Inter-area Oscillation Damping Control using Machine Learning

By

Mazahir Saleem

Spring 2019-MS-CSE-00000281792

Supervisor

Dr. Jawad Arif

Department of Computational Science and Engineering

School of Interdisciplinary Engineering and Sciences (SINES)

National University of Sciences and Technology (NUST)

Islamabad, Pakistan

December 2022

Inter-area Oscillation Damping Control using Machine Learning

By

Mazahir Saleem

Spring 2019-MS-CSE-00000281792

Supervisor

Dr. Jawad Arif

Co-supervisor

Dr. Mian Ilyas Ahmad

A thesis submitted in conformity with the requirements for the degree of *Master of Science* in Computational Science and Engineering

Department of Computational Science and Engineering

School of Interdisciplinary Engineering and Sciences (SINES)

National University of Sciences and Technology (NUST)

Islamabad, Pakistan

December 2022

Declaration

I, *Mazahir Saleem* declare that this thesis titled "Inter-area Oscillation Damping Control using Machine Learning" and the work presented in this thesis has been generated by me as a result of my own original research.
I confirm that:

1. This work was done wholly or mainly while in candidature for a Master of Science degree at NUST.

2. Where any part of this thesis has previously been submitted for a degree or any other qualification at NUST or any other institution, this has been clearly stated.

3. Where I have consulted the published work of others, this is always clearly attributed.

4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.

5. I have acknowledged all main sources of help.

6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Mazahir Saleem,

Spring 2019-MS-CSE-00000281792

Copyright Notice

This thesis is dedicated to *my beloved parents and my siblings*

Abstract

Inter-area oscillations are one of the major threats to the stability of an interconnected power system due to the fact that these oscillations involve generators from different areas of a power system and hence may result in higher oscillation in the tie-line by adding up effects from each of the participating generators. The success of oscillation damping within a given duration, such as less than 15 seconds, is substantially correlated with the safe operation of a contemporary power system. Since power systems are highly non-linear in nature, with time varying parameters, and a specific control design based on the linearized model may not ensure satisfactory performance under varied operating scenarios. Therefore, a non-linear self-tuning controller is required that damps inter-area oscillations in interconnected power systems under varied post-disturbance operating conditions without requiring manual adjustment or re-tuning of controller parameters. An effective controller for power oscillation damping has been designed in this research. Moreover, a powerful online batch training method for neural networks known as the Levenberg-Marquardt (LM) algorithm is modified for computationally efficient online estimation of power system's dynamic behaviour and is referred as Computationally Efficient Levenberg-Marquardt (CELM) algorithm. A unique form of neural network, referred as Computationally Efficient Neural Network (CENN), is proposed that is compatible with the Feedback Linearizable Controller (FBLC), is used to ensure non-linear self-tuning control. It has been demonstrated that using the modified version of LM algorithm i.e. CELM algorithm for successive disturbance leads to better accuracy, faster convergence and offer less computational time than using the classical LM algorithm.

Keywords: *Inter-area oscillation, Damping, Non-linear estimation, Computationally efficient neural network (CENN), Levenberg-Marquardt (LM), Computationally efficient Levenberg-Marquardt (CELM), Feedback linearization control (FBLC).*

Acknowledgments

I am thankful to my Creator Allah Subhana-Watala to have guided me throughout this work at every step and for every new thought which You setup in my mind to improve it. Indeed I could have done nothing without Your priceless help and guidance. Whosoever helped me throughout the course of my thesis, whether my parents or any other individual was Your will, so indeed none be worthy of praise but You. I am profusely thankful to my beloved parents who raised me when I was not capable of walking and continued to support me throughout in every phase of my life. I would also like to express special thanks to my supervisor Dr. Jawad Arif and co-supervisor Dr. Mian Ilyas Ahmad for their help throughout my thesis and also for Linear Control Systems and System Identification courses which he has taught me. I can safely say that I haven't learned any other engineering subject in such depth than the ones which he has taught.

I would also like to thank Dr. Absaar Ul Jabbar and Dr. Salma Sherbaz for being on my thesis guidance and evaluation committee.

Finally, I would like to express my gratitude to all the individuals who have rendered valuable assistance to my study.

Contents

List of Figures

List of Tables

List of Abbreviations

ANN	Artificial Neural Network
ARX	Auto-Regressive with External Input
BP	Back-Propagation
BPTT	Back-Propagation Through Time
CENN	Computationally Efficient Neural Network
CELM	Computationally Efficient Levenberg-Marquardt
ENN	Elman Neural Network
FACTS	Flexible AC Transmission Systems
FBLC	Feedback Linearizable Controller
LM	Levenberg-Marquardt
MLP	Multilayer Perceptron
NN	Neural Network
PSS	Power System Stabilizers
RBF	Radial Basis Function
RLS	Recursive Least Square
RNN	Recurrent Neural Network
STC	Self-Tuning Control
SRN	Simultaneous Recurrent Neural Network

Chapter 1

Introduction

1.1 Background

Electric power supply networks are huge and integrated in a complex way. Deregulation, reregulation, restructuring, and the uncertainty of what is still to come have caused utilities in many regions of the world to make varied investment decisions. Over the years, the process of obtaining approval to build new transmission lines has become extraordinarily complicated, time-taking, and costly [4]. The load on current transmission lines will increase continuously, necessitating the most efficient use of existing transmission assets. Furthermore, the producing sites (for example, in Pakistan) are far away from the load centers, so power must be transmitted across a great distance. The main challenge in this scenario is ensuring the security of power supply, when there are low frequency electro-mechanical oscillations [2, 5]. The oscillations, in a frequency range of 0.1Hz to 1.0Hz, are mostly owing to the back and forth movement of one generator group in relation to another [6]. Oscillatory instability has been blamed for a number of instances, some of which resulted in infamous blackouts affecting huge areas [3]. This causes network operators to take a cautious approach in order for the system to run within a reasonable stability margin, resulting in inefficient use of current assets.

Over the last four decades, significant progress has been made in the addition of supplementary control to damp the unwanted oscillations [7–22]. The criteria for damping these unwanted oscillations vary by benefit, for instance, in european countries, a settling time of 10 to 15 seconds is mandatory [23]. The utilization of power system stabilizers (PSS) [7] is a well-known and economical method to improve power system stability.

In recent times, flexible ac transmission system (FACTS) devices has entered the control space and has shown to be a long-term, low-cost solution for improving reliability and transmission capacity [24–29]. FACTS devices make it possible to regulate power flow more precisely and load transmission lines safely such that they are loaded close to their thermal limits [30–33], making them a practical substitute for the building of new transmission lines. Furthermore, because they are installed on transmission lines, these instruments have direct access to the factors influencing inter-area oscillation.

Typically, the dynamics of a power system is enhanced by modulating the excitation system of generators. For years, PSS have been primarily utilized to mitigate oscillations [34]. For control design, traditional model-based procedures are utilized, which depend on the provision of correct system parameters. This is dependent on precise knowledge about the factors involved, such as load current, voltage, frequency, and so on [35]. In a real-time context, obtaining these critical parameters can be difficult. Independent power producers (IPPs) will make the expanding interconnected power systems even more complex. As a result, the focus of study is on control design based on input-output mapping of the system, which eliminates the need for precise knowledge about the system parameters.

1.2 Problem Statement

Power systems are highly non-linear with time varying parameters, and a fixed control design based on the linearized model may not guarantee satisfactory performance over various operating conditions. Therefore, a non-linear self-tuning controller is required that damps inter-area oscillations in interconnected power systems under varied post-disturbance operating conditions without requiring manual adjustment or re-tuning of controller parameters.

1.3 Research Motivation

Although traditional control strategies are simple, but they have a limited working range. Traditional non-adaptive designs are constructed close to the nominal operating point, they may produce oscillations that are poorly damped or even unstable, due to uncertainties and drastic changes in operating conditions [28]. Use of robust control

techniques can increase the performance range of such a linear controller [36–38]. However, with significant disturbance, the post-disturbance system may deviate significantly from its nominal operational states and may even outperform the robust controllers performance range. The key risk for network operators is that the system may migrate to an unanticipated operating condition or network structure after a disruption, such as a bus malfunction followed by a transmission network loss. Hence, a self-tuning controller is needed to maintain proper damping throughout a wide range of operating conditions. In order to increase stability, the controller must quickly recognise electro-mechanical oscillations, estimate the suitable parameters of the controller, and implement the necessary control actions.

The control of FACTS devices has recently been the focus of efforts. Until now, the PSS/FACTS controller settings have been derived using a linearized power system model. The parameters of the FACTS devices must be adjusted in reaction to the oscillations, in order for them to offer suitable damping over variety of operating points. With time-varying parameters, power systems are really non-linear, so a specific control strategy based on the linear system might not perform well enough in a range of operating scenarios. [39]. Furthermore, restricting the controller to the linear domain may not be feasible, particularly under extreme stress circumstances [40]. As a result, a non-linear self-tuning controller that considers system non-linearities and adjusts to changing operating conditions could potentially produce better outcomes [41].

The major intent of this study is to develop a non-linear self-tuning controller that damps inter-area oscillations in interconnected power systems under varied post-disturbance operating conditions without requiring manual adjustment or re-tuning of controller parameters. Also, it should only necessitate a basic understanding of the system and its post-disturbance operating conditions. Moreover, a powerful online batch training method for neural networks known as the Levenberg-Marquardt (LM) algorithm is modified for improved online estimation of power system dynamic behaviour and is named as Computationally Efficient Levenberg-Marquardt (CELM) Algorithm. A unique form of neural network, named as Computationally Efficient Neural Network (CENN), is proposed that is compatible with the Feedback Linearizable Controller (FBLC) is used to enable non-linear self-tuning control. It has been demonstrated that using the modified version of LM algorithm i.e. CELM for successive disturbance leads to improved accuracy, faster convergence and offer less computational time than using the typical LM

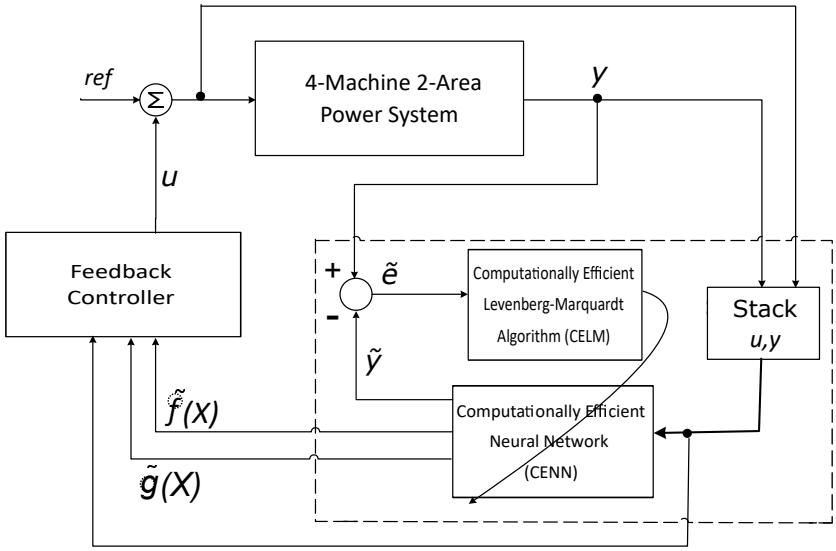

Figure 1.1: Control architecture employing computationally efficient neural network (CENN) with computationally efficient levenberg-marquardt (CELM) in conjunction with feedback linearizable controller (FBLC)

algorithm.

1.4 Research Objectives

The main objectives of this thesis are:

- Modelling of power system behaviour using the neural network modelling technique(s).
- Simulate the behavior of the power system using the online estimation technique(s).
- Optimized neural network design to make the model computationally inexpensive.
- Design a controller based on the proposed neural network model for inter-area oscillation.

Power system design is not considered in this research work. A schematic outline of the control architecture is shown in Figure. 1.1.

In summary, the non-linear self-tuning controller is investigated in this thesis in order to damp oscillations in power system. Non-linear model has considered for the

power system. For estimation of parameters, online modified levenberg-marquardt i.e. computationally efficient levenberg-marquardt (CELM) in batch mode is used. On the 4-machine 2-area power system [2], the computationally efficient neural network - computationally efficient levenberg-marquardt (CENN-CELM) in conjunction with feedback linearizable controller (FBLC) is tested.

1.5 Description of the power system

The method to simulate a variety of power system components is quite typical. The main goal of power system stability is to maintain all of the connected machines in sync [42, 43]. Their stability is also influenced by a number of other factors. such as speed governors, generator excitation systems, loads, and the FACTS devices etc. A set of non-linear differential algebraic equations (DAE) describes the dynamic behaviour of an interconnected power system. The model employed in this study, which is useful for studying inter-area oscillations, is decribed in [44].

1.5.1 4-Machine 2-Area Power System

The case study uses a 4-machine, 2-area power system [2], as depicted in Figure. 1.2. which is a benchmark model for studying inter-area oscillations. Approximately 400MW flows from area 1 to area 2 via a 220 km transmission line at steady state. A thyristor controlled series capacitor (TCSC) is inserted in one of the lines to manage and facilitate this tie-line power flow [45]. It provides 10% compensation in steady state and has a dynamic range of variation from 1% to 50%. When a disturbance occurs in the system, poorly damped oscillations are brought on by the presence of a lightly damped inter-area mode. The goal is to create a TCSC control scheme that reduces undesirable oscillations.

For this simple system, there are two plausible line outage scenarios that can be implemented while maintaining system integrity. These are either a line outage linking buses 7 and 8 or a line outage linking buses 8 and 9, both of which cause inadequately damped inter-area oscillations. For the reasons stated in [6], an extra damping control for the TCSC using the observed flow in line 10 - 9 will be created.

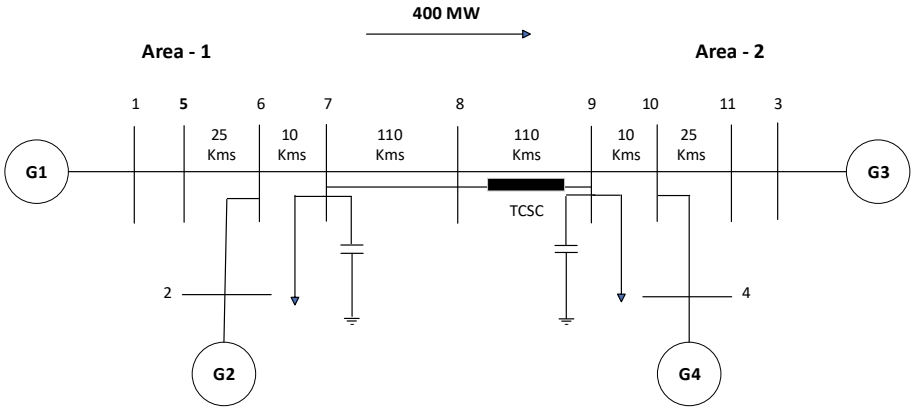

Figure 1.2: 4-Machine 2-Area Power System [2]

1.6 Thesis Organization

The thesis has organized as follows:

Chapter 2 discusses literature review, and it focuses on the techniques, algorithms and methods applied by researchers for power system oscillation damping control.

Chapter 3 explains the methodology used in this research. It explains the fundamental concept of self-tuning controller. The detailed study and modelling of neural network (NN) and computationally efficient neural network (CENN), that is compatible with feedback linearizable controller (FBLC) is done. Later, this chapter explains the classical Levenberg-Marquardt (LM) algorithm and its modified version (CELM) in detail. The chapter ends with the explanation and derivation of feedback linearizable controller (FBLC).

Chapter 4 compares the results between computationally efficient neural network (CENN) - classical Levenberg-Marquardt (LM) and computationally efficient neural network (CENN) - computationally efficient Levenberg-Marquardt (CELM) in terms of accuracy, convergence and computational time. The efficiency of online modified LM approach i.e. CELM for non-linear estimation is shown through a case study on a 4-machine 2-area power system.

Finally, **Chapter 5** highlights conclusions from this work, and gives direction towards future research.

1.7 Thesis Contributions

In this study, a measurement-based controller i.e Feedback linearizable controller (FBLC), is designed that requires little prior knowledge of the system's parameters and post-disturbance operating conditions to damp inter-area oscillations in the power system. For the estimation of the power system, a classical Levenberg-Marquardt (LM) algorithm has been transformed into an online Levenberg-Marquardt approach referred as computationally efficient Levenberg-Marquardt (CELM) in sliding window batch mode. To capture the non-linearities in system responses, a novel structure of neural network i.e. computationally efficient neural network (CENN) has been implemented with the computationally efficient Levenberg-Marquardt (CELM) algorithm. The computationally efficient neural network structure (CENN) in conjunction with computationally efficient Levenberg-Marquardt (CELM) estimator has utilized in a closed loop using a feedback linearizable controller (FBLC), which has been demonstrated to produce improved performance in terms of accuracy, convergence, and computational time than computationally efficient neural network structure (CENN) in conjunction with classical Levenberg-Marquardt (LM) estimator.

Chapter 2

Literature Review

Electric power supply networks are extensive and intricately linked. Utilities have made a variety of investment decisions across the globe as a result of deregulation, reregulation, restructuring, and the unknown future. The approval procedure for new transmission lines has grown incredibly arduous, pricey, and time-taking over the years [4]. The most effective utilization of the currently available transmission assets is required due to the increasing demand on the current transmission networks. Furthermore, power must be carried across a considerable distance because the producing sites, for instance, in Pakistan are far from the load centers. In this situation, maintaining the security of the power supply, when there are low frequency electro-mechanical oscillations, is the key difficulty [2, 5]. These oscillations, which occur between 0.1 and 1.0 Hz, are mostly caused by the back and forth movement of one group of generators in relation to another [6]. Numerous occurrences, most of which resulted in infamous blackouts [3] impacting vast areas, have been attributed to oscillatory instability. In order for the system to operate within a sufficient stability margin, this compels network operators to take a cautious stance, which results in the inefficient utilization of current assets.

The capability of power systems to keep them stable and guarantee uninterrupted electrical supply to clients in the event of a disturbance is of critical importance [46, 47]. Since the power system is dispersed over wide geographic areas, it is likely that it may experience a variety of faults and failures [48]. Unfortunately, unforeseen errors and cascading events frequently result in blackouts that could have an impact on modern life [3]. Contemporary power systems are operated close to their steady state stability as energy demand rises, which can quickly result in a crisis condition [49–51]. There-

Country/Region	Date	cause(s)
Mexico and USA	September 8, 2011	Transmission line tripping
Brazil	February 4, 2011	Transmission line fault and fluctuated power flow
India	July 30, 2012	Transmission line overload
Philippines	August 6, 2013	Voltage collapse
Bangladesh	November 1, 2014	HVDC station outage
Pakistan	January 26, 2015	Plant technical fault
Turkey	March 31, 2015	Power system failure
Sri Lanka	March 3, 2016	A severe thunderstorm
USA	March 1, 2017	Cascading failure in transmission system
Sudan	January 10, 2018	Cascading failures
Azerbaijan	July 3, 2018	Unexpectedly high temperatures
Brazil	March 21, 2018	Transmission line failure
Canada	December 20, 2018	Winds reached speeds of 100 km/h

Table 2.1: A list of the severe power outages that have been reported globally in the past ten years [3]

fore, in order to handle disruptions, contemporary power systems must be outfitted with proper control and protection mechanisms. Decades of research have been conducted on a power system's capacity to maintain steady state and transient stability [52, 53].

Significant progress has been achieved over the past 40 years in the inclusion of supplemental damping control to increase security margin by removing undesirable oscillations [7–22]. The requirements for reducing these oscillations differ depending on the utility; for instance, in Europe, a settling time of 10–15 seconds is necessary [23]. One of the most practical and affordable methods to increase power system stability is the employment of power system stabilizers (PSS) [7]. In recent times, flexible ac transmission system (FACTS) devices has entered the control domain and has shown to be a long-term, low-cost solution for increasing transmission capacity and improving reliability [24–29]. FACTS devices are practical substitute for building new transmission lines because they enable more precise management of power flow and safe loading of transmission lines close to their thermal limits [30–33]. However, they are installed on transmission lines, these instruments have direct access to the factors influencing

inter-area oscillations.

Typically, the dynamics of a power system is enhanced by modulating the excitation system of generators. For years, PSS have been primarily utilized to mitigate oscillations [34]. For control design, traditional model-based procedures are utilized, which depend on the provision of correct system parameters. This requires detailed information of the contributing components, including load current, voltage, frequency, and others [35]. It can be challenging to gather these crucial parameters in a real-time setting. The interconnection of the power system will become more complex as a result of independent power producers (IPPs). The study eliminates the need for exact information about system components by concentrating on control design based on system input-output measurements.

Although traditional control strategies are simple, they have a limited working range. Traditional non-adaptive designs are constructed close to the nominal operating point, they may produce oscillations that are poorly damped or even unstable, due to uncertainties and drastic changes in operating conditions [28]. Use of robust control techniques can increase the performance range of such a linear controller [36–38]. However, with significant disturbance, the post-disturbance system may deviate significantly from its nominal operational states and may even outperform the robust controllers performance range. The key problem for network operators is the migration of system to an unanticipated operating condition or network topology after a disruption, such as a bus malfunction followed by a transmission network loss. A self-tuning controller is needed to maintain proper damping throughout different operating points. To increase stability, the controller must quickly recognize electro-mechanical oscillations, determine the optimal controller settings, and implement the necessary control actions, such as damping the least stable modes.

Recently, initiatives have been made to enhance FACTS device control. Uptil now, a linearized power system model has been used to create the PSS/FACTS controller parameters. To adequately dampen at a variety of operating points, the settings of FACTS devices must be fine-tuned in response to the oscillations. With time-varying parameters, power systems are highly non-linear, and a specified control scheme based on the linear model may not provide sufficient performance under a variety of operating conditions [39]. Additionally, it might not be possible to limit the controller to the linear

domain, especially in high-stress circumstances [40]. Because of this, a non-linear self-tuning controller that takes into account system non-linearities and adapts to altering operating conditions may result in superior outcomes [41].

Power systems have highly non-linear behaviour, and stressful operating circumstances amplify the effects. On the other hand, linear controllers are usually built to provide acceptable performance in a single operating circumstance. Additionally, it calls for accurate system parameter knowledge, which is generally challenging to attain. To ensure that the controller is "self-tuned" in each working state, reducing the dependency on a correct system model, many adaptive procedures have been developed [54–56]. Similar approaches have been utilised for applications involving the power system. For power system stabilisers (PSS) [7] and flexible ac transmission systems (FACTS) devices [28], self-tuning control based only on observed signals has been proposed to address some of the difficulties of model-based designs. Self-tuning control scheme, has guaranteed better performance as compared to classical control schemes [57–59].

In order to predict the oscillatory behaviour of the power system using least square methods, the auto-regressive with an external input (ARX) form or standard neural network (NN) structures are frequently utilised [60, 61]. For linear control based on approximation models, pole-shifting controllers have received a lot of support [62]. On the other hand, the existence of non-linearities in the measured signal can affect the performance of the linear controller [63]. Although numerous researchers have created neural network-based non-linear approximators and controllers [39, 64–67], it is challenging to obtain a standard non-linear control framework.

The auto-regressive with external input (ARX) model is commonly used in linear system identification, in combination with the RLS algorithm [68–70]. With this approach, the neural network structure can be used, however the model must be expressed as linear in parameter [71]. These strategies may not always produce satisfactory results due to the substantial non-linearities in power systems. The non-linear model and precise parameter estimation are therefore essential for the representation of the power system. Because of their ability to generalise and learn, neural networks are frequently used to explain non-linear systems [39]. In recent years, online estimation of input-output mapping of non-linear systems has been reported using different frameworks of artificial neural networks (ANN) [72] that includes, multi-layer perceptron (MLP),

Elman neural network (ENN), radial basis function (RBF), recurrent neural network (RNN) and simultaneous recurrent neural network (SRN) [26,39,73–76]. To update the neural network parameters online, these methods commonly use back-propagation (BP) or back-propagation through time (BPTT). These algorithms can quickly adjust the settings and return the system to normal because of their capacity to learn and adapt to system changes. The back-propagation algorithm has limitations in terms of convergence speed and accuracy [26,39,75] and its learning process is slow [77]. However, Levenberg-Marquardt [78] algorithm used in sliding window batch mode, allow model parameters to converge faster and with more precision than the back-propagation approach [79]. Levenberg-Marquardt (LM) algorithm has the benefit to save training time and to reduce error oscillations [80]. Recent developments in modifying Levenberg-Marquardt (LM) algorithm, for better performance has proven to be more effective [81,82].

Chapter 3

Methodology

3.1 Self-Tuning Control - An Overview

The topic intents to provide a summary of self-tuning control (STC), which includes the identifier and controller. Modeling, estimation, and control design techniques make up the self-tuning control's overall framework.

3.1.1 Basic Concepts

The majority of industrial processes are stochastic in nature. Traditional controllers for these systems often work with specific parameters and are inadequate because variation is always present. This is because of how things are made, the materials used, the fuel used, the usage of machinery, etc [83]. Applying self-tuning control is one way to enhance the standard of control for such processes.

The basic purpose of a self-tuning control is to develop an algorithm that will automatically adjust the model's parameters to satisfy the intended specifications. Performance is achieved using an adjustment mechanism that keeps track of the system and the controller's associated coefficients.

The dynamics of systems can change in control engineering and these changes can happen at any time and might not be accounted for by a robust design. The adjustment mechanism, however, offers a means of adjustment to system change in self-tuning control. The controller is also simple to tune if the system changes in future. For example, The change in flow control is the most frequent control loop in the process industry.

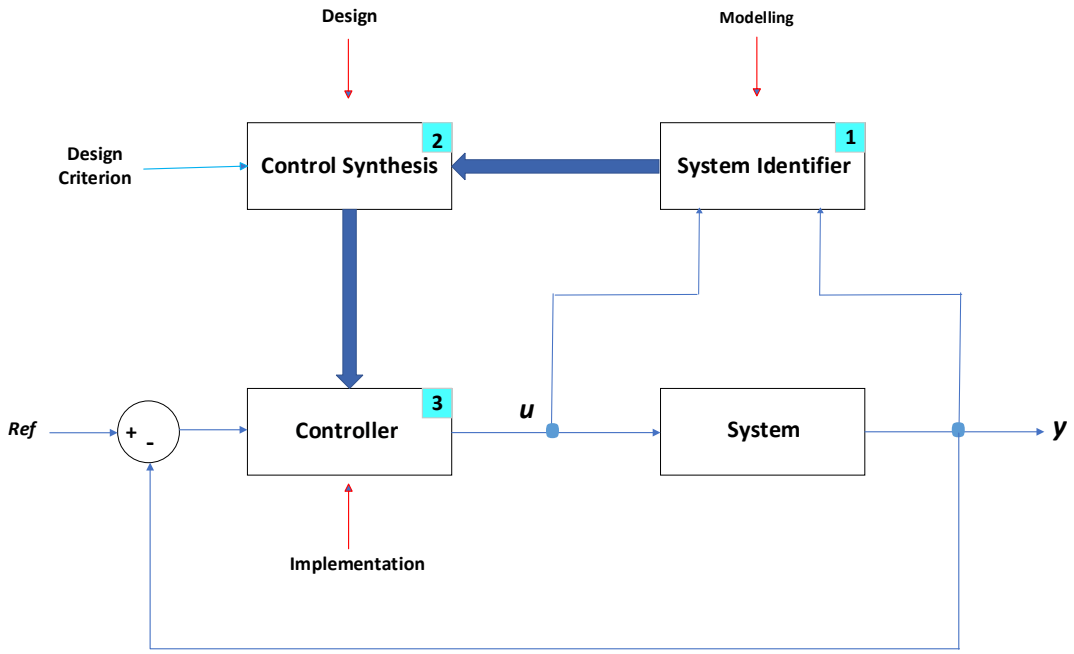

Figure 3.1: Self-tuning control scheme

The practice of replacing valves with differing specifications is prevalent. Such a loop's self-tuning control automatically tunes to updated valve specifications.

3.1.2 Composition of Self-tuning Control

Automating the tuning of the control systems is the goal of self-tuning control. The following are the key phases in the creation of such control systems:

1. System's behavior modeling.
2. Controller design.
3. Controller implementation.

The idea of self-tuning control is demonstrated in Figure 3.1, where the controller is given the estimated system parameters in order to produce the desired response. System identifier, control synthesis, and a controller are the related components of self-tuning control. Utilizing the available input-output measurements, the modeling and identification of the system behavior constitute the initial phase. This is used to estimate parameters of the controller and get the system to respond as required [84]. It is important to remember that the controller's parameters are determined by extrapolating the system parameter estimates from the real system parameters.

Depending on one's viewpoint, system modeling and simulation can have a variety of different meanings. For a system designer, simulation often entails utilizing a slightly reduced model of the system that is computationally simple, efficient and physically realistic enough to represent the behavior of the system. Designing model of a system is essentially, a combination of empirical techniques and physical laws based on the system's observable behaviour. System identification is the process of creating a model using measured input and output data [85]. The desired controller intends to make use of the mathematical model of the fundamental information produced by the system identification. Synthesis is carried out in accordance with an objective function and may adhere to design criteria or algorithms. These are algorithms that result in a certain controller from a particular model representation [60]. This strategy, called "self-tuning control" is employed in this research work. The three steps that make up the design approach are as follows:

- Step 1: Model employed in System Identification
- Step 2: Estimation Algorithm
- Step 3: Control Design

These three steps are described in the next sections.

3.2 Model employed in System Identification

3.2.1 Neural Network (NN) - An Overview

An artificial neural network (ANN) is a computational model that resembles the behaviour of nerve cells in the human brain. Artificial neural network (ANN), sometimes referred as neural network (NN) make use of learning algorithms that enable them to freely adjust or learn as they process new data. Neural network is incredibly effective and expanding quickly in the numerous applications, such as those requiring information processing, learning, and adaption activities. Following the publication of Defense Advanced Research Projects Agency (DARPA), neural network study report in 1988, NN applications in the fields of aerospace, automotive, defense, electronics, robotics, oil and gas, and other fields are growing swiftly [78]. Mostly, NN is an adaptable system

that modifies its structural parameters in response to data entering or leaving the network. In contrast to the biological role model, components of mathematics, statistics, and optimization are pursued in our study. A tool for simulating complex interactions between inputs and outputs is provided by the modelling of non-linear data using neural networks. The significant characteristics of NN are:

- It has a large number of neurons, which are interlinked processing units, that perform all operations.
- The basic unit of information stored in neurons is the weighted link of neurons.
- Through links and linking weights, the input signals reach the processing units.
- By appropriately assigning and adjusting weights, it may learn from the provided data and recall it as well as make generalisations.

Depending on the type of application, the structures of neural network are simple to modify into a required form. A basic NN structure consists of:

- Input layer
- Hidden layer(s)
- Output layer

An input neuron is a node at the input, and the input layer is made up of all input neurons. An output neuron is a node at the output, and the output layer is made up of output neuron(s). The hidden layer is the collective term for all of the central neurons that realize the activation function known as hidden layer neurons. Through input and output links known as input/output weights, these three layers i.e. input layer, hidden layer(s), and output layer are interconnected.

Figure 3.2 shows the basic structure of neural network (MLP) with a bias, a hidden layer and an output. This topology is termed as '$h-1$' feed-forward MLP neural network where $\tilde{y}(k+1)$ is the estimated output. 'h' is the number of neurons in hidden layer and has a single node in output layer. This structure is capable to perform non-linear input-output mapping $\mathbf{R}^{n+m} \rightarrow \mathbf{R}$. Every connection made from one layer to the next is represented by a synaptic weight matrix for that layer [71].

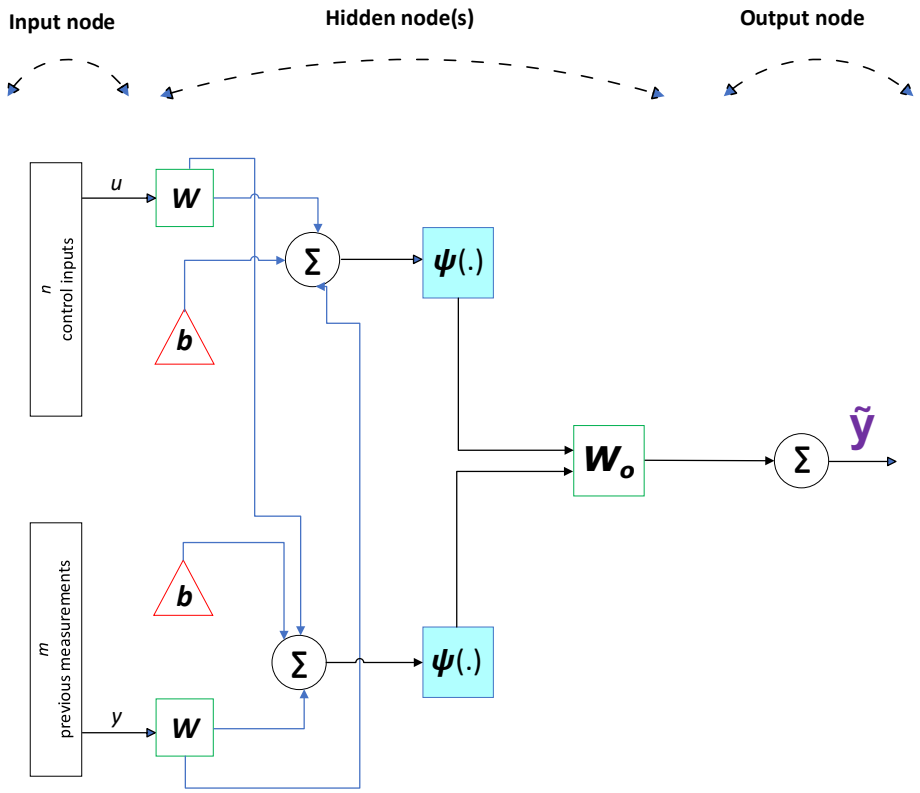

Figure 3.2: A Basic Structure of Neural Network (MLP)

'n' is the number of past control inputs and 'm' is the number of past measurements. 'W_{ij}' are the weights of input layer. 'W_{o_i}' are the weights of output layer. 'b_i' is the vector of bias. For $i = 1, 2, ..., h$ and $j = 1, 2, ..., n + m$.

In basic function formulation, the MLP can be written as:

$$\tilde{y}(k+1) = \sum_{i=1}^{h} W_{o_i} \times \Psi_i\{b_i + (\sum_{j=1}^{n+m} W_{ij} \times X_j)\} \tag{3.2.1}$$

where $X_j = [\ u_1\ \ u_2\ \ .\ .\ u_n\ \ \ y_1\ \ y_2\ \ .\ .\ y_m\ ]$ is the NN input vector and Ψ is the activation function in the hidden layer. Thr neuron in hidden layer use non-linear logistic function defined as:

$$\Psi(\chi) = \frac{1}{1 + e^{-(\chi)}} \tag{3.2.2}$$

where

$$\chi = \sum_{j=1}^{n+m} W_{ij} \times X_j\ \ \epsilon \mathbf{R}$$

Define:

$$\Psi \triangleq [\ \psi_1\ \ \psi_2\ \ .\ .\ \psi_h\ ]^T\ \ \epsilon \mathbf{R^h} \tag{3.2.3}$$

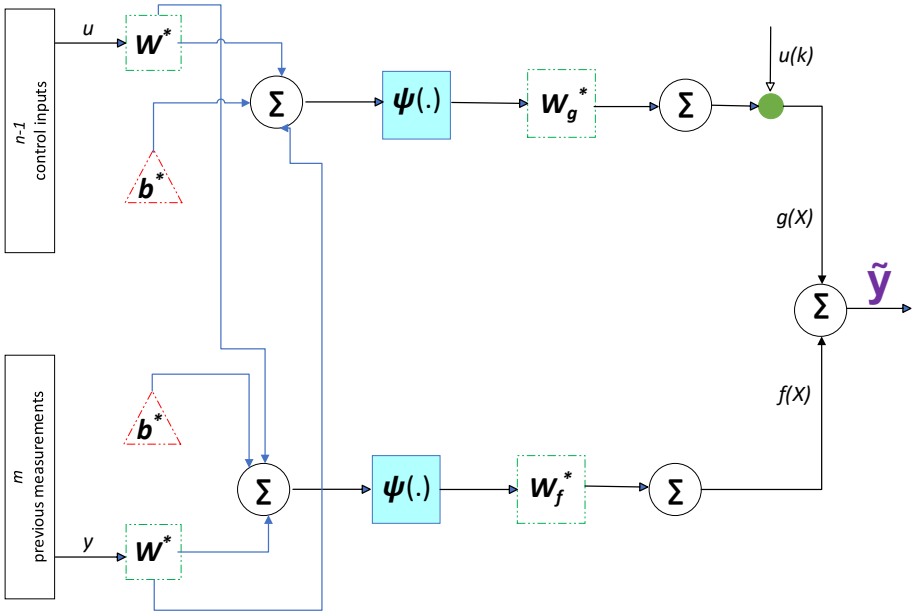

Figure 3.3: Computationally Efficient Neural Network (CENN)

and

$$\mathbf{W_o}^{\mathbf{T}} \triangleq [\ \mathbf{W_{o_1}}\ \ \mathbf{W_{o_2}}\ \ .\ .\ \mathbf{W_{o_h}}\ ]\ \epsilon\, \mathbf{R^h} \tag{3.2.4}$$

by substituting (3.2.3) and (3.2.4), (3.2.1) becomes:

$$\tilde{y}(k+1) = \mathbf{W_o}^{\mathbf{T}} \mathbf{\Psi} \tag{3.2.5}$$

For updating the parameters of NN, we have the following options:

1. Update all the input weights W_{ij} and output weights W_{o_i}. The output $\tilde{y}$ in (3.2.5) is a non-linear function of the parameters W_{ij}. This is owing to the selection of non-linear activation functions in the hidden layer.

2. Since (3.2.5) is linear in parameter, so the output weights are just tuned in this technique.

$$\mathbf{W_o}^{\mathbf{T}} = \mathbf{W_{o_i}} \tag{3.2.6}$$

The output $\tilde{y}$ in (3.2.5) is a linear function of the parameter vector $\mathbf{W_o}$.

3.2.1.1 Computationally Efficient Neural Network (CENN)

The feed-forward MLP NN's incompatibility with the traditional non-linear control framework is a challenge. To address this challenge, a non-linear model that is com-

putationally efficient and is compatible with feedback linearizable controller (FBLC) is proposed and is referred as computationally efficient neural network (CENN) shown in Figure 3.3.

The output of computationally efficient neural network (CENN) with a single hidden layer and a single output with bias can be expressed as in (3.2.7).

$$\tilde{y}(k+1) = {W_o}^* \times \Psi\{b^* + (\mathbf{W}^* \times [\bar{u}, \bar{y}])\} \tag{3.2.7}$$

where, $\tilde{y}(k+1)$ is the estimated output

$${W_o}^* = [\ {W_g}^*{}_{(1,n-1)}\ \ {W_f}^*{}_{(1,m)}\ ]$$

$$b^* = [\ b_1^*\ \ b_2^*\ .\ .\ b_h^*\ ]$$

$$\mathbf{W}^* = \begin{bmatrix} W_{11}^* & W_{12}^* & \cdots & \cdots & W_{1(n-1+m)}^* \\ W_{21}^* & W_{22}^* & \cdots & \cdots & W_{2(n-1+m)}^* \\ \vdots & \vdots & \cdots & \cdots & \vdots \\ W_{h1}^* & W_{h2}^* & \cdots & \cdots & W_{h(n-1+m)}^* \end{bmatrix}$$

where:

h = No. of hidden layer neurons of size, $\mathbf{R}^{n-1+m}$

$\bar{u} = [\ u_1\ \ u_2\ .\ .\ u_{n-1}\ ]^T$

$\bar{y} = [\ y_1\ \ y_2\ .\ .\ y_m\ ]^T$

Ψ = hidden layer activation function (log-sigmoid)

m = no. of past measurements

n = no. of past control inputs

${W_f}^*$ = output weights to function *f(x)*

and ${W_g}^*$, output weights to function *g(x)* is kept constant.

3.3 Estimation Algorithm

When using self-tuning control, parameters are estimated iteratively, allowing the estimated model to be updated as soon as new input-output data become available at

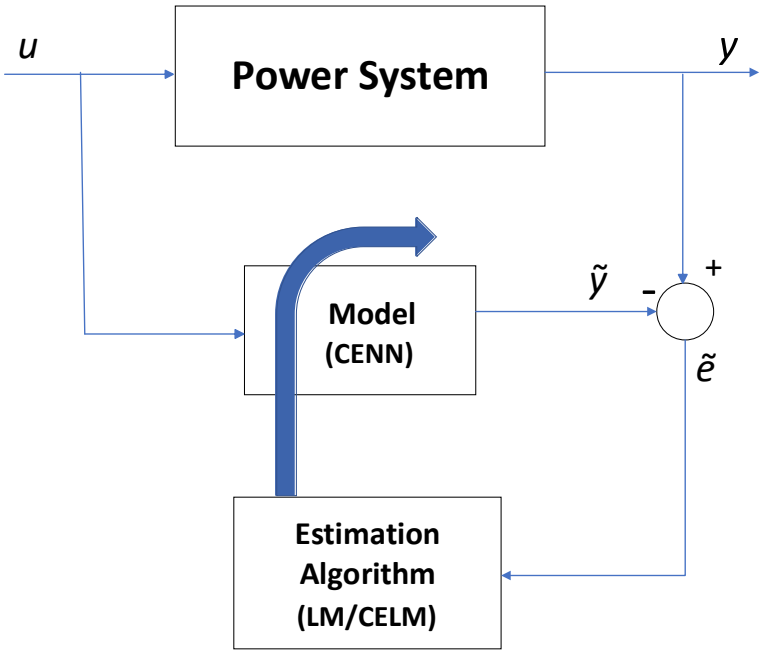

Figure 3.4: Model estimation scheme

sampling moment 'k'. Model estimation scheme is shown in Figure 3.4, which compares the observed output '$\tilde{y}$' with the measured output 'y' to produce an error $\tilde{e}(k)$. The estimation algorithm LM and CELM use this error to update the model parameters at each sampling instant.

3.3.1 Levenberg-Marquardt (LM) Algorithm

For the least squares estimate of non-linear parameters, the damped Gauss-Newton approach was proposed by Levenberg in 1944 and Marquardt in 1963 [86]. LM, which combines the steepest descent and Gauss-Newton methods in a unique way, has recently emerged as a formal approach for solving non-linear least squares problems. It is an iterative approach that is stated as the sum of squares of non-linear functions and is typically used for off-line optimization [86]. This approach behaves like a Steepest-Descent method when the present solution is far from the correct one, yet it still converges. On the other hand, it turns into a Gauss-Newton method when the present solution reaches close to the correct one [78].

The aim is to create an algorithm that minimizes the cost function $f_c(x)$, optimizes the performance index, and determines the value of the parameter vector x. The Newton method will be evaluated in the following in order to derive the LM algorithm because

of its higher performance and usage of precise knowledge about second order derivative. The great benefit of this method is that the Hessian is only approximated using the first order derivative.

The typical Newton method, based on second order Taylor's series is expressed as:

$$f_c(x_{k+1}) = f_c(x_k + \Delta x_k \approx f_c(x_k) + \nabla {f_c}^T(x)|_{x=x_k}\Delta x_k + \frac{1}{2}\Delta {x_k}^T \nabla {f_c}^2(x)|_{x=x_k}\Delta x_k \tag{3.3.1}$$

The stationary point of $f_c(x)$ is located here as:

$$\nabla {f_c}^T(x_k) + \nabla {f_c}^2(x_k)\Delta x_k = 0$$

solving for Δx_k produces

$$\Delta x_k = -[\nabla {f_c}^2(x_k)]^{-1}\nabla {f_c}^T(x_k)$$

since

$$\Delta x_k = x_{k+1} - x_k$$

Hence, the typical Newton method becomes:

$$x_{k+1} = x_k - [\nabla {f_c}^2(x_k)]^{-1}\nabla {f_c}^T(x_k) \tag{3.3.2}$$

Assuming $f_c(x)$ is a sum of square function and we find the vector 'x^+' which minimizes the squared distance $e^T e$, stated as:

$$f_c(x) = \sum_{i=1}^{d} {e_i}^2(x) = e^T(x)e(x) \tag{3.3.3}$$

where 'd' is number of elements in data set. A non-linear optimization technique such as LM has to be used, if the gradient of the cost function is non-linear in parameter 'x', to search for the optimal parameter 'x^+'. The gradient of j^{th} element would be:

$$|\nabla f_c(x)|_j = \frac{\partial f_c(x)}{\partial x_j} = 2\sum_{i=1}^{d} e_i(x) \cdot \frac{\partial e_i(x)}{\partial x_j}$$

In matrix form:

$$\nabla f_c(x) = 2\mathbf{J^T}(\mathbf{x})\mathbf{e}(\mathbf{x}) \tag{3.3.4}$$

where,

$$\mathbf{J} = \begin{bmatrix} \frac{\partial e_1(x)}{\partial x_1} & \frac{\partial e_1(x)}{\partial x_2} & \cdots & \frac{\partial e_1(x)}{\partial x_n} \\ \frac{\partial e_2(x)}{\partial x_1} & \frac{\partial e_2(x)}{\partial x_2} & \cdots & \frac{\partial e_2(x)}{\partial x_n} \\ \vdots & \vdots & \vdots & \\ \frac{\partial e_d(x)}{\partial x_1} & \frac{\partial e_d(x)}{\partial x_2} & \cdots & \frac{\partial e_d(x)}{\partial x_n} \end{bmatrix}$$

Now, find the Hessian matrix of (3.3.2). The k, j element of the Hessian would be:

$$|\nabla {f_c}^2(x)|_{k,j} = \frac{\partial f_c^2(x)}{\partial x_k \partial x_j} = 2\sum_{i=1}^{d} \left(\frac{\partial e_i(x)}{\partial x_k} \cdot \frac{\partial e_i(x)}{\partial x_j} + e_i(x) \cdot \frac{\partial^2 e_i(x)}{\partial x_k \partial x_j} \right)$$

In matrix form:

$$\nabla^2 f_c(x) = 2\mathbf{J^T(x)J(x)} + \mathbf{2Q(x)}$$

where,

$$\mathbf{Q(x)} = \sum_{\mathbf{i=1}}^{\mathbf{d}} \mathbf{e_i(x)} \cdot \nabla^{\mathbf{2}} \mathbf{e_i(x)}$$

Assuming $\mathbf{Q(x)}$ is small and can be ignore, then the approximated Hessian becomes:

$$\nabla^2 f_c(x) = 2\mathbf{J^T(x)J(x)} \tag{3.3.5}$$

by Substituting (3.3.4) and (3.3.5) into (3.3.2), we get:

$$x_{k+1} = x_k - [2\mathbf{J^T(x)J(x)}]^{-\mathbf{1}} \mathbf{2J^T(x)e(x)}$$

$$= x_k - [\mathbf{J^T(x)J(x)}]^{-\mathbf{1}} \mathbf{J^T(x)e(x)} \tag{3.3.6}$$

This approach may not yield the invertible matrix $\mathbf{J^T J}$. To solve this issue, we change the (3.3.6) into a form that ultimately results in the Levenberg-Marquardt (LM) algorithm:

$$x_{k+1} = x_k - [\mathbf{J^T(x)J(x)} + \lambda \mathbf{I}]^{-\mathbf{1}} \mathbf{J^T(x)e(x)} \tag{3.3.7}$$

where, 'λ' is the learning rate. LM acts like a steepest descent algorithm when λ tends to increase. We get:

$$x_{k+1} = x_k - \frac{1}{\lambda} \mathbf{J^T(x)e(x)}$$

Also, it acts like a Gauss-Newton method when λ tends to zero.

An effective way to select the value of λ is that, a small value is chosen initially and if this value of λ does not produce the smaller value in the cost function expressed in (3.3.3), then the value of λ is increased further to a factor $v > 1$. This results in decrement of the cost function $f_c(x)$ and LM algorithm becomes steepest-descent method. by increasing the value of λ, if it produces smaller value in the cost function $f_c(x)$, then λ is divided by a factor $v > 1$. This converts the LM algorithm into a Gauss-Newton method. As a result, the LM algorithm achieves a good balance between the assured 'convergence' of the steepest- descent method and the 'speed' acquired by the Gauss-Newton method [78].

3.3.1.1 Working of CENN-LM and CENN-CELM

The CENN updating equation is given as:

$$\tilde{y}(k+1) = f(\bar{u}, \bar{y}) + [g(\bar{u}, \bar{y}) \times u(k)] \tag{3.3.8}$$

For LM, subject to:

$$||\Delta P|| \leq \epsilon_2 \times ||P|| + \epsilon_2$$

where,

$$P = \left[\begin{array}{cccccccccc} W^*_{i1} & W^*_{i2} & \cdots & W^*_{ih} & ; & b^*_1 & \cdots & b^*_h & ; & W^*_{f1} \quad \cdots \quad W^*_{fm} \end{array} \right]$$

$$For, \;\; i = 1, ..., h \quad and \quad h \, \epsilon \, \mathbf{R}^{n-1+m}$$

For CELM, subject to:

$$||\Delta W^*|| \leq \epsilon_2 \times ||W^*|| + \epsilon_2$$
$$||\Delta b^*|| \leq \epsilon_2 \times ||b^*|| + \epsilon_2$$
$$||\Delta W^*_f|| \leq \epsilon_2 \times ||W^*_f|| + \epsilon_2$$

The equation (3.3.8) can be rewritten as:

$$\begin{aligned} \tilde{y}(k+1) = & \sum_{i=n+1}^{h} [W^*_{o_i} \times \Psi_i \{{b_i}^* + \sum_{j=1}^{n-1+m} (W^*_{ij} \times X_j)\}] \\ & + \sum_{i=1}^{n-1} [W^*_{o_i} \times \Psi_i \{{b_i}^* + \sum_{j=1}^{n-1+m} (W^*_{ij} \times X_j)\} \times u(k)] \end{aligned} \tag{3.3.9}$$

where: $h =$ no. of hidden layer neurons of size, $\mathbf{R}^{n-1+m}$.
$X = [\bar{u}, \bar{y}]^T, u = [\, u(k-n+1),\ u(k-n),\ .\ .\ u(k)\,]$ is the vector of previous n control inputs and $y = [\ y(k-m+1)\ y(k-m)\ .\ .\ y(k)\ ]$ is the vector of past m measurements. $\Psi(.)$ is a non-linear activation function, here, $\Psi(.) = log - sigmoid(.) = \frac{1}{1+e^{-(.)}}$.
The estimated output in (3.3.9) can be expressed as (3.3.10).

$$\tilde{y}(k+1) = W_f{}^* \times \Psi_f(\bar{u}, \bar{y}) + [W_g{}^* \times \Psi_g(\bar{u}, \bar{y})] \times u(k) \tag{3.3.10}$$

Case - I: CENN in conjunction with typical LM

Here, batch mode operation with sliding window for online estimation is carried out using the standard LM technique. First, a suitable window size is chosen.
The error vector $\tilde{e}$ over a window containing w_{size} samples is given by:

$$\tilde{e} = \begin{bmatrix} y(k+1) - \tilde{y}(k+1) \\ \\ y(k) - \tilde{y}(k) \\ \\ \vdots \\ \\ y(k - w_{size} + 1) - \tilde{y}(k - w_{size} + 1) \end{bmatrix} \tag{3.3.11}$$

where, $y(.)$ is the actual output, $\tilde{y}(.)$ is the estimated output and w_{size} is the number of samples in a window.

The weights that need to be updated (unknown parameters) are placed in a form of a vector P as follows in (3.3.12) to calculate the error derivatives throughout an entire window:

$$P = \left[\ W_{i1}^*\ W_{i2}^*\ \cdots\ W_{ih}^*\ ;\ b_1^*\ \cdots\ b_h^*\ ;\ W_{f1}^*\ \cdots\ W_{fm}^*\ \right] \tag{3.3.12}$$

where, $\quad i = 1, ..., h \quad$ and $\quad h\,\epsilon\,\mathbf{R}^{n-1+m}$

The total number of unknown parameters is given by: $U_p\,\epsilon\,\mathbf{R}^{(h\times h)+h+m}$, which is the size of the parameter vector P.
The corresponding error derivatives for weight update equation can be written as:

$$\mathbf{J} = \frac{\partial \tilde{e}}{\partial P}$$

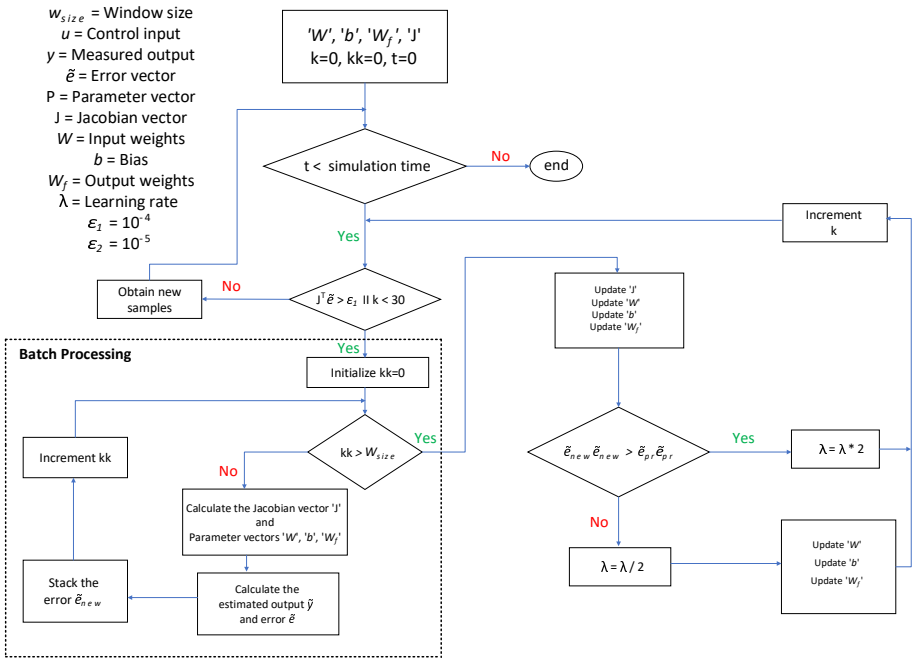

Figure 3.5: Typical LM scheme (CENN-LM)

$$= \begin{bmatrix} \frac{\partial\tilde{y}(k+1)}{\partial P(1)} & \frac{\partial\tilde{y}(k+1)}{\partial P(2)} & \cdots & \frac{\partial\tilde{y}(k+1)}{\partial P(U_p)} \\ \\ \frac{\partial\tilde{y}(k)}{\partial P(1)} & \frac{\partial\tilde{y}(k)}{\partial P(2)} & \cdots & \frac{\partial\tilde{y}(k)}{\partial P(U_p)} \\ \\ \vdots & \vdots & \vdots & \\ \\ \frac{\partial\tilde{y}((k-w_{size}+1)}{\partial P(1)} & \frac{\partial\tilde{y}((k-w_{size}+1)}{\partial P(2)} & \cdots & \frac{\partial\tilde{y}((k-w_{size}+1)}{\partial P(U_p)} \end{bmatrix} \tag{3.3.13}$$

The derivatives for each sample within a window are calculated with respect to each adjustable parameter and stacked to form a matrix, as shown in (3.3.13). The weight parameters are then updated according to (3.3.14).

$$P_{updated} = P_{previous} + [\mathbf{J}^{\mathrm{T}}\mathbf{J} + \lambda I]^{-1}\mathbf{J}^{\mathrm{T}}\tilde{e} \tag{3.3.14}$$

where:

$$\Delta P = [\mathbf{J}^{\mathrm{T}}\mathbf{J} + \lambda I]^{-1}\mathbf{J}^{\mathrm{T}}\tilde{e}$$

The flowchart in Figure 3.5 shows how the typical LM algorithm update equations (3.3.13)-(3.3.14) to train the computationally efficient neural network (CENN) numerous times for a single window.

Each sliding window receives an online update. *The weights of neural network, stacked in a vector are initialized at the beginning of the first window. A fixed window size w_{size} applies to each sample, and the output of neural network is computed. later, the estimated samples are compared with actual measured samples throughout the entire window to create an error vector ($\tilde{e}$). The new squared error over the entire window $\tilde{e}_{new}\tilde{e}_{new}$ is then compared with the previous squared error $\tilde{e}_{pr}\tilde{e}_{pr}$. If $\tilde{e}_{new}\tilde{e}_{new}$ is less than $\tilde{e}_{pr}\tilde{e}_{pr}$, the value of λ is divided by 2 and the weights update is accepted otherwise, the value of λ is multiplied by 2 without updating the weights. The iteration for the window is continued until the convergence criteria is met. During the first few windows, the convergence is slower but it accelerates once the weight parameters get stable.*

Case - II: CENN in conjunction with CELM

Here typical LM algorithm is reshaped in such a way that, instead of updating P on the whole, we update input weights 'W^*', bias 'b^*' and output weights 'W_f^*' separately as shown in (3.3.15) to (3.3.20). This approach guarantees better convergence than updating P collectively. In this scenario, the weights once converges to the defined criteria will not update further, and this results in less computations.

From (3.3.14):

$$\Delta P = [\mathbf{J}^{\mathrm{T}}\mathbf{J} + \lambda I]^{-1}\mathbf{J}^{\mathrm{T}}\tilde{e}$$

$$\Delta W^* = \sum_{i=1}^{h\times h} \Delta P_i \tag{3.3.15}$$

$$\Delta b^* = \sum_{i=(h\times h)+1}^{(h\times h)+h} \Delta P_i \tag{3.3.16}$$

$$\Delta W_f^* = \sum_{i=(h\times h)+h+1}^{U_p} \Delta P_i \tag{3.3.17}$$

where, $U_p \, \epsilon \, \mathbf{R}^{(h\times h)+h+m}$

$$W_{new}^* = W^* + \Delta W^* \tag{3.3.18}$$

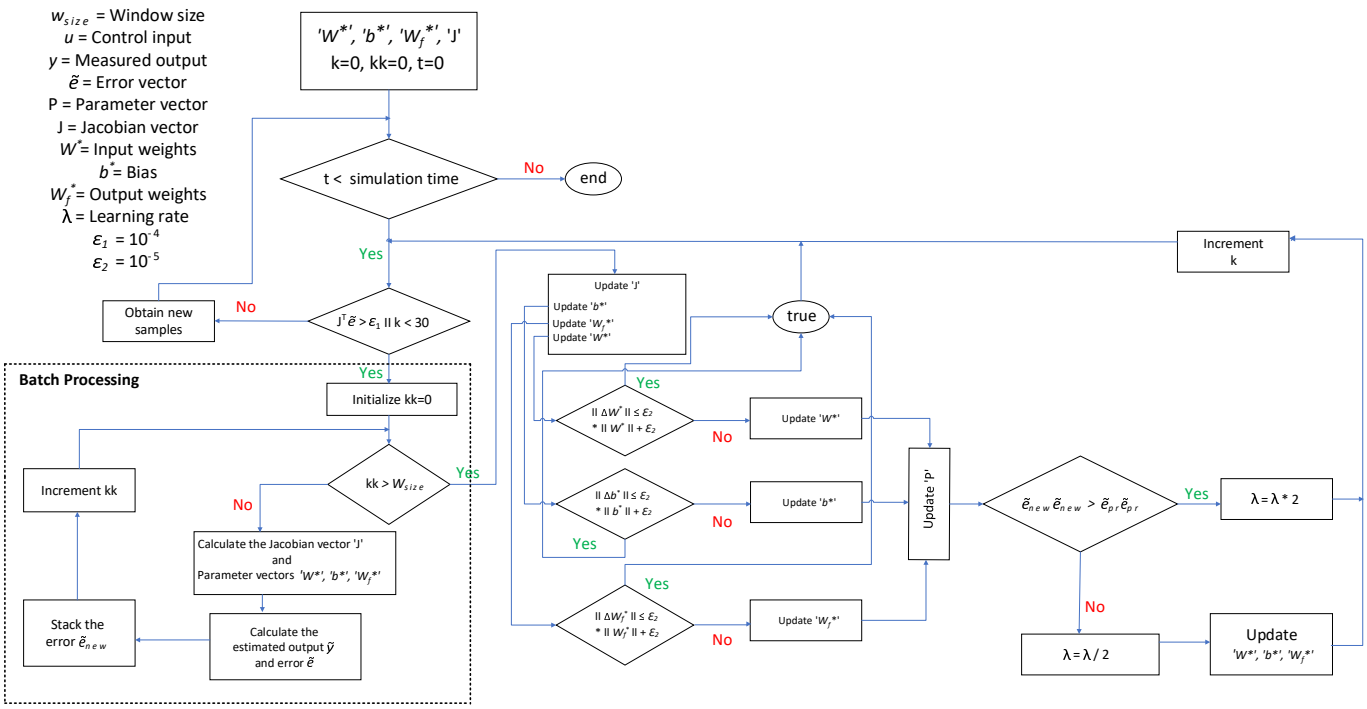

Figure 3.6: Computationally Efficient LM scheme (CENN-CELM)

$$b^*_{new} = b^* + \Delta b^* \tag{3.3.19}$$

$$W^*_{f_{new}} = W^*_f + \Delta W^*_f \tag{3.3.20}$$

Each movable window receives an online update as shown in Figure 3.6. After updating input weights, bias and output weights, these are combined to form updated parameter vector P_{new} as shown in (3.3.21).

$$P_{new} = [\ W^*_{new}\ \ b^*_{new}\ \ W^*_{f_{new}}\ ] \tag{3.3.21}$$

The rest of the algorithm works exactly same as previously discussed in section 3.3.1*.

3.4 Controller Design

A linearized model around a theoretical operating point may serve as the foundation for the controller design. An exciting topic of controller design study is determining what characteristics of a model make it suitable for control and how to effectively recognize them. In order to solve the reference tracking problem, self-tuning control can be used in a variety of ways. This Section presents a feedback linearization controller (FBLC),

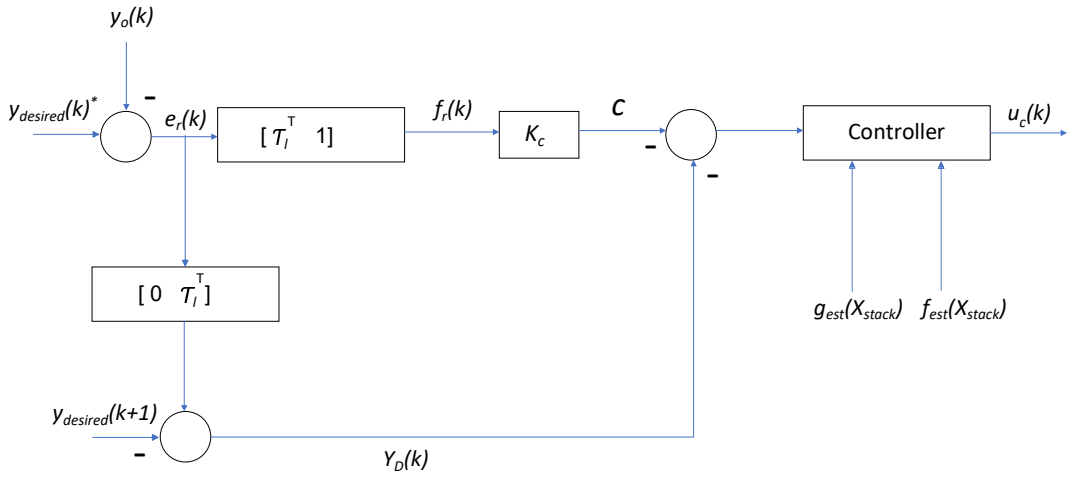

Figure 3.7: A general structure of Feedback Linearizable Controller (FBLC) [1]

an existing control strategy, for regulating system output. The goal is to investigate the design options for the inter-area oscillation damping in power systems.

3.4.1 Feedback Linearizable Controller (FBLC) [1]

A control approach known as feedback linearization is the conversion of the original non-linear model into an approximate linear model with a more straightforward form. It is applicable to the class of non-linear systems. Choosing a model to represent the dynamics of the system in the desired frequency range is the first stage in creating self-tuning control [87].

According to Figure 3.7, feedback linearization cancels out the non-linearities in a non-linear system, resulting in a linear closed-loop dynamics [87, 88]. It is applicable to a class of non-linear systems known as:

$$y_o(k+1) = f_{est}(X_{stack}(k)) + g_{est}(X_{stack}(k))u_c(k) \tag{3.4.1}$$

where $X_{stack} = [\bar{u}\ \bar{y}]^T$ and $\bar{u}$, $\bar{y}$ are control inputs and past measurements vectors, respectively. The non-linear functions $f_{est}(X_{stack})$, $g_{est}(X_{stack})\ \epsilon\, \mathbf{R}^n \rightarrow \mathbf{R}$ are assumed to be unknown.

Deriving the control action 'u_c' in this case is the goal in order to ensure that the plant accurately follows the intended trajectory $y_{desired}(k)$. The following are the definitions of the intended trajectories envelop over windows:

$$y_{desired}(k)^* \triangleq [\, y_{desired}(k-m+1) \ . \ . \ y_{desired}(k-1) \ \ y_{desired}(k)\,]^T$$

In this practical design, following assumption is used:

$$g_{est}(X_{stack}) > 0.$$

An error vector is defined as:

$$e_r(k) = y_{desired}(k)^* - y_o(k) \tag{3.4.2}$$

where,

$$y_o(k) = [\, y_o(k-m+1) \;\; . \;\; . \;\; y_o(k-1) \;\; y_o(k) \,]^T$$

A filter error is defined as:

$$f_r(k) = [\tau_l{}^T \;\; 1\,] e_r(k) \tag{3.4.3}$$

where,

$$\tau_l = [\, \tau_{l1} \;\; \tau_{l2} \;\; \cdots \;\; \tau_{lm-1} \,]^T$$

is appropriately chosen coefficient vector such that $e_r(k)$ tends to 0 as $f_r(k)$ tends to 0 is stable. Then, the (3.4.1) can be written in term of filtered error:

$$f_r(k+1) = f_{est}(X_{stack}) + g_{est}(X_{stack})u_c + Y_D \tag{3.4.4}$$

where,

$$Y_D \approx -y_{desired}(k+1) + [\, 0 \;\; \tau_l^T \,]\, e_r(k)$$

Hence, the control law is given by:

$$u_c = \frac{1}{g_{est}(X_{stack})}[\, -f_{est}(X_{stack}) - K_c f_r - Y_D \,] \tag{3.4.5}$$

It would result in $f_r(k)$ tending to zero for any positive K_c. Since the assumption is that, these functions are estimated by LM/CELM algorithm, we can choose a control signal:

$$u_c = \frac{1}{g_{est}(X_{stack})}[\, -f_{est}(X_{stack}) + c \,] \tag{3.4.6}$$

where,

$$c = -K_c f_r - Y_D \tag{3.4.7}$$

The $f_{est}(X_{stack})$ and $g_{est}(X_{stack})$ estimates will be created by neural network. Note that when $g_{est}(X_{stack}) = 0$, the control law (3.4.6) is not accurately described. As a result, one must be cautioned to ensure the boundedness of the controller. Fixing the estimate $g_{est}(X_{stack})$ constant, is a simple way to overcome this issue.

Chapter 4

Results and Discussions

4.1 Evaluation - I: Estimation with CENN-LM and CENN-CELM

In evaluation - I, the estimation of non-linear computationally efficient neural network (CENN), which is trained through online standard Levenberg-Marquardt (LM) algorithm is carried out and later, it is compared with the estimation of non-linear computationally efficient neural network (CENN), which is trained through modified online computationally efficient Levenberg-Marquardt (CELM) algorithm. This test is carried out on the 4-machine 2-area power system as shown in Figure 1.2. A line between buses 7 - 8 is taken out at 5s. Following a disturbance at bus 8, results in oscillations on the line 10 - 9. The output data is power flowing through line 10 - 9, and the input signal is a square wave at 0.01Hz. The numerical values of the parameters used in the estimation of CENN-LM and CENN-CELM is shown in table 4.1.

Parameter	Value	Equation
m	6	3.3.9
n	6	3.3.9
h	11	3.3.9
w_{size}	25	3.3.11
λ	0.1	3.3.14

Table 4.1: Parameters used in evaluation - I

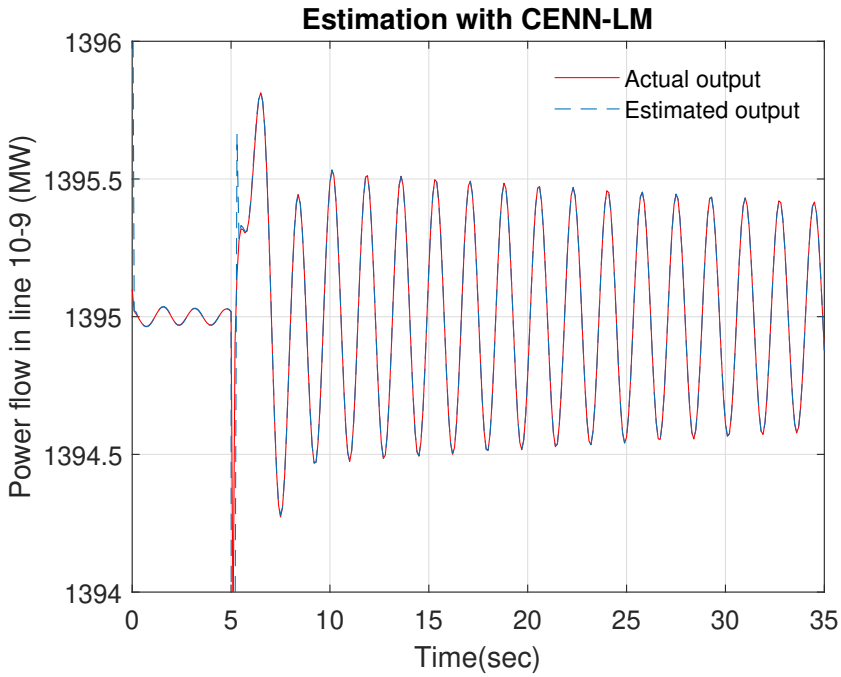

Figure 4.1: Estimation with CENN-LM

Figure 4.1 displays the actual output and estimated output with CENN-LM. Figures. 4.2, 4.3, 4.4 shows the variation of input weights, bias and output weights with CENN-LM respectively.

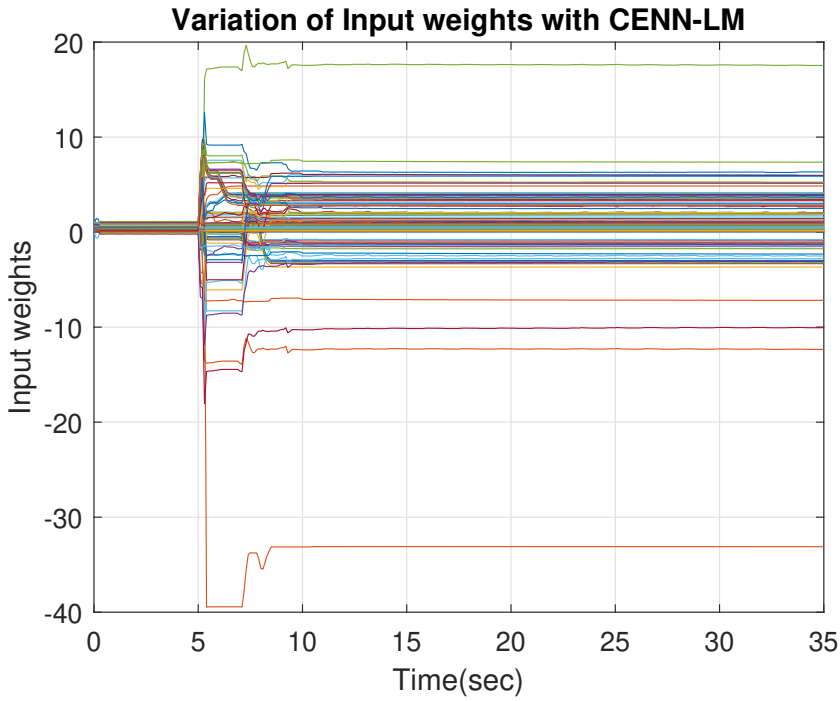

Figure 4.2: Variation of Input weights with CENN-LM

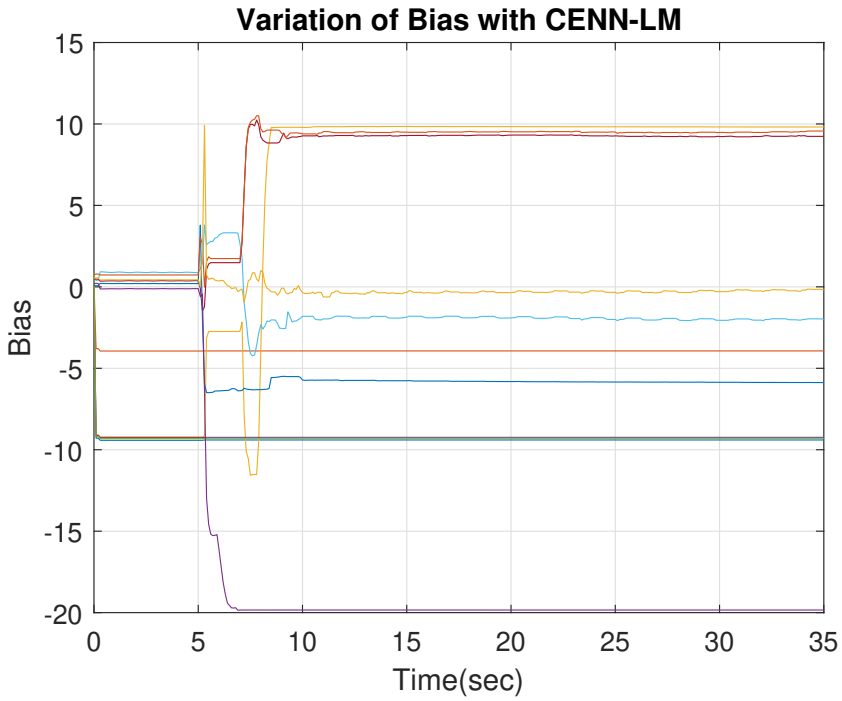

Figure 4.3: Variation of Bias with CENN-LM

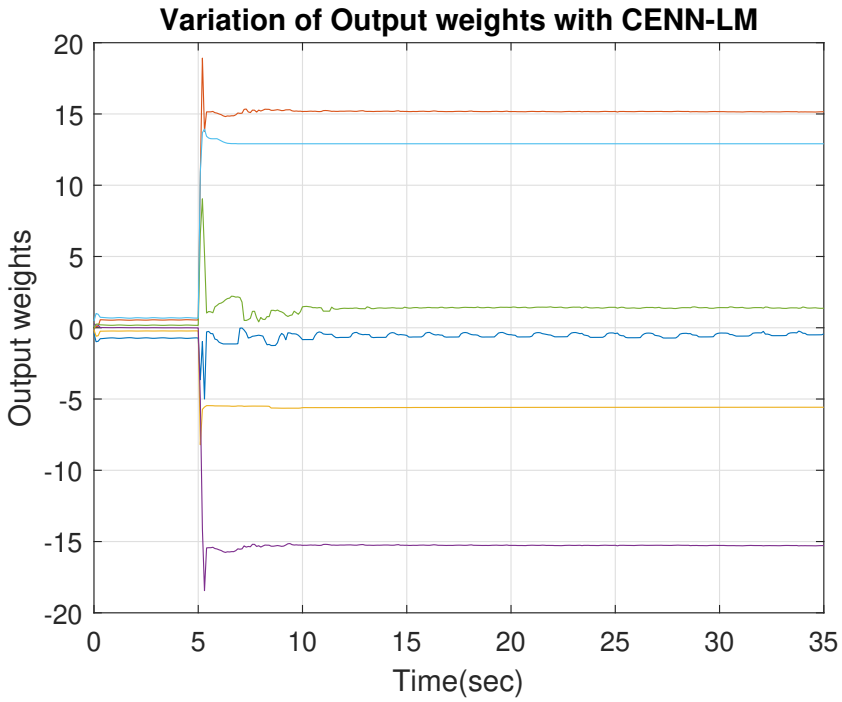

Figure 4.4: Variation of Output weights with CENN-LM

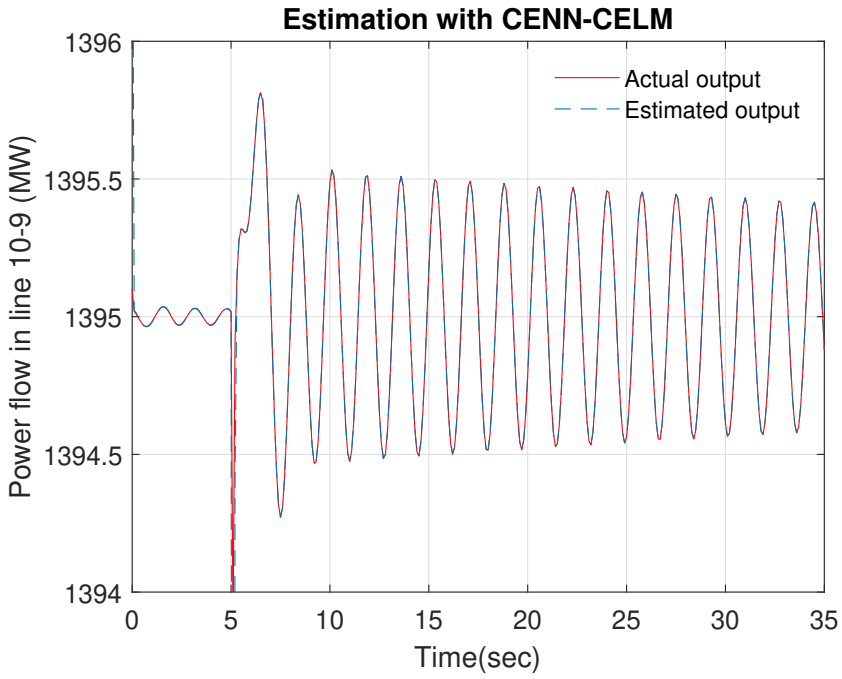

Figure 4.5: Estimation with CENN-CELM

Figure 4.5 displays the actual output and estimated output with CENN-CELM. Figures. 4.6, 4.7, 4.8 shows the variation of input weights, bias and output weights with CENN-CELM respectively.

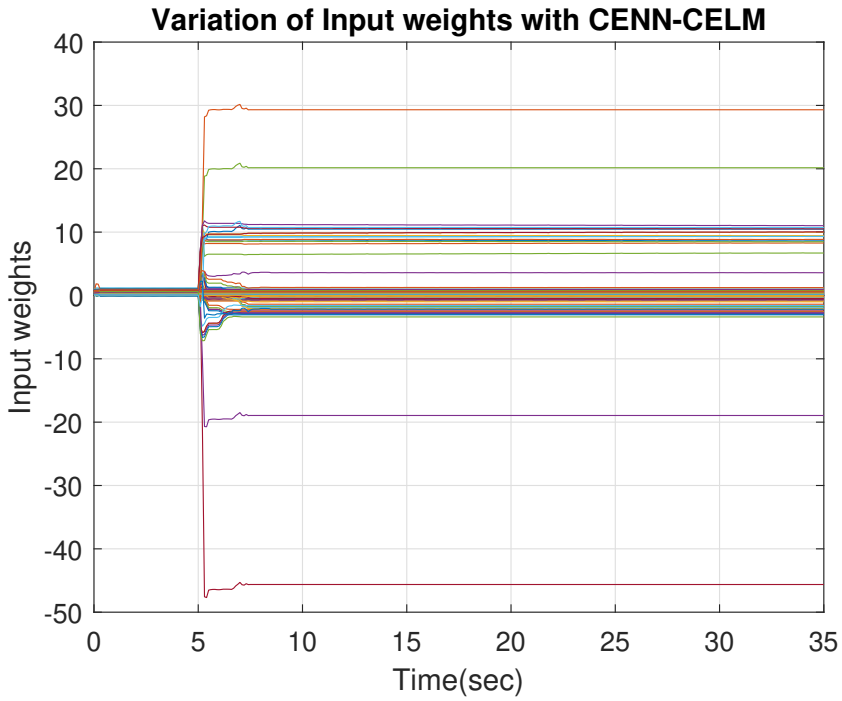

Figure 4.6: Variation of Input weights with CENN-CELM

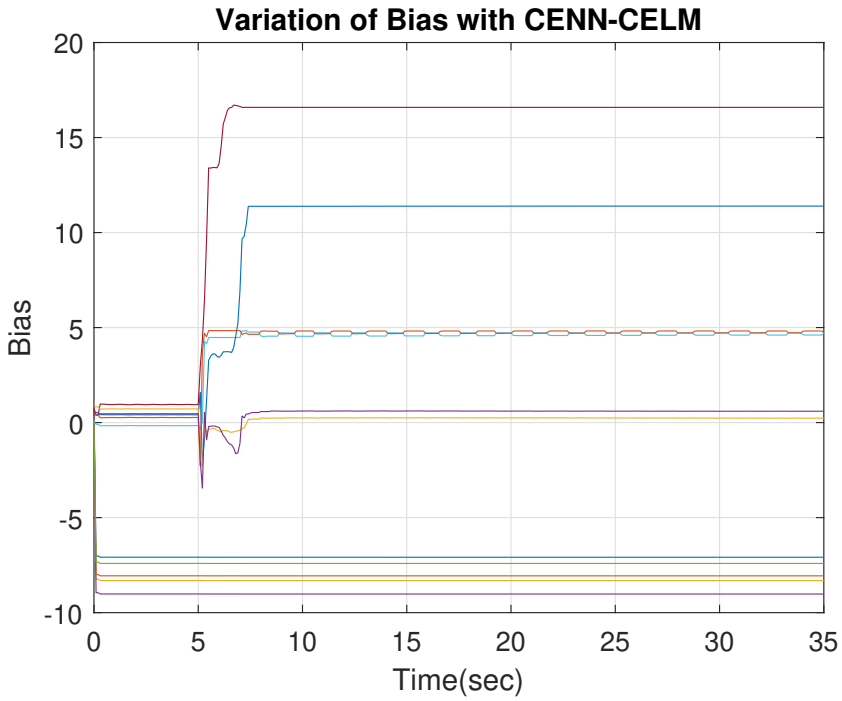

Figure 4.7: Variation of Bias with CENN-CELM

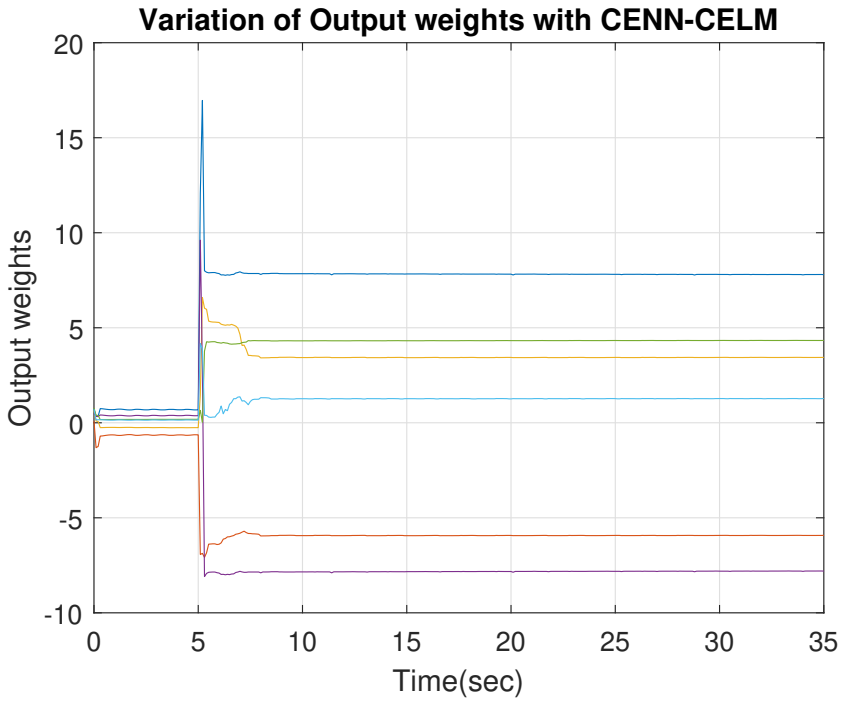

Figure 4.8: Variation of Output weights with CENN-CELM

From the comparison of figures 4.2 and 4.6, 4.3 and 4.7, 4.4, and 4.8, it is clearly shown that the weight parameters in case of CENN-CELM converges faster than CENN-LM.

Figure 4.9 presents the comparison of prediction error between CENN-LM and CENN-CELM. It clearly demonstrates that better estimation is carried out with CENN-CELM as compared to CENN-LM.

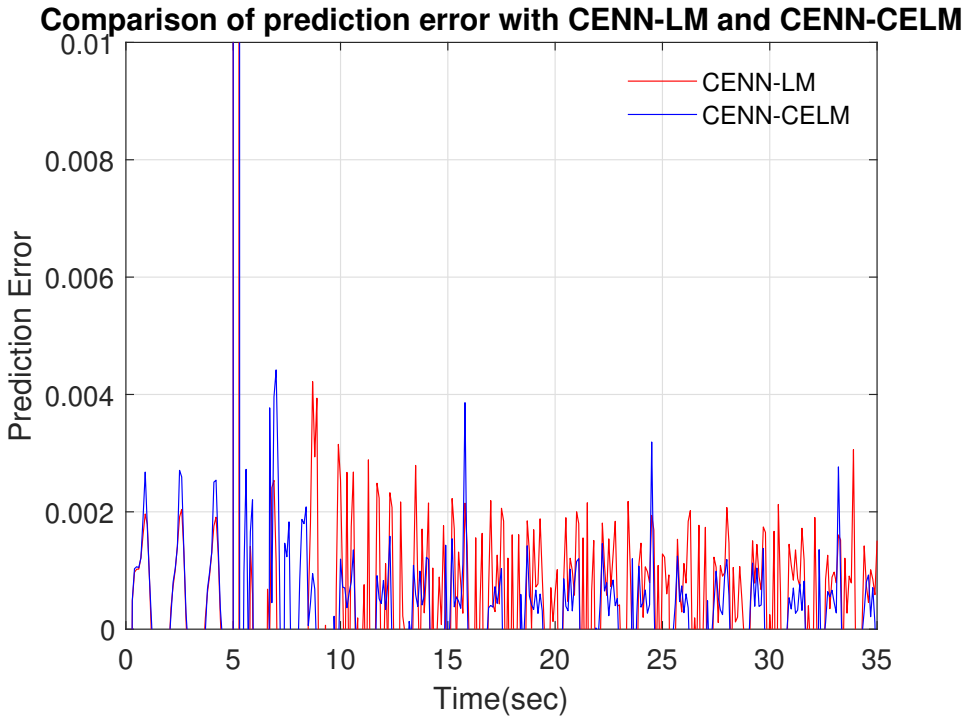

Figure 4.9: Comparison of prediction error between CENN-LM and CENN-CELM

The computation time for estimating weight parameters in CENN-LM is 52.914736 seconds and the computation time for estimating weight parameters in CENN-CELM is 21.237873 seconds. This is obvious because, in CENN-CELM, once the weight parameters gets convergence, they will not compute again and again at each epoch but in CENN-LM, weight parameters will continue to compute again and again irrespective of some of the weights gets convergence. Hence, this comparison shows that CENN-CELM is computationally faster than CENN-LM.

4.2 Evaluation - II: Damping performace of FBLC with LM and CELM algorithm

FBLC is based on the non-linear CENN structure, and the parameters are estimated at each sampling instant using the classical LM algorithm and its modified version i.e. CELM algorithm. The performance of FBLC is demonstrated for the power system

described in section 1.5. A fault is created near bus 8 at 5s that results in outage of one of the lines connecting to buses 7-8 and It results in power oscillations. The parameters of the CENN model, using the classical LM and CELM algorithm, has captured the power sytems's oscillatory behvaiour. The numerical values of the parameters used in the Damping of power oscialltions with LM-FBLC and CELM-FBLC is shown in table 4.2.

Parameter	Value	Equation
m	6	3.4.1
n	6	3.4.1
h	11	3.4.1
w_{size}	5	3.4.2
τ_l	-0.8±3.61ι, -10, -5, -8	3.4.3
K_c	0.9	3.4.5

Table 4.2: Parameters used in evaluation - II

Figure 4.10 shows the damping performance with classical Levenberg-Marquardt (LM) algorithm in conjunction with feedback linearizable controller (FBLC).

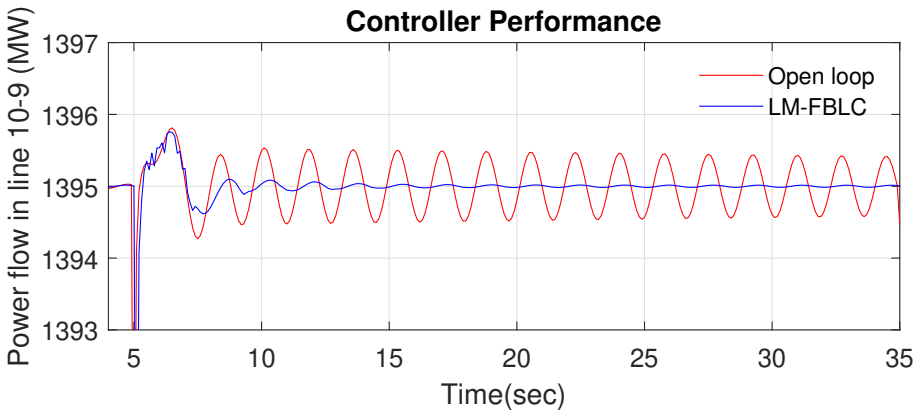

Figure 4.10: Controller performance with LM-FBLC

Figure 4.11 shows the damping performance with computationally efficient Levenberg-Marquardt (CELM) algorithm in conjunction with feedback linearizable controller (FBLC).

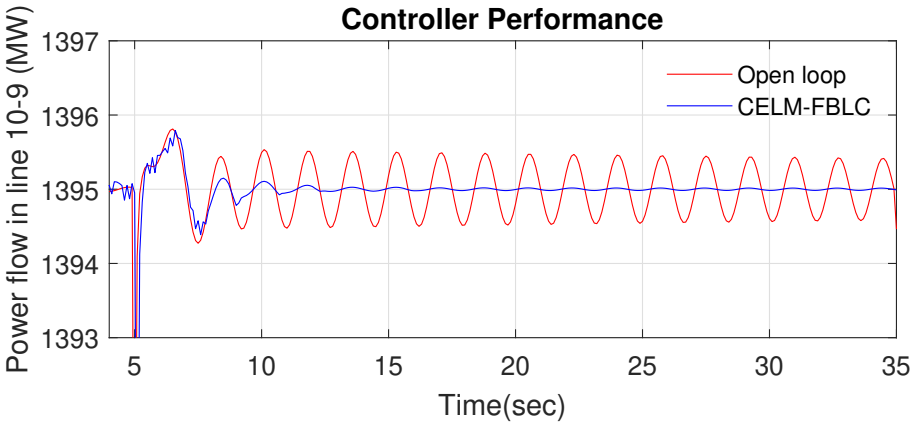

Figure 4.11: Controller performance with CELM-FBLC

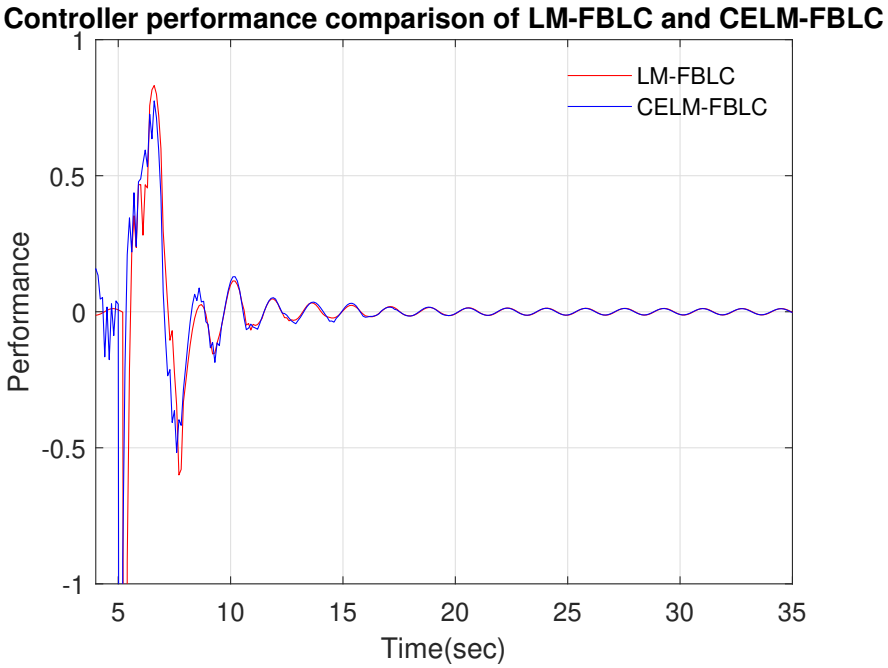

Figure 4.12: Comparison of controller performance between LM-FBLC and CELM-FBLC

Figure 4.12 presents the comparison of damping performance between LM-FBLC and CELM-FBLC. FBLC in conjunction with classical LM algorithm as well as with CELM algorithm, successfully dampens the power oscillation within 10-15s. The comparison plot clearly shows that FBLC performs almost same for both, classical LM algorithm and CELM algorithm.

Chapter 5

Conclusions

A powerful online batch training method for neural networks known as the Levenberg-Marquardt (LM) algorithm is transformed for computationally efficient online estimation of power system's dynamic behaviour, referred as Computationally Efficient Levenberg-Marquardt (CELM) algorithm. A unique form of neural network, referred as Computationally Efficient Neural Network (CENN), is proposed that is compatible with the Feedback Linearizable Controller (FBLC), is used to allow non-linear self-tuning control. It has been demonstrated that using the modified version of LM algorithm i.e. CELM algorithm for successive disturbance leads to improved accuracy, faster convergence and offer less computational time than classical LM algorithm. The efficiency of CELM algorithm for non-linear estimation is shown via a case study on a 4-machine 2-area power system. Feedback linearization control framework (FBLC) has been implemented for damping power oscillation, with classical LM algorithm and it's modified version i.e. CELM algorithm. Moreover, the comparison between LM-FBLC and CELM-FBLC, showed approximately the same damping performace.

5.1 Future Works

In this research, a non-linear activation function i.e. log-Sigmoid is employed in non-linear computationally efficient neural network (CELM) structure to get better performance in terms of convergence, speed and accuracy. however, more efficient results can be achieved using rectified linear unit (ReLU) as an activation function in the neural network structure as ReLU activation function solves the vanishing gradiant problem,

allowing models to learn faster and might perform better. This thesis is based on the damping of oscillation using the non-linear computationally efficient neural netwok structure in conjunction with typical Levenberg-Marquardt (LM) algorithm and it's modified version, computationally efficient Levenberg-Marquardt (CELM) algorithm through feedback linearization control (FBLC) approach. This exercise was carried out for single-input, single-output (SISO) structure on a 4-machine 2-area power system. This exercise can be carried out with multi-inputs, single-output (MISO) and multi-inputs, multi-outputs (MIMO) structures in future and this may yield better outcomes.

Feedback linearization control (FBLC) is based on choosing fixed values of τ_l. Due to this, FBLC works well with a certain fault or operating condition but it will not yield better response in case of varied operating conditions or faults. This issue may be resolved by designing a non-linear controller, which rules out the τ_l factor and this non-linear controller may work effectively in case of various operating scenarios or faults by automatically capturing the behaviour of the system in case of contingencies.

References

[1] J. Arif, S. Ray, and B. Chaudhuri, "Mimo feedback linearization control for power systems," International Journal of Electrical Power Energy Systems, vol. 45, no. 1, pp. 87–97, 2013.

[2] M. Klein, G. Rogers, and P. Kundur, "A fundamental study of inter-area oscillations in power systems," Transactions on Power Systems, vol. 6, no. 3, pp. 914–921, 1991.

[3] H. Haes Alhelou, M. E. Hamedani-Golshan, T. C. Njenda, and P. Siano, "A survey on power system blackout and cascading events: Research motivations and challenges," Energies, vol. 12, no. 4, 2019.

[4] X. H. Chao, "System impact studies for DG projects under development in the US," IEEE Power Engineering Society Summer Meeting, vol. 1, no. 2, pp. 772–774, 2001.

[5] P. Kundur, Power systems stability and control. New York: McGraw-Hill, 1989.

[6] B. Pal and B. Chaudhuri, Robust control in power systems. Power electronics and power systems, New York: Springer, 2005.

[7] S.-j. Cheng, Y. Chow, O. Malik, and G. Hope, "An adaptive synchronous machines stablizer.," IEEE Transaction on Power Systems, vol. 1, no. 3, pp. 101–107, 1986.

[8] E. N. Lerch, D. Povh, and L. Xu, "Advanced svc control for damping power system oscillations," IEEE Transactions on Power Systems, vol. 6, no. 2, pp. 524–535, 1991.

[9] L. Angquist, B. Lundin, and J. Samuelsson, "Power oscillation damping using controlled reactive power compensation-a comparison between series and shunt approaches," IEEE Transactions on Power Systems, vol. 8, no. 2, pp. 687–700, 1993.

[10] M. Noroozian and G. Adersson, "Damping of power system oscillations by use of controllable components," IEEE Transactions on Power Delivery, vol. 9, no. 4, pp. 2046–2054, 1994.

[11] E. V. Larsen, J. J. Sanchez-Gasca, and J. H. Chow, "Concepts for design of FACTS controllers to damp power swings," IEEE Transactions on Power Systems, vol. 10, no. 2, pp. 948–956, 1995.

[12] H. F. Wang and F. J. Swift, "A unified model for the analysis of FACTS devices in damping power system oscillations. i. single-machine infinite-bus power systems," IEEE Transactions on Power Delivery, vol. 12, no. 2, pp. 941–946, 1997.

[13] J. H. Chow, J. J. Sanchez-Gasca, R. Haoxing, and W. Shaopeng, "Power system damping controller design-using multiple input signals," IEEE Control Systems Magazine, vol. 20, no. 4, pp. 82–90, 2000.

[14] B. Wu and O. Malik, "Multivariable adaptive control of synchronous machines in a multimachine power system," IEEE Transaction on Power Systems, vol. 21, no. 4, pp. 1722–1781, 2006.

[15] Z. Wu, W. Gao, T. Gao, W. Yan, H. Zhang, S. Yan, and X. Wang, "State-of-the-art review on frequency response of wind power plants in power systems," Journal of Modern Power Systems and Clean Energy, vol. 6, pp. 1–16, 2018.

[16] Y. Zhu, Control and Placement of Battery Energy Storage Systems for Power System Oscillation Damping. PhD thesis, University of Tennessee, 2018.

[17] G. Chen, C. Liu, C. Fan, X. Han, H. Shi, G. Wang, and D. Ai, "Research on damping control index of ultra-low-frequency oscillation in hydro-dominant power systems," Sustainability, vol. 12, no. 18, 2020.

[18] S. Feng, X. Wu, Z. Wang, T. Niu, and Q. Chen, "Damping forced oscillations in power system via interline power flow controller with additional repetitive control," Protection and Control of Modern Power Systems, vol. 6, no. 1, pp. 1–12, 2021.

[19] I. Zenelis, Inter-Area Oscillations in Power Systems with Uncertainties: Data-Driven Wide-Area Damping Control Design and Monitoring Enhancement Using Energy Storage. PhD thesis, McGill University, 2021.

[20] N. Taheri, E. Akbari, N. Askari, V. Ahmad Tajik, H. Orojlo, M. A. Kazemi, and G. Ghasemi, "Supplementary damping controllers design in vsc hvdc systems and wind farms to improve stability and energy conversion in wind turbine using proposed genetic-bat algorithm," Karafan Quarterly Scientific Journal, vol. 19, no. 1, pp. 357–381, 2022.

[21] Y. Xing, E. Kamal, B. Marinescu, and F. Xavier, "Robust H ∞ Dynamic Output-feedback Control of Power Oscillation Damped via VSC-HVDC Transmission Systems," in 2020 Electrical Power and Energy Conference, (Edmonton, Alberta, Canada), Nov. 2020.

[22] Z. Jinlei, Z. Yao, W. Feixiang, Y. Jiachang, and W. Jiaqi, "The unified supplementary damping method based on voltage-soured converter high-voltage direct current transmission through robust control," Frontiers in Energy Research, vol. 10, 2022.

[23] B. C. Pal, Robust Damping Control of Inter-area Oscillations in Power System with Super-conducting Magnetic Energy Storage Devices. PhD thesis, Imperial College London, 1999.

[24] N. Mithulananthan, C. A. Canizares, J. Reeve, and G. J. Rogers, "Comparison of PSS, SVC, and STATCOM controllers for damping power system oscillations," IEEE Transactions on Power Systems, vol. 18, no. 2, pp. 786–792, 2003.

[25] U. Mhaskar and A. Kulkarni, "Power oscillation damping using FACTS devices:modal controllability, observability in local signals, and location of transfer function zeros," IEEE Transaction on Power Systems, vol. 21, no. 1, pp. 285–294, 2006.

[26] S. Ray and G. K. Venayagamoorthy, "Nonlinear modified pi control of multi-module gcscs in a large power system," Conference Record of the IEEE Industry Applications Society, vol. 3, pp. 1345–1351, 2006.

[27] S. Ray, G. K. Venayagamoorthy, B. Chaudhuri, and R. Majumder, "Comparison of adaptive critic-based and classical wide-area controllers for power systems," Transactions on Systems, Man, and Cybernetics, Part B, vol. 38, no. 4, pp. 1002–1007, 2008.

[28] R. Sadikovic, P. Korba, and G. Andersson, "Self-tuning controller for damping of power system oscillations with FACTS devices," IEEE Power Engineering Society General Meeting, Montreal, 2006.

[29] P. Korba, M. Larsson, B. Chaudhuri, B. Pal, R. Majumder, R. Sadikovic, and G. Andersson, "Towards real-time implementation of adaptive damping controllers for FACTS devices," IEEE Power Engineering Society General Meeting, vol. 19, no. 1, pp. 499–506, 2004.

[30] N. G. Hingorani, "High power electronics and flexible AC transmission system," IEEE Power Engineering Review, vol. 8, no. 7, pp. 3–4, 1988.

[31] N. G. Hingorani, "FACTS technology and opportunities," IEE Colloquium on (Digest No.1994/005), pp. 4/1 – 410, 1994.

[32] A. Edris, "FACTS technology development: an update," IEEE, Power Engineering Review, vol. 20, no. 3, pp. 4–9, 2000.

[33] G. Reed, J. Paserba, and P. Salvantis, "The FACTS on resolving transmission gridlock," IEEE Power and Energy Magazine, vol. 1, no. 5, pp. 41–46, 2003.

[34] G. E. Boukarim, W. Shaopeng, J. H. Chow, G. N. Taranto, and N. Martins, "A comparison of classical, robust, and decentralized control designs for multiple power system stabilizers," IEEE Transactions on Power Systems, vol. 15, no. 4, pp. 1287–1292, 2000.

[35] K. Okada, H. Asano, R. Yokoyama, and T. Niimura, "Reliability-based impact analysis of independent power producersfor power system operations under deregulation," IEEE Canadian Conference on Electrical and Computer Engineering, vol. 3, pp. 1325–1330, 1999.

[36] N. Hemati, J. Thorp, and M. Leu, "Robust nonlinear control of brushless dc motors for direct-drive robotic applications," IEEE Transactions on Industrial Electronics, vol. 37, no. 6, pp. 460 – 468, 2002.

[37] L. Lei and W. Hong-Rui, "Robust tracking control of rigid robotic manipulators based on fuzzy neural network compensator," International Conference on Machine Learning and Cybernetics, Hong Kong, vol. 1, pp. 550 – 555, 2007.

[38] B. Chaudhuri, B. C. Pal, A. C. Zolotas, I. M. Jaimoukha, and T. C. Green, "Mixed-sensitivity approach to H-infinity control of power system oscillations employing multiple facts devices," IEEE Transactions on Power Systems, vol. 18, no. 3, pp. 1149–1156, 2003.

[39] W. Liu, G. Venayagamoorthy, and D. Wunsch, "Adaptive neural network based power system stabilizer design," Proceedings of the International Joint Conference on Neural Networks, vol. 4, no. 1, pp. 2970–2975, 2003.

[40] G. P. Liu, Nonlinear Identification and Control: A Neural Netwrok Approach. Advance in Industrial Control, London: Spinger-Verlag, 2001.

[41] D. K. Chatuvedi and O. P. Malik, "Experimental studies of generalized neuron based adaptive power system stablilizer," Soft Computing - A Fusion of Foundations, Methodologies and Applications, vol. 11, no. 2, pp. 149–155, 2006.

[42] G. Rogers, Power System Oscillations. USA: Kluwer Academic, 2000.

[43] P. W. Sauer and M. A. Pai, Power System Dynamics and Stability. Prentice Hall, 1998.

[44] K. K. Anaparthi, Measurement Based Identification and Control of Electromechanical Oscillations in Power Systems. PhD thesis, Imperial College London, 2006.

[45] N. G. Hingorani. and L. Gyugyi., Understanding FACTS: Concepts and Technology of Flexible AC Transmission Systems. New York: IEEE Press, 2000.

[46] M. R. Salimian and M. R. Aghamohammadi, "A three stages decision tree-based intelligent blackout predictor for power systems using brittleness indices," IEEE Transactions on Smart Grid, vol. 9, no. 5, pp. 5123–5131, 2018.

[47] S. Amini, F. Pasqualetti, and H. Mohsenian-Rad, "Dynamic load altering attacks against power system stability: Attack models and protection schemes," IEEE Transactions on Smart Grid, vol. 9, no. 4, pp. 2862–2872, 2018.

[48] R. Billinton and W. Li, Basic Concepts of Power System Reliability Evaluation, pp. 9–31. Boston, MA: Springer US, 1994.

[49] C. Makdisie, B. Haidar, and H. Haes Alhelou, An Optimal Photovoltaic Conversion System for Future Smart Grids, pp. 601–657. 03 2018.

[50] H. Haes Alhelou, An Overview of Wide Area Measurement System and Its Application in Modern Power Systems, pp. 289–307. 02 2019.

[51] F. Mahfoud, B.-D. Guzun, G. C. Lazaroiu, and H. Haes Alhelou, Power Quality of Electrical Power Systems, pp. 265–288. 02 2019.

[52] I. Ahmad, F. Khan, S. Khan, A. Khan, A. W. Tareen, and M. Saeed, "Black-out avoidance through intelligent load shedding in modern electrical power utility network," Journal of Applied and Emerging Sciences, vol. 8, no. 1, pp. 48–57, 2018.

[53] F.A.Shaikh, M. S. Alam, M.S.J.Asghar, and F. Ahmad, "Blackout mitigation of voltage stability constrained transmission corridors through controlled series resistors," Recent Advances in Electrical and Electronic Engineering, vol. 11, no. 1, pp. 4–14, 2018.

[54] J.-J. E. Slotine and W. Li, "On the adaptive control of robot manipulators," The International Journal of Robotics Research, vol. 6, no. 3, pp. 49–59, 1987.

[55] S. Nicosia and P. Tomei, "Self-tuning control of robot manipulators," International Journal of Adaptive Control and Signal Processing, vol. 7, no. 5, pp. 405–416, 1993.

[56] T. Johnson, C. Harvey, and G. Stein, "Self-tuning regulator design for adaptive control of aircraft wing/store flutter," IEEE Transactions on Automatic Control, vol. 27, no. 5, pp. 1014–1023, 1982.

[57] Z. Chekakta, M. Zerikat, and Y. Bouzid, "Self-tuning pid controller design using fuzzy logic for a vtol quadrotor," 12 2017.

[58] A. K. Pal and T. Nestorović, "Artificial intelligence neural network approach to self tuning of a discrete-time pid control system," in 2021 9th International Conference on Systems and Control (ICSC), pp. 146–151, 2021.

[59] H. Zhang, J. Yan, W. Wang, M. Xu, and W. Ma, "Self-tuning predictive control applicable to ship magnetic levitation damping device," IET Collaborative Intelligent Manufacturing, vol. 4, no. 1, pp. 58–70, 2022.

[60] P. E. Wellstead and M. B. Zarrop, Self-Tuning Systems Control and Signal Processing. Great Britian: John Wiley and Sons, 1991.

[61] K. J. Astrom and B. Wittenmark, Adaptive control. New York: Addison-Wesley, 2nd ed., 1995.

[62] Y. Zeng and O. Malik, "Robust adaptive controller design based on pole-shifting technique.," Proceedings of the 32nd IEEE Conference on Decision and Control, vol. 3, pp. 2353–2357, 1993.

[63] J. Arif, N. R. Chaudhuri, S. Ray, and B. Chaudhuri, "Online levenberg-marquardt algorithm for neural network based estimation and control of power systems," in 2009 International Joint Conference on Neural Networks, pp. 199–206, 2009.

[64] D. K. Chaturvedi and O. Malik, "Generalized neuron-based adaptive PSS for multi-machine environment," IEEE Transaction on Power Systems, vol. 20, no. 1, pp. 358–366, 2005.

[65] P. K. Dash, S. Mishra, and G. Panda, "Damping multimodal power system oscillation using a hybrid fuzzy controller for series connected FACTS devices," IEEE Transaction on Power Systems, vol. 15, no. 4, pp. 1360–1366, 2000.

[66] D. Chaturvedi and O. Malik, "Experimental studies of a generalized neuron based adaptive power system stabilizer," Soft Computing - A Fusion of Foundations, Methodologies and Applications, vol. 11, no. 2, pp. 149–155, 2007.

[67] P. Shamsollahi and O.P.Malik, "An adaptive power system stabilizer using on-line trained neural networks," IEEE Transaction on Energy Conversion, vol. 12, no. 4, pp. 382–387, 1997.

[68] S. H. Ardalan and L. J. Faber, "A fast ARMA transversal RLS filter algorithm," Acoustics, Speech and Signal Processing, IEEE Transactions on, vol. 36, no. 3, pp. 349–358, 1988.

[69] L. Patomaki, J. P. Kaipio, and P. A. Karjalainen, "Tracking of nonstationary EEG with the roots of ARMA models," IEEE 17th Annual Conference Engineering in Medicine and Biology Society, vol. 2, pp. 877–878, 1995.

[70] T. W. S. Chow, H.-Z. Tan, and G. Fei, "Third-order cumulant RLS algorithm for nonminimum ARMA systems identification," Signal Processing, vol. 61, no. 1, pp. 23–38, 1997.

[71] K. M. Passino, Biomimicry for Optimization, Contorl and Automation. London, UK: Springer-Verlag London Limited, 2005.

[72] "State-of-the-art in artificial neural network applications: A survey," Heliyon, vol. 4, 2018.

[73] "Analysis of artificial neural network: Architecture, types, and forecasting applications," Journal of Electrical and Computer Engineering, 2022.

[74] W. Zang, "A generalized ADALINE neural network for system identification.," IEEE International Conference on Control and Automation, pp. 2705–2709, 2007.

[75] S. Ray and G. K. Venayagamoorthy, "Real-time implementation of a measurement-based adaptive wide-area control system considering communication delays," IET Generation, Transmission & Distribution, vol. 2, no. 1, pp. 62–70, 2008.

[76] J. Sarangapani, Neural Network Control of Nonlinear Discretem Time Systems. CRC, 2006.

[77] A. Novokhodko and S. Valentine, "A parallel implementation of the batch back-propagation training of neural networks," International Joint Conference on Neural Networks, IJCNN, vol. 3, pp. 1783–1786, 2001.

[78] M. T. Hagan, H. B. Demuth, and M. H. Beale, Neural network design. Boston, London, UK: PWS Pub, 1996.

[79] C.-T. Kim, J.-J. Lee, and H. Kim, "Variable projection method and Levenberg-Marquardt algorithm for neural network training," IEEE Industrial Electronics, IECON - 32nd Annual Conference, pp. 4492–4497, 2006.

[80] Y. T. Kwak, J. won Hwang, and C. J. Yoo, "A new damping strategy of levenberg-marquardt algorithm for multilayer perceptrons," Neural Network World, vol. 21, pp. 327–340, 2011.

[81] "A new modified efficient levenberg–marquardt method for solving systems of non-linear equations," Mathematical Problems in Engineering, 2021.

[82] Z. Yan, S. Zhong, L. Lin, and Z. Cui, "Adaptive levenberg–marquardt algorithm: A new optimization strategy for levenberg–marquardt neural networks," Mathematics, vol. 9, no. 17, 2021.

[83] J. F. V. Bob´al, J. Böhm and J. Mach´a¡cek, Digital Self-tuning Controllers Algorithms, Implementation and Applications. Germany: LE-TEX Jelonek, Leipzig, 2005.

[84] I. D. Landua and M. M'Saad, Adaptive Control. London: Springer, 1998.

[85] G. C. Goodwin and K. S. Sin, Adaptive filtering prediction and control. Englewood Cliffs, New Jersey: Prentice Hall, 1984.

[86] D. Marquardt, "An algorithm for least squares estimation on nonlinear parameters," SIAM J. APPL. MATH, vol. 11, pp. 431–441, 1963.

[87] J.-J. E. Slotine and W. Li, Applied Nonlinear Control. New Jersey: Prentice Hall, 1991.

[88] M. A. Henson and D. E. Seborg, "Input-output linearization of general nonlinear processes," AIChE Journal, vol. 36, no. 11, pp. 1753–1757, 1990.