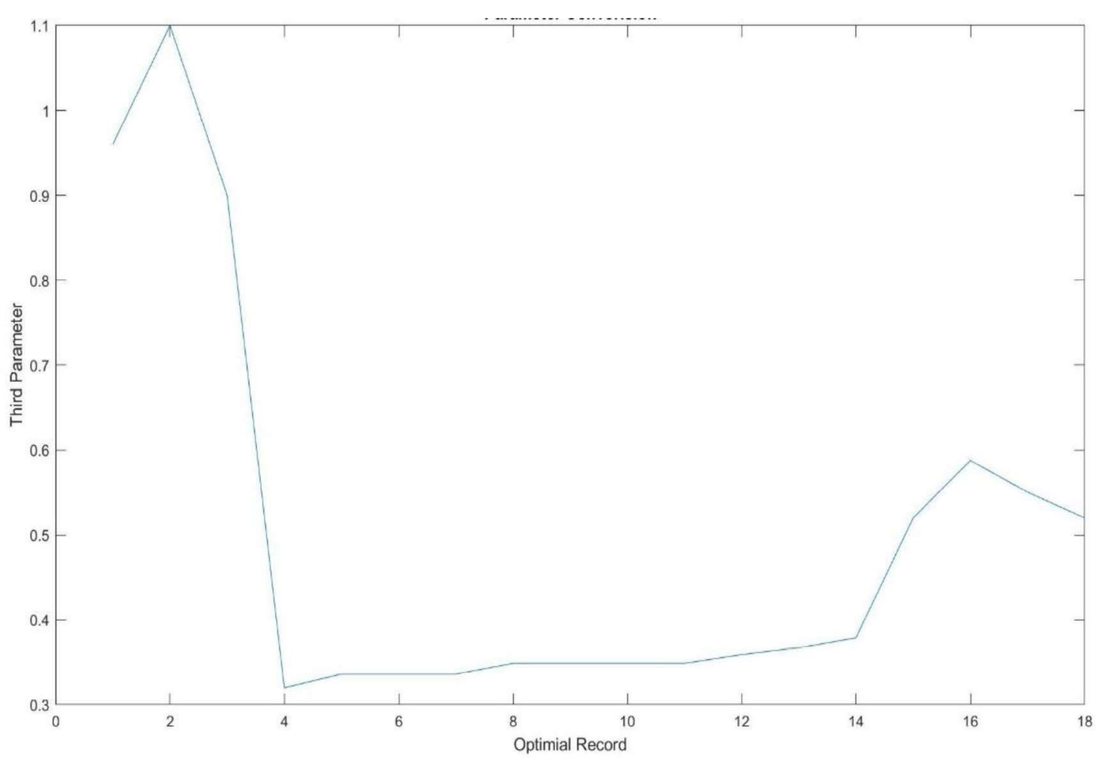


Figure 5.76: Third Parameter Convergence for Third Hammerstein Solution Trial

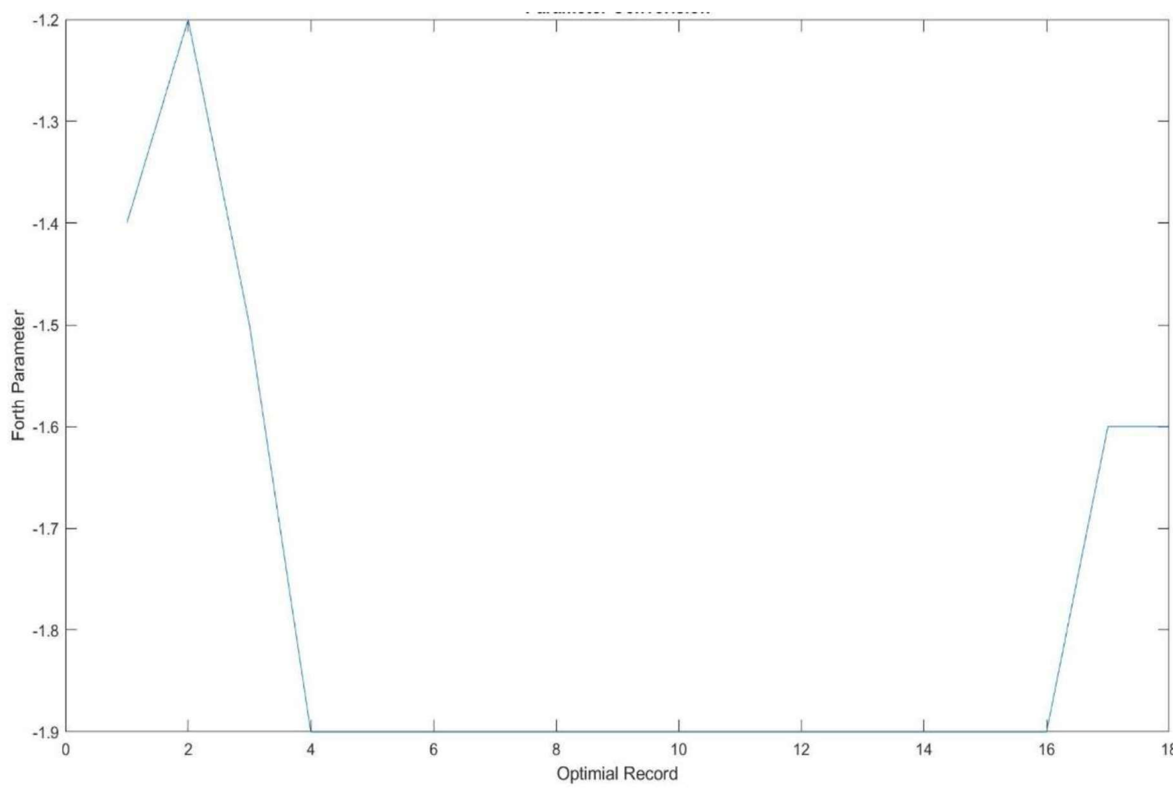


Figure 5.77: Forth Parameter Convergence for Third Hammerstein Solution Trial

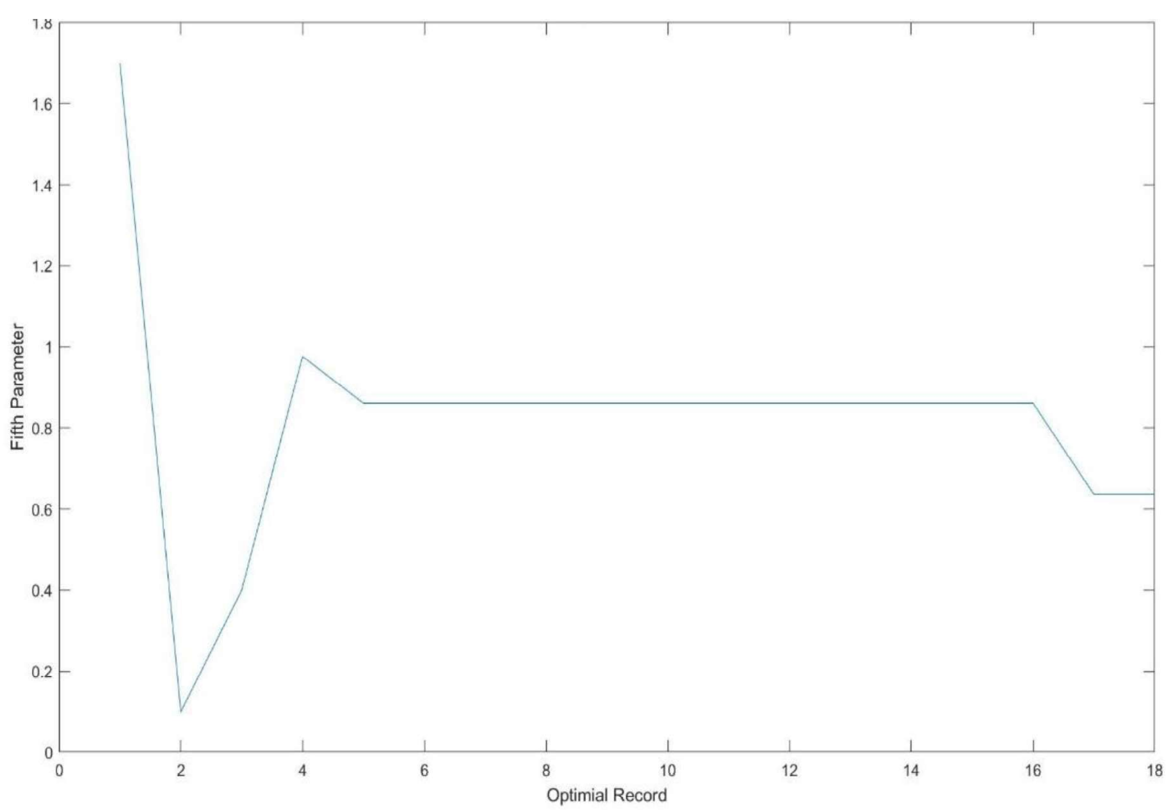


Figure 5.78: Fifth Parameter Convergence for Third Hammerstein Solution Trial

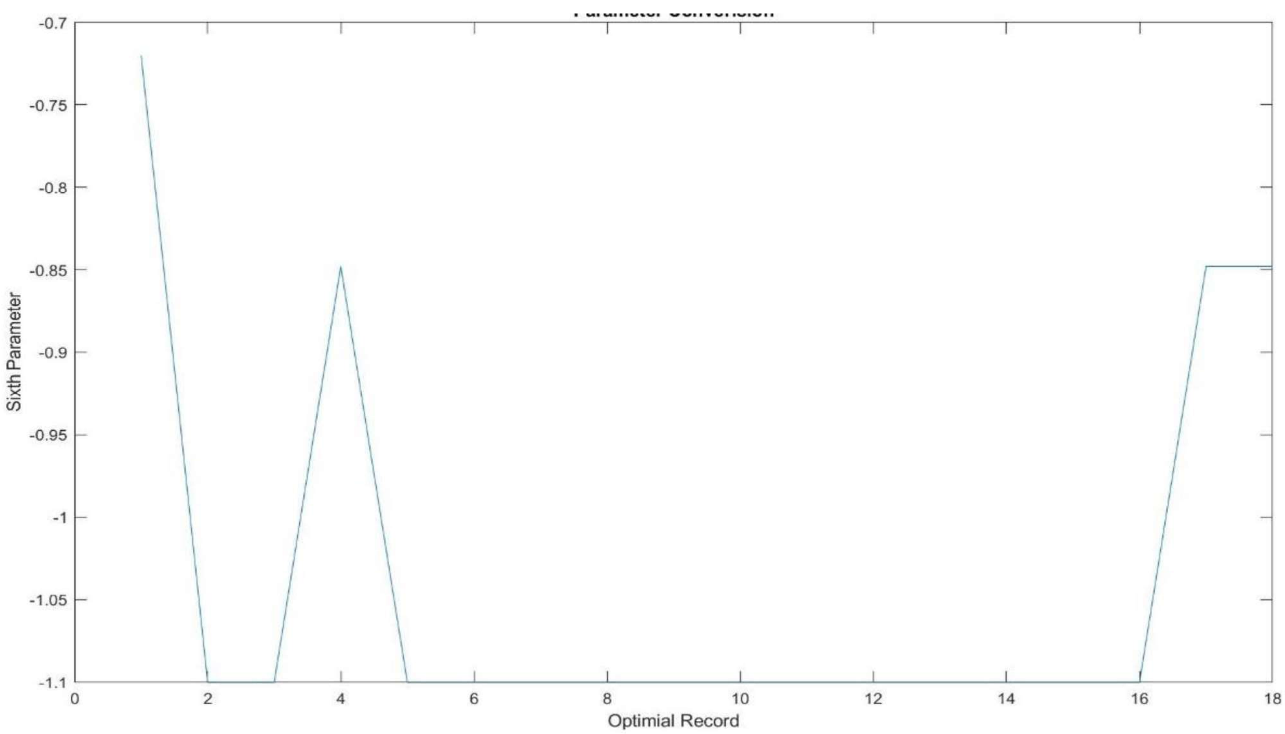


Figure 5.79: Sixth Parameter Convergence for Third Hammerstein Solution Trial

In summary, three trial runs were done. The minimum obtained MSE was 0.0207. Better results could be obtained with more trials.

5.9.1 Hammerstein Models Identification Summary

In this chapter the evolutionary algorithms have been used to identify systems. Several examples of the proposed algorithm have been done and good results were obtained. Real coded genetic algorithm was used to obtain the model structure and least squares algorithm was used to estimate the parameters. In other examples, the evolutionary algorithms were used to identify the parameters assuming the model structure is available. The example includes input-output data of a turbojet system. Other examples involved nonlinear Hammerstein system.

CHAPTER 6

SUMMARY AND CONCLUSION

6.1 Summary

In this thesis evolutionary algorithms were used in identifying a model of the system from input-output historical data. For comparison, eight randomly selected model structures were identified using classical least squares method. The real codded GA algorithm was used with the same data to obtain the best model structure and least squares was used to estimate the parameter of the model. This was shown to give better result. Other examples were done assuming a given model structure and evolutionary algorithms were used to obtain the parameters of the nonlinear system. Thirdly, the non-parametric system identification of neural networks with three different training methods are applied with various neurons numbers where shows better results over the increment of the neurons number but goes worse after some limited numbers. Finally, the evolutionary algorithms still showing the best results over all and promising good future of the nonlinear model structure optimization over the large number of possible models.

6.2 Conclusion and Analysis

The turbojet engine historical data is selected for system identification simulation. The traditional parametric identification is applied for eight different nonlinear model structures. The MSE and fitness compression tests shows exceedingly small differences which reflect the lack of used data and required different model structure selection functional strategy. Thus, introduced the real coded genetic algorithm as a useful tool to search for better model structure on the global population and generation possibilities in short time and resulted two different structured in two trial runs. The real coded genetic algorithm tool can be improved further with more used regressor components and regression vector parameters number flexibility. The results show that the genetic algorithms are useful for system identification and can be applied on wider areas. In addition, the artificial nonparametric identification is applied for the NARX Neural Network model in three training algorithms and different trial neurons number. The results show better mean square error with the neurons number increment, but it went worse after high a limited range. The NARX Neural Network is functional tool for the nonlinear dynamic systems identification. Finally, evolutionary algorithms show the best results over all which gives the signal of the more possible areas of extensions and investigations with constraints and non-linear regressors selections and populations startup design. The Programming codes are developed by MATLAB and used the Simulink toolboxes for simulations and comparisons.

6.3 Recommendations

The thesis covered a good area of identification of parametric models and non-parametric models using neural network. Simulation shows good results. There are several ways that can be used to extend this work. Recommendation for extensions includes the following:

- Use the same approach to identify different dynamical system especially the nonlinear systems.
- Other evolutionary algorithms such as Particle Swarm Optimization and Artificial Bee Colony to be used for system identification and compare their performances.

References:

[1] T. S¨oderstr¨om, P. Stoica, " **System Identification** ", Prentice-Hall, 1989.

[2] Ho B.L., Kalman R.E. **"Effective Construction of Linear State-Variable Models form Input/Output Functions"**. Regelungstechnik, Vol.34, No.12, pp. 545-548, 1966

[3] Astrom K.J., Bohlin T. **" Numerical Identification of Linear Dynamic Systems from Normal Operating Records"**. IFAC Proceedings Volumes, Vol.2, No.2, pp.96-111,1965.

[4] Lennart Ljung, "**System Identification: Theory for the User**", Prentice Hall, 1987.

[5] Michel Gevers, "**System Identification in a Historical Perspective**", IEEE Control Systems Magazine, DISC Summer School, 2015.

[6] Jenkins and Reinsel**, " Time Series Analysis"**, John Wiley & Sons, Inc., 2008.

[7] Lennart Andersson et al, "**A Manual for System Identification**", MathWorks.

[8] Arun K. Tangirala, " **Principles of System Identification Theory and Practice**" Taylor and Francis Group LLC, 2019.

[9] Ian. Gough et al, **"Darwinian Evolutionary Theory and The Social Sciences"** Taylor & Francis Group, Twenty-first century society, 3 (1). pp. 65- 86, 2008.

[10] John H. Holland, "**Genetic Algorithms and Adaptation**", Springer, 1984.

[11] Kenneth De Jong, **"Handbook of Evolutionary Computation"**, IOP Publishing Ltd and Oxford University Press,1997.

[12] Mirjalili, S., **"Evolutionary Algorithms and Neural Networks"** Springer, Vol.780, 2019

[13] Maier, H. R. et al. **" Introductory overview: Optimization Using Evolutionary Algorithms and Other Metaheuristics"** Environmental Modelling & Software, Vol.114, pp 195-213, 2019

[14] Melanie Mitchell, **"An Introduction to Genetic Algorithm"**, MIT Press, 1999.

[15] William S. Hong and aul D. Collopy, **" Technology for Jet Engines: Case Study in Science and Technology Development"** Journal of Propulsion and Power, Vol. 21, No. 5, 2005.

[16] Roger Storm et al, "**Pushing the Envelope: A NASA Guided to Engines**", National Aeronautics and Space Administration, 2007.

[17] E. Naderi , N. Meskin and K. Khorasani, "**Nonlinear Fault Diagnosis of Jet Engines by Using a Multiple Model-Based Approach**", American Society of Mechanical Engineers, 2002.

[18] Tsiamis et al, **"Finite Sample Analysis of Stochastic System Identification"** IEEE 58th Conference on Decision and Control (CDC), pp. 3648-3654, 2019.

[19] Tolson et al, **"Optimization using Evolutionary Algorithms and Other Metaheuristics"**, Environmental Modeling and Software, Vol.114, pp. 195-213, 2019.

[20] Limin Maa, b et al, **"Energy Consumption Optimization of High Sulfur Natural Gas Purification Plant based on Backpropagation Neural Network and Genetic Algorithms"**. Energy Procedia, Vol. 105, pp. 5166–5171, 2017.

[21] Acharjee, P., **"Optimal Power Flow with UPFC using Security Constrained Self-Adaptive Differential Evolutionary Algorithm for Restructured Power System"**, International Journal of Electrical Power & Energy Systems, Vol. 76, pp. 69-81, 2016.

[22] Wei Yan et al, **"A Novel Bi-subgroup Adaptive Evolutionary Algorithm for Optimizing the Degree of Hybridization of HEV Bus"**. Cluster Computing, Springer, Vol 20, No.1, pp. 497-505, 2017.

[23] Hongjun Zhang et al. **"Parameter estimation of nonlinear chaotic system by improved TLBO strategy"**, Springer-Verlag Berlin Heidelberg, Vol. 20, No.1, pp.4965–4980, 2016.

[24] C. West et al, **" A Genetic Algorithm Approach for Parameter Optimization of a 7DOF Robotic Manipulator"** IFAC-Papers on Line Vol.49, No.12, pp.1261–1266, 2016.

[25] Olteanu Marius et al. **"Genetic Algorithm for System Modelling"**, IEEE, Computers and Artificial Intelligence, 2017.

[26] Daniel R. Ojeda G., Luiz A.L.de Almeida, Omar A.C. Vilcanqui, **"Parameter Estimation of Nonlinear Thermoelectric Structures Using Particle Swarm Optimization"**, Simulation Modelling Practice and Theory, Vol.81, pp.1-10, 2018.

[27] Wenhao Yu1 et al, **"Learning a Universal Policy with Online System Identification"**, Georgia Institute of Technology, Google Brain, Google, USA,

[28] K.Worden et al, **" On Evolutionary System Identification with Applications to Nonlinear Benchmarks"**, Mechanical Systems and Signal Processing, Vol.112, pp.194-232, 2018.

[29] Abdulla Ayyad et al, **" Real-Time System Identification Using Deep Learning for Linear Processes with Application to Unmanned Aerial Vehicles"** IEEE, Vol. 8, pp.122539 – 122553, 2020.

[30] Asma Atitallah , Saïda Bedoui, Kamel Abderrahim, **" System Identification: Parameter and Time-Delay estimation for Wiener Nonlinear Systems with Delayed Input"** Transactions of the Institute of Measurement and Control, Vol. 40 No.3, pp. 1035–1045, 2018.

[31] Mohammad Jahani, Hamed Mojallali, Mohammad Teshnehlab, **"Recursive Identification of Multiple-Input Single-Output Fractional-Order Hammerstein Model with Time Delay"** Applied Soft Computing, Vol. 70, pp. 486–500, 2018.

[32] Dongmin Yu et al, **" System Identification of PEM Fuel Cells using an Improved Elman Neural Network and a New Hybrid Optimization Algorithm"** Energy Reports, Vol. 5, pp. 1365-1374, 2019.

[33] Timothy Sand, **" Nonlinear-Adaptive Mathematical System Identification"** MDPI Journal, USA Computation, Vol. 5, No. 4, 2017.

[34] Irina Carmen Andrei et al, **"The Completion of the Mathematical Model by Parameter Identification for Simulating a Turbofan Engine".** National Institute for Aerospace Research, Incas Bulletin, Vol. 7, No. 3, pp.25-37, 2015.

[35] Ladislav Nyulász et al, **"Comparison of Experimental Identification Methods Using Measured Data from a Turbojet Engine"** IEEE 14th International Symposium on Applied Machine Intelligence and Informatics, 2016.

[36] Károly Beneda, Ladislav Főző, **"Identification of Small-Scale Turbojet Engine with Variable Exhaust Nozzle"** IEEE 15th International Symposium on Applied Machine Intelligence and Informatics, 2017.

[37] K´aroly Beneda, Rudolf Andoga, Ladislav Főző, **"Linear Mathematical Model of Single-Spool Micro Turbojet Engine with Fixed Exhaust Nozzle"** IEEE 16th World Symposium on Applied Machine Intelligence and Informatics, 2018.

[38] Evgeny Filinov et al, **"Correlation-Regression Models for Calculating the Weight of Small-Scale Aircraft Gas Turbine Engines",** MATEC Web of Conferences, Samara National Research University, 2018.

[39] Khaoula Derbel , Károly Beneda, **"Linear Dynamic Mathematical Model and Identification of Micro Turbojet Engine for Turbofan Power Ratio Control"**, Aviation Journal Vol. 23, No. 2, pp. 54–64, 2019.

[40] Qianjing Chen et al, **"A Novel Real-Time Mechanism Modeling Approach for Turbofan Engine"**, Energies Journal, Vol.12, 2019.

[41] Ecaterina Vladu et al,"**Using Genetic Algorithms in System Identification**", Department of Electrical Engineering and Information Technology, University of Oradea, 410087 Oradea, România, 2005.

[42] Wei-Der Chang, "**Nonlinear System Identification and Control using a Real-Coded Genetic Algorithm**", Applied Mathematical Modelling, Vol. 31, No.3, pp. 541-550, 2006.

[43] Songtao Xue , Hesheng Tang, Jin Zhou "**Identification of Structural Systems Using Particle Swarm Optimization**", Journal of Asian Architecture and Building Engineering, Vol.8 No.2, 2009.

[44] Mahboobeh Habibinejad, Mohsen Ramani, Mostafa Khosravi, "**System identification by genetic algorithm for predicting the gradation of API-X70 steel**", Global Journal on Technology, Vol 1, 2012.

[45] Ozden Erçin, Ramazan Coban, "**Identification of Linear Dynamic Systems using the Artificial Bee Colony Algorithm**", Turk J Elec Eng. & Comp Sci, Vol. 20, No.1, 2012.

[46] Thanh N. Trinh, Chan Ghee Koh "**An Improved Sub Structural Identification Strategy for Large Structural Systems**", International Association for Structural Control and Monitoring, Vol.19, No.8, pp. 686-700, 2012.

[47] Tavakolpour-Saleh, Nasib, Sepasyan and Hashemi, **"Parametric and Nonparametric System Identification of An Experimental Turbojet Engine**", Aerospace Science and Technology, Vol. 43, pp. 21-29, 2015.

[48] Andre Felipe et al**," A Modified Matricial PSO Algorithm Applied to System Identification with Convergence Analysis**", Journal of Control, Automation and Electrical Systems, Vol. 26, pp.149–158, 2015.

[49] Partha S.Pal et al," **An Efficient Identification Approach for Stable and Unstable Nonlinear Systems using Colliding Bodies Optimization Algorithm** ", ISA Transactions, Vol.59, pp. 85-104, 2015.

[50] Loris Vincenzi, Marco Savoia," **Coupling Response Surface and Differential Evolution for Parameter Identification Problems**", Computational Intelligence in Structural Engineering and Mechanics, Vol.30, No. 5, 2015.

[51] Hong Peng, Jun Wang " **A Hybrid Approach Based on Tissue P Systems and Artificial Bee Colony for IIR System Identification**", Springer, Neural Computing and Applications Vol.28 ,No.9, pp.2675–2685, 2017.

[52] Jinyao Yan, J.R.Deller ," **NARMAX Model Identification Using a Set-Theoretic Evolutionary Approach** ", Signal Processing, Vol.123, pp. 30-41, 2016.

[53] Yuhao Huang et al, " **An Application of Evolutionary System Identification Algorithm in Modelling of Energy Production System** " Measurement ,Vol.114, No.1, pp. 122-131, 2018.

[54] Faizal Hafiz et al, " **Structure Selection of Polynomial NARX Models using Two-Dimensional (2D) Particle Swarms**", IEEE Institute of Electricial and Electronics Engineers, 2018.

[55] K. Worden et al, "**On Evolutionary System Identification with Applications to Nonlinear Benchmarks**", Mechanical Systems and Signal Processing , Vol. 112, pp. 194-232, 2018.

[56] Laura S. de Assiset et al," **Efficient Volterra Systems Identification Using Hierarchical Genetic Algorithms** ", Applied Soft Computing , Vol. 85, 2019.

[57] Helon Vicente et al, "**Nonlinear Black-box System Identification Through Coevolutionary Algorithms and Radial Basis Function Artificial Neural Networks** ",Applied Soft Computing, Vol. 87 , 2020.

[58] John H. Holland, **"Genetic Algorithms and Adaptation"**, Springer, 1984

[59] F. Herrera, M. Lozano and J.L. Verdegay, "**Tackling Real Coded Genetic Algorithms Operators and Tools for Behavioral Analysis**" Springer, Vol. 12, pp.265-319, 1998.

[60] Eberhart RC, Kennedy J, A **"Particle Swarm Optimization"** IEEE international conference on neural networks, pp 1942–1948, 1995

[61] Cao L., Schwartz H.M. **"Exponential Convergence of the Kalman Filter-Based Parameter Estimation Algorithm"**. International Journal of Adaptive Control and Signal Processing, Vol.17, No.10, pp. 763-783, 2003.

[62] Ding F., Chen T. **"Parameter Estimation of Dual-Rate Stochastic Systems by using an Output Error Method"**. IEEE Trans. Autom. Control, Vol.50, No.9, pp. 1436-1441, 2005.

[63] Ding F., Chen T. **"Combined Parameter and Output Estimation of Dual-Rate Systems using an Auxiliary Model"**. Automatica, Vol.40, No.10, pp. 1739-1748, 2004.

Vitae

Name	:Mohammed Hussain AlBahrani
Nationality	:Saudi
Date of Birth	:12/14/1989
Email	:malbahrani55@gmail.com
Address	:Jubail Industrial City
Academic Background	:Instrumentation and Control Engineering