A MAJOR PROJECT REPORT ON

IMPLEMENTATION OF NARMA-L2 NEUROCONTROLLER FOR SPEED REGULATION OF SERIES CONNECTED DC MOTOR

Submitted in partial fulfillment of the
Requirements for the award of the degree
of

MASTER OF TECHNOLOGY
In
CONTROL & INSTRUMENTATION

By

NAWRATNA KUMAR
Roll No. 05/C&I/2K10

Under the Guidance
of
SUDARSHAN K. VALLURU
Associate Professor
Department of Electrical Engineering

Department of Electrical Engineering
Delhi Technological University
(Formerly Delhi College of Engineering)
Delhi-110042
2010-12

Department of Electrical Engineering

Delhi Technological University

(Formerly Delhi College of Engineering)

CERTIFICATE

This is to certify that Nawratna Kumar, Roll No. 05/C&I/10, student of M.Tech, Control & Instrumentation, Department of Electrical Engineering, Delhi Technological University has submitted the dissertation entitled "IMPLEMENTATION OF NARMA-L2 NEUROCONTROLLER FOR SPEED REGULATION OF SERIES CONNECTED DC MOTOR" under my guidance towards partial fulfillment of the degree of M.Tech, Control & Instrumentation. This dissertation is a bonafied record of project work carried out by him under my guidance and supervision during the academic session 2011-2012.

SUDARSHAN K.VALLURU

Associate Professor
Electrical Engineering Department
Delhi Technological University

ACKNOWLEDGEMENT

I wish to express my profound sense of deepest gratitude to my guide and motivator Sudarshan K. Valluru, Associate Professor, Electrical Engineering Department, for his valuable guidance, co-operation and finally help for providing necessary facilities and sources during the entire period of this project.

I am thankful to the HOD and all the faculty members of Electrical Engineering Department who enlightened me during my studies.

I am grateful to my parents for their moral support all time. They have been always supportive. I am also thankful to my friends for their unconditional support and motivation during this work.

NAWRATNA KUMAR

M.Tech, Control & Instrumentation
Electrical Engineering Department
Roll No. 05/C&I/10

ABSTRACT

Series connected DC Motor Drive (SCDM) has been the most suitable drive for electric rapid transit railway system and electric vehicles, characterized by speed and torque control using PID based choppers. However there are some serious problems underlying such as nonlinear switching, variable nature of load and difficulty in exact mathematical model representation. These may cause poor dynamic response of the system. To improve the performance of the system these nonlinearities are to be properly incorporated in system representation and appropriate model of SCDM to be considered. Based on the nonlinear mathematical model of series connected DC motor system dynamics, the analysis and design of two speed controller presented, which are non linear PID and nonlinear autoregressive-moving average L-2 (NARMA L-2) controllers. The entire system has been modeled using MATLAB 8.0 Simulink and power system block sets. The speed response of SCDM is observed by giving reference inputs as speed and load torque in terms of step variation. The response of the system using NARMA L-2 Neuro Controller and classical PID Controller are compared. It was observed that the NARMA L-2 Neuro Controller performance was better in terms of rising time, overshoot and steady state error over classical PID Controller.

CONTENTS

ITEMS	TITLE	PAGE NO.
1	LIST OF FIGURES	vi
2	LIST OF TABLES	vii
3	GLOSSARY	viii
CHAPTER 1	INTRODUCTION	1
	1.1 Literature Review	2
	1.2 Aims and Objectives	4
	1.3 Outline of Dissertation	5
CHAPTER 2	FUNDAMENTALS OF CHOPPER AND NEURAL NETWORK	6
	2.1 Introduction	7
	2.1.1 Four Quadrant Chopper	8
	2.1.2 IGBT Gate Driver circuit	8
	2.1.3 IGBT Power circuit	9
	2.2 Speed and Current Controller for SCDM	9
	2.2.1 Controller Fundamentals	9
	2.2.2 Deciding factors of Controller	10
	2.2.3 Importance of Current Controller	12
	2.3 Artificial Neural Network	13
	2.3.1 Multilayer Perceptron architecture	14
	2.3.2 Network Architectures	16
	2.3.3 Multiple Layers of Neurons	17
	2.3.4 Training of Multilayer Networks	19
CHAPTER 3	MATHEMATICAL MODELING OF NARMA L-2 NEUROCONTROLLER	20
	3.1 Mathematical Modeling of NARMA-L2 Neuro Controller	21
	3.1.1 NARMA-L2 (Feedback Linearization) Control	22
	3.1.2 Identification of the NARMA-L2 Model	23
	3.1.3 NARMA-L2 Controller	25
CHAPTER 4	MODELING AND SIMULATION STUDIES OF SCDM SYSTEM	26
	4.1 Dynamics of SCDM	27

4.2 Derived model of SCDM from Simulink 28
4.2.1. Simulation model 28
4.2.2. Function Block with its Parameters 29
4.3 Nonlinear Speed Controllers 30
4.4 Speed control of SCDM using 4-Quadrant chopper 33
4.4.1 Simulink model 34
4.5 Speed control of SCDM using NARMA-L2 Controller 35
4.5.1 NARMA-L2 Controller 35
4.5.2 System Identification and controller design Stage 36
4.5.3 Implementation of NARMA L-2 Neuro controller 41
4.5.4 Simulink model of NARMA L-2 Controlled SCDM 42
CHAPTER 5 RESULTS AND DISCUSSION 43
5.1 SCDM system parameters 44
5.2 Speed responses of SCDM system 44
5.2.1 PID controller 44
5.2.2 NARMA-L2 controller 46
5.3 Comparison of Speed responses of SCDM by PID and NARMA L-2 controller 46
CONCLUSION 50
FUTURE SCOPE 50
REFERENCES 50

LIST OF FIGURES

Page No.

Figure 2.1 Four Quadrant chopper configuration 9

Figure 2.2 Speed control of SCDM using 4-Quadrant chopper 12

Figure 2.3 Neural Networks as Function Approximator 13

Figure 2.4 Single-Input Neuron 14

Figure 2.5 Log-Sigmoid Transfer Function 15

Figure 2.6 Multiple-Input Neuron 15

Figure 2.7 Representation of the neuron in matrix form 16

Figure 2.8 Layer of *S* Neurons 17

Figure 2.9 Layer of S Neurons, Matrix Notation 17

Figure 2.10 Three-Layer Networks 18

Figure 3.1 NARMA-L2 controlled SCDM 22

Figure 3.2 Approximated NARMA-L2 Plant Model 24

Figure 3.3 Block diagram of NARMA-L2 Controller 25

Figure 3.4 Implementation of NARMA-L2 Controller 25

Figure 4.1Equivalent circuit of SCDM drive system 27

Figure 4.2 SCDM for conventional PID controller 28

Figure 4.3 SCDM for NARMA-L2 controller 29

Figure 4.4 Subsystem for the chopper controlled SCDM 29

Figure 4.5Subsystem for NARMA-L2 controlled SCDM 29

Figure 4.6 Parameterized linear closed- loop system 32

Figure 4.7 Simulink model for Speed control of DCSM using 4-Quadrant chopper 34

Figure 4.8 Simulink model of Four Quadrant chopper model 34

Figure 4.9 Subsystem of Four Quadrant chopper model 35

Figure 4.10 Simulink model for PID Controller 35

Figure 4.11 MATLAB based Subsystem for NARMA L-2 Controller 35

Figure 4.12 Neural network training 36

Figure 4.13 Specifications of the plant model 37

Figure 4.14 SCDM input-output data 37

Figure 4.15 Performance of NARMA L-2 controller 38

Figure 4.16 Training state of NARMA L-2 controller 38

Figure 4.17 Regression of NARMA L-2 controller 39

Figure 4.18 Testing data of NARMA L-2 controller 39
Figure 4.19 Training data of NARMA L-2 controller 40
Figure 4.20 Validation data of NARMA L-2 controller 40
Figure 4.21 NARMA-L2 controlled SCDM 42
Figure 5.1 Speed response of SCDM 45
Figure 5.2 Speed response of SCDM with modified PID gains 45
Figure 5.3 Armature current 45
Figure 5.4 Load Torque with PID controller 45
Figure 5.5 Speed response of SCDM with NARMA L-2 controller 46
Figure 5.6 Load Torque with NARMA L-2 controller 46
Figure 5.7 Comparison of speed responses for SCDM with 200V 47
Figure 5.8 Comparison of speed responses for SCDM with 40V 48
Figure 5.9 Desired Speed responses of both the controller for SCDM 49

LIST OF TABLES

Table 2.1 Comparison of various processes and its controller structure 11
Table 2.2 Comparison of various control modes of controller 11
Table 5.1 Series connected DC Motor: list of parameters 44
Table 5.2 gains for PID Controller 44
Table 5.3 gains for PID Controller with modification 45
Table 5.4 Parametric values of both the controllers with 200V 47
Table 5.5 Parametric values of both the controllers with 40V 48
Table 5.6 Desired parametric values of both the controllers with 200V 49

GLOSSARY

SYMBOL	DESCRIPTION
R_a , R_f	Resistance of the armature, and field circuit
L_a ,L_f	Inductance of the armature, and field
L, R	Total armature and field circuit inductance, and resistance
A1-A2	Rotating armature windings
i ,i(U)	armature current or Field current, current of motor at equilibrium
ω, ω(U)	Rotational speed of the motor, rotational speed of motor at equilibrium
k_1(U),k_2(U),k_D	Proportional, integral and derivative gain
J	Moment of inertia associated with both motor and the load
i_a , i_f	Armature and field current
τ_e , τ_L	Electromagnetic Torque, Load Torque
p ,a	Scalar input and output of neural network
W, w	Weight in neural network
b ,n ,f	Bias, Summer output (net input), and transfer function
N ,G, g	Nonlinear function, Function to minimize mean square error
V, U	Terminal control voltage, Voltage at equilibrium
pq , tq	Input to the network and corresponding target output
u(k), y(k)	System input, and output
SCDM	Series Connected DC Motor
IGBT	Insulated Gate Bipolar Transistors
NNC	Neural Network Controllers
NARMA L-2	Nonlinear Auto Regressive Moving Average Level -2
TDL	Time delay line

CHAPTER 1

INTRODUCTION

1.1 Literature Review

Direct current (DC) motors have been widely used in many industrial applications such as electric vehicles, steel rolling mills, electric cranes, and robotic manipulators due to precise, wide, simple, and continuous control characteristics. Traditionally rheostatic armature control method was widely used for the speed control of low power dc motors. However the controllability, cheapness, higher efficiency, and higher current carrying capabilities of static power converters brought a major change in the performance of electrical drives. The desired torque-speed characteristics could be achieved by the use of conventional proportional integral-derivative (PID) controllers. As PID controllers require exact mathematical modeling, the performance of the system is questionable if there is parameter variation. During recent past neural network controllers (NNC) were effectively introduced to improve the performance of nonlinear systems. The application of NNC [1]-[2] is very promising in system identification and control due to learning ability, massive parallelism, fast adaptation, inherent approximation capability, and high degree of tolerance.

A DC Motor in which the field circuit is connected in series with the armature circuit is referred to as a Series Connected DC Motor (SCDM). Due to this electrical connection, the torque produced by this motor is proportional to the square of the current (below field saturation), resulting in a motor that produces more torque per ampere of current than any other dc motor. Control of a SCDM system suffers from the considerable nonlinearities including the square of current and the product of current and speed. Traditional way to control such a nonlinear system is to use linear controllers, which are based on linearized system model about a nominal operating point. These controllers with constant gains can be expected to perform satisfactorily in the vicinity of the operating point. However, they may not be capable of dealing

with a situation where the motor system operates over wide dynamic regimes. To overcome the drawbacks of the traditional linear controllers, one approach is to employ extended linearization design [3]-[5]. This approach is to design a number of linear controllers corresponding to many operating points that cover the whole dynamic region of the motor system. Then these linear controllers are pieced together to obtain a nonlinear controller, which is known to demonstrate good property of gain scheduling with respect to constant operating equilibrium points.

Choppers are used for the control of DC motors because of the numbers of advantages such as high efficiency, flexibility in control, light weight, small sizes, quick response, and regeneration down to very low speeds. The control of the armature currents is done with the help of 4-Quadrant Chopper [6]. The switching sequence of chopper is in pairs of IGBT1, IGBT4 and IGBT2, IGBT3. Due to the selection of IGBT as a switching device the motor is capable to rotate in both the directions, thus there is forward and reverse regenerative breaking takes place. Since uncertainties and disturbances are unavoidable in practice, changing load torques must be taken into account when designing a controller for a SCDM. A number of advanced robust control methods have been ex-ploited to handle uncertainties and disturbances in motor systems. For speed tracking control of SCDM by nonlinear **c**ontrol approaches, such as back stepping [7] and feedback linearization [8] methods, usually give asymptotic tracking results with constant decay rates. If we want a fast transient response, large control "gains" must be applied but in many cases which is impossible due to high cost. The significance of the problem is to get good transient response without increasing "control gains" [9]. The Integral Linear quadratic speed control, current control with dominant pole compensation and pole placement techniques [10] are still exhibits partial performance due to parametric variation.

The classical PID controller cannot perform effectively since it is developed based on linear system theory, but the AI based controllers like NARMA model can be used to identify and control nonlinear dynamic systems since they can approximate a wide range of nonlinear functions to any desired degree of accuracy. The NARMA model is an exact representation of input-output behaviour for finite dimensional nonlinear discrete dynamic systems. To solve nonlinearity problems in advanced control of dynamic systems two versions of NARMA models are proposed [11]-[13] NARMA L-1 and NARMA L-2. The later is more convenient to be implement practically using multilayer ANN due to the advantages such as the training of NARMA L-2 controller is straight forward because the controller is simply rearrangement of neural network plant model which is trained offline, batch form and there is no separate dynamic training for the controller. Another advantage is only online computation is forward passed through the neural network controller. The work presented in this thesis is concentrating on demonstration, operation and simulation of NARMA L-2 Neuro controller for speed regulation of SCDM.

1.2 Aims and Objectives

The Aims and objectives of this project is to speed regulation of SCDM.

I. To control the speed of SCDM by PID Controller based on 4-Quadrant chopper are simulated by using MATLAB 8.0 Simulink and power system block sets.
II. To control the speed of SCDM by NARMA-L2 controller are simulated by using MATLAB 8.0 Simulink and neural network training block sets.

1.3 Outline of Dissertation

The dissertation consists of five chapters. A brief overview of the contents for each chapter is given below:

Chapter 1 Introduction, literature review, aims and objectives of the project.

Chapter 2 Discusses the Four quadrant chopper and Artificial Neural Network. This chapter also covers the selection of the appropriate power system block sets, Neural Network Controller and its requirements.

Chapter 3 This chapter covers the concepts of speed and current controllers and the implementation of nonlinear autoregressive-moving average (NARMA) L-2 controller.

Chapter 4 Discusses the system as a whole, including an overall description, modeling and simulation of Series Connected DC Motor and finally speed control of SCDM using conventional PID and NARMA L-2 Controllers is shown.

Chapter 5 This final chapter gives Results, simulation studies and conclusion with respect to the overall project.

CHAPTER 2

FUNDAMENTALS
OF
CHOPPER AND NEURAL NETWORK

2.1 Introduction

Electric motor controls have only been made possible by the development of high power solid-state devices. The most common power devices that have been used in motor drives are MOSFET, IGBT, BJT, and SCR. Early designs used BJTs and SCRs, but advancements in semiconductor technology have made these devices obsolete and replaced with MOSFETs and IGBTs in modern drives. SCRs are no longer normally chosen for motor drive systems as they cannot be easily turned off after they have been gated on. This makes it difficult to use a PWM signal to drive the motor. Power BJTs are also an uncommon choice for high power drives as this type of transistor is a current gain device. This means that to put the transistor into saturation, sufficient base current must be available. A base current requirement of one-tenth of the collector current is not uncommon, so this results in a design that dissipates a large amount of power in the gate drive circuit. However, BJTs are still commonly used for low power applications due to their low cost. IGBTs offer performance similar to BJTs but are driven like a MOSFET.

Electric motors are frequently used as the final control element in positional or speed control system. As there are two types such as D.C & A.C motors. The control of the D.C motor speed by the chopper is required where the supply is d.c. or a.c. that has already rectified by the d,c. voltage. The most important applications of the chopper are in the speed control of the d.c motor used in the industrial applications or traction drives. Choppers are used for the control of d.c motors because of the numbers of advantages such as high efficiency, flexibility in control, light weight, small sizes, quick response, and regeneration down to very low speeds.

The IGBT advantage are very high input impedance which is voltage controlled device , low level of loss in conduction state, low switching loss , high operating frequency(up to 50 KHz), simple protection circuits. They have wide area of applications like, used in Traction

Drives for railways, buses & electrically driven vehicles, also in steelworks, hot strip mills, transformer winding machines, in position controls etc.

2.1.1 Four Quadrant Chopper

The four quadrant chopper has certain advantages such as high operational frequency, smooth and linear control, high efficiency and fast response. The block diagram of the system is shown in figure 2.1. The converter is full bridge chopper with four quadrant operation. Because of the four quadrants there is rotation of the D.C motor in the both direction i.e. reverse and forward direction, this has been done with the help of direction changing logic circuit.

2.1.2 IGBT Gate Driver circuit

IGBT is a voltage controlled device and has high input capacitance of 3000 to 7000 PF between its gate and source terminal. The on state voltage across IGBT depends on gate to source voltages V_{gs} therefore to keep the on state voltage low relatively high positive gate to source voltage must be applied .However the voltage should not exceed breakdown voltage of the gate. V_{gs} should around 15V.During the off state a negative V_{gs} should be applied. It is about 2 to 5V.The V_{gs} must be applied continuously or else IGBT will be turn off. The output current of the driver circuit should be sufficient to charge and discharge the gate to source capacitance as quickly as possible. This will help in reducing T_{on} and T_{off} for IGBT. It will also reduce switching losses. The IGBT and control circuit must be electrically isolated. The wiring to the drive circuit to IGBT must be short as possible to avoid oscillations at the gate. The wires must be twisted to eliminate the effect of EMI. The drain current of IGBT must be sensed by sensing circuit. As soon as the drain current exceeds the saturation value the gate drive to the IGBT must be turned off. The gate driver circuits consist of opto-isolator along with Darlington pair which is

used for the driving of the IGBT. The function of opto-isolator is to provide the isolation between the control circuitry and to control the short circuit.

2.1.3 IGBT Power circuit

IGBT power circuit consists of four IGBT, these are turned ON and OFF in pairs of IGBT1 & IGBT4, IGBT2 & IGBT4 as shown in figure2.1. The IGBT used is IRG4B30FD which have in built snubber circuit with ultra fast soft recovery diode has following specifications: operating frequency 1-5 KHz in hard switching, >20 KHz in resonant mode, Vces =600v, Ic @ 25ºC = 31 A, Vge= ±20V.

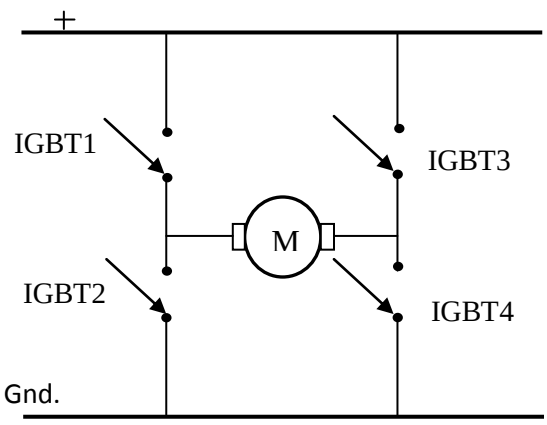


Figure 2.1. Four Quadrant chopper configuration

2.2 Speed and Current Controller for SCDM

2.2.1 Controller Fundamentals

The controller used in a closed loop provides a very easy and common technique of keeping motor speed at any desired set-point speed under changing load conditions [14]-[15]. This controller can also be used to keep the speed at the set-point value when, the set-point is stepping up or down at a defined rate. The essential addition required for this condition to the previous system is a means for the present speed to be measured. In this closed loop speed controller, a voltage signal obtained from a tacho attached to the rotor which is proportional to the motor speed is fed back to the input where signal is subtracted from the set-point speed to produce an

error signal. This error signal is then fed to controller and the controller output will be to make the motor run at the desired set-point speed. For example, if the error speed is negative, this means the motor is running slow so that the controller output should be increased and vice-versa.

2.2.2 Deciding factors of Controller

The control action can be imagined at first sight as something simple like if the error speed is negative, then multiply it by some scale factor generally known as gain and set the output drive to the desired level. But this approach is only partially successful due to the following reason: if the motor is at the set-point speed under no load there is no error speed so the motor free runs. If a load is applied, the motor slows down and a positive error speed is observed. Then the output increases by a proportional amount to try and restore the desired speed. However, when the motor speed recovers, the error reduces drastically and so does the drive level. The result is that the motor speed will stabilize at a speed below the set-point speed at which the load is balanced by the product of error speed and the gain. This basic technique discussed above is known as "proportional control" and it has limited use as it can never force the motor to run exactly at the set-point speed.

From the above discussion an improvement is required for the correction to the output which will keep on adding or subtracting a small amount to the output until the motor reaches the set-point. This effect can be done by keeping a running total of the error speed observed for instant at regular interval and multiplying this by another gain before adding the result to the proportional correction found earlier. This approach is basically based on what is effectively the integration of the error in speed. Till now we have two mechanisms working simultaneously trying to correct the motor speed which constitutes a PI (proportional-integral) controller. The

proportional term does the job of fast-acting correction which will produce a change in the output as quickly as the error arises. The integral action takes a finite time to act but has the capability to make the steady-state speed error zero. A further refinement uses the rate of change of error speed to apply an additional correction to the output drive. This is known as Derivative approach. It can be used to give a very fast response to sudden changes in motor speed. The table 2.1.depicts the comparison of various process and its controller structure and Table 2.2 depicts the comparison of various control modes of controller.

Table 2.1. Comparison of various processes and its controller structure

Process	Controller structure			
	P	PD	PI	PID
Pure dead time	unsuitable	unsuitable	Very suitable or pure I controller	
First- order with short dead time	Suitable if derivative is acceptable	Suitable if derivative is acceptable	Highly suitable	Highly suitable
Second-order with short dead time	Deviation mostly too high	Deviation mostly too high	Not as good as PID	Highly suitable
Higher-order	**unsuitable**	**unsuitable**	**Not as good as PID**	**Highly suitable**
Without self-limitation with delay	suitable	suitable	suitable	suitable

Table 2.2. Comparison of various control modes of controller

Controller Mode	Settling Time	Offset	Overshoot
P	Lower than highest	highest	Lower than highest
PI	Highest	Zero	Highest
PD	Lower	Lower than highest	Lower
PID	**More than lowest**	**Zero**	**More than lowest**

Hence, a suitable combination of the three basic modes- proportional, integral and derivative (PID) can improve all aspects of the system performance.

2.2.3 Importance of Current Controller

When the machine is made to run from zero speed to a high speed then motor has to go to specified speed. But due to electromechanical time constant motor will take some time to speed up. But the speed controller used for controlling speed acts very fast [15]. Speed feedback is zero initially, so this will result in full controller output and hence converter will give maximum voltage. So a very large current flow at starting time because counter emf is zero at that time which sometime exceeds the motor maximum current limit and can damage the motor windings. Hence there is a need to control current in motor armature. To solve the above problem we can employ a current controller which will take care of motor rated current limit. The applied voltage will now not dependent on the speed error only but also on the current error. We should ensure that Voltage is applied in such a way that machine during positive and negative torque, does not draw more than the rated current. So, an inner current loop hence current controller is required. The schematic of Speed control of SCDM using 4-quadrant chopper as shown in figure 2.2

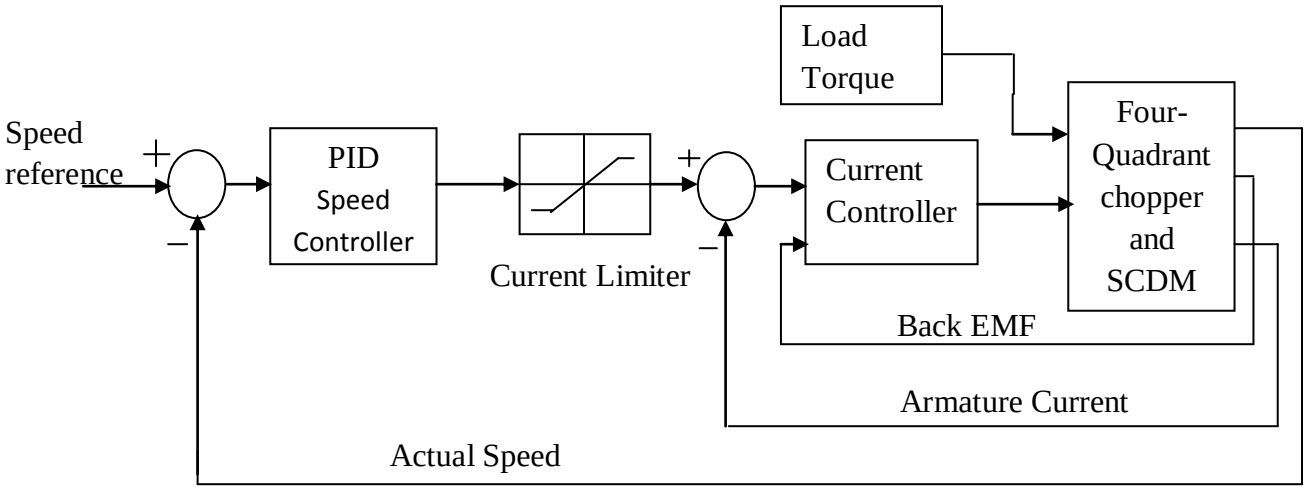


Figure 2.2 Speed control of SCDM using 4-Quadrant chopper

In the above model the current reference is obtained by amplification of the angular rotational speed error. This reference is limited to the maximum value tolerable by the motor.

2.3 Artificial Neural Network

The field of neural networks covers a very broad area. Instead, we will concentrate on the most common neural network architecture – the multilayer perceptron. We will describe the basics of this architecture, discuss its capabilities and show how it has been used in several different control system configurations. As shown in Figure2.3, we have some unknown function that we wish to approximate. We want to adjust the parameters of the network so that it will produce the same response as the unknown function, if the same input is applied to both systems [16].For our applications, the unknown function may correspond to a system we are trying to control, in which case the neural network will be the identified plant model. The unknown function could also represent the inverse of a system we are trying to control, in which case the neural network can be used to implement the controller. In this work we will present NARMA L-2 control architecture demonstrating a function approximator neural network.

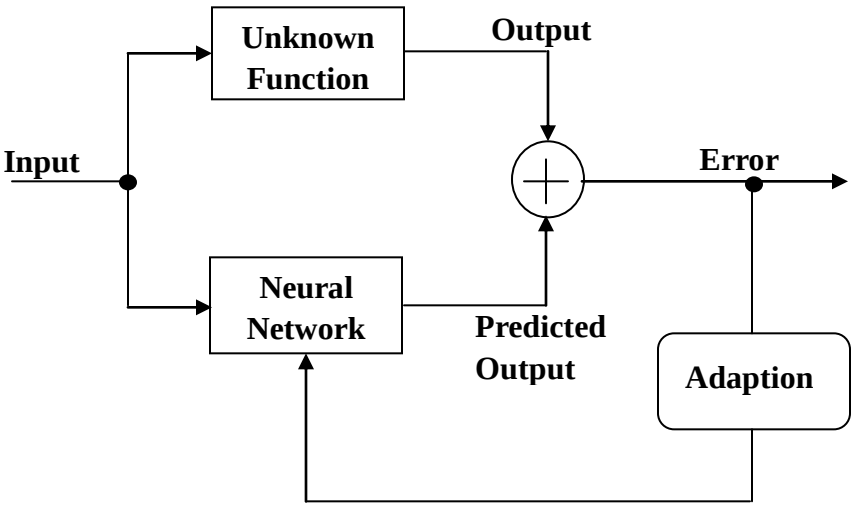


Figure 2.3 Neural Networks as Function Approximator

In the next section we will present the multilayer perceptron neural network, and will demonstrate how it can be used as a function approximator.

2.3.1 Multilayer Perceptron architecture

The multilayer perceptron neural network is built up of simple components. We will begin with a single-input neuron, which we will then extend to multiple inputs [17]. We will next stack these neurons together to produce layers. Finally, we will cascade the layers together to form the network.A single-input neuron is shown in Figure 2.4. The scalar input p is multiplied by the scalar weight w to form wp, one of the terms that is sent to the summer. The other input, 1, is multiplied by a bias b and then passed to the summer. The summer output n, often referred to as the net input, goes into a transfer function f, which produces the scalar neuron output a.

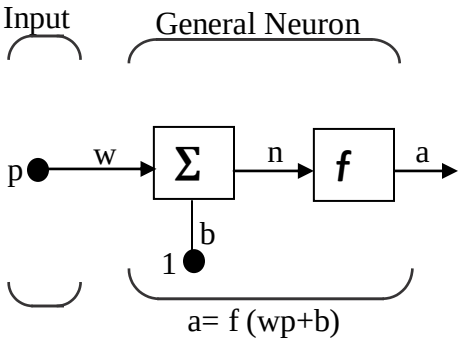


Figure 2.4 Single-Input Neuron

The neuron output is calculated as

$$a = f(wp + b) \quad \text{..... (2.1)}$$

Note that and are both adjustable scalar parameters of the neuron. Typically the transfer function is chosen by the designer, and then the parameters and is adjusted by some learning rule so that the neuron input/output relationship meets some specific goal. The transfer function in Figure2.4

may be a linear or a nonlinear function. One of the most commonly used functions is the log-sigmoid transfer function, which is shown in Figure 2. 5.

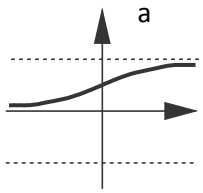


Figure 2.5 Log-Sigmoid Transfer Function

This transfer function takes the input (which may have any value between plus and minus infinity) and squashes the output into the range 0 to 1, according to the expression (2.2)

$$a = \frac{1}{1+e^{-n}} \quad \text{..... (2.2)}$$

The log-sigmoid transfer function is commonly used in multilayer networks that are trained using the back propagation algorithm.Typically; a neuron has more than one input. A neuron with R inputs is shown in Figure 2.6. and its matrix representation as shown in figure 2.7. The individual inputs p_1 , p_2 ,..., p_R are each weighted by corresponding elements $w_{1,1}$,, $w_{1,2}$... ,$w_{1,R}$ of the weight matrix W**.**

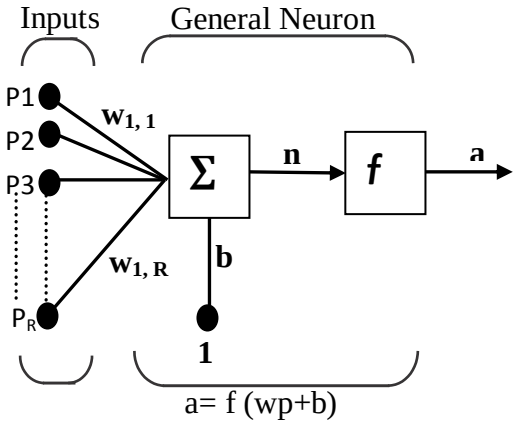


Figure2.6. Multiple-Input Neuron

The neuron has a bias, which is summed with the weighted inputs to form the net input

$$n = w_{1,1}p_1 + w_{1,2}p_2 + w_{1,R}p_R + b \quad \text{..... (2.3)}$$

This expression can be written in matrix form

$$n = Wp + b \quad \text{..... (2.4)}$$

Where, the matrix W for the single neuron case has only one row.

Now the neuron output can be written as

$$a = f(Wp + b) \quad \text{..... (2.5)}$$

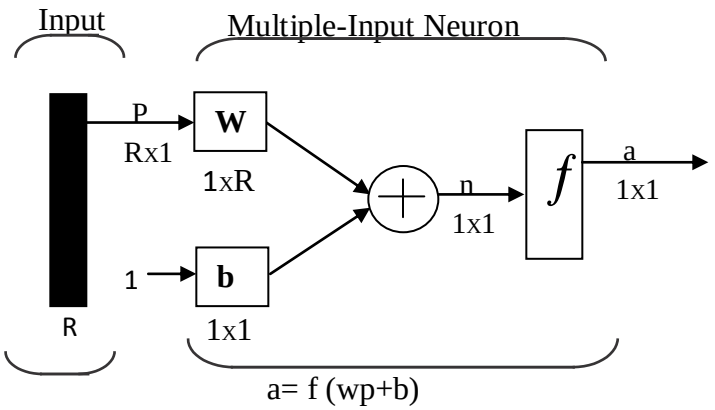


Figure 2.7 Representation of the neuron in matrix form.

2.3.2 Network Architectures

Commonly one neuron, even with many inputs, is not sufficient. We might need five or ten, operating in parallel, in what is called a layer. A single-layer network of S neurons is shown in Figure 2.8. Note that each of the R inputs is connected to each of the neurons and that the weight matrix now has S rows. The layer includes the weight matrix W, the summers, the bias vector b, the transfer function boxes and the output vector a. Some authors refer to the inputs as another layer, but we will not do that here. It is common for the number of inputs to a layer to be different from the number of neurons (i.e., R≠S)

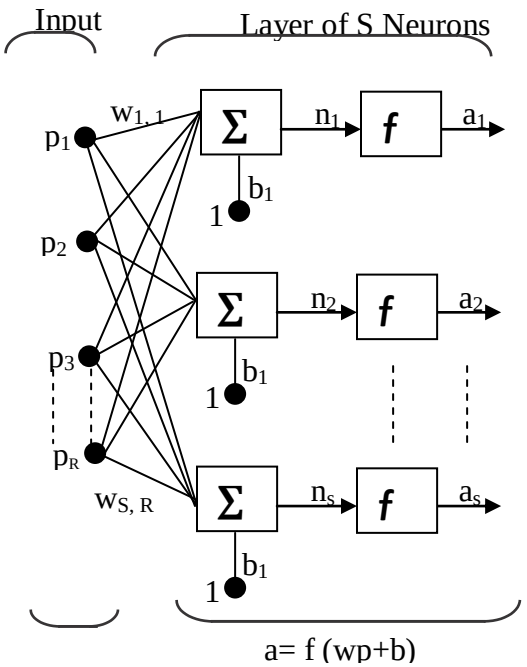


Figure 2.8 Layer of *S* Neurons

The *S*-neuron, *R*-input, one-layer network also can be drawn in matrix notation, as shown in Figure 2.9.

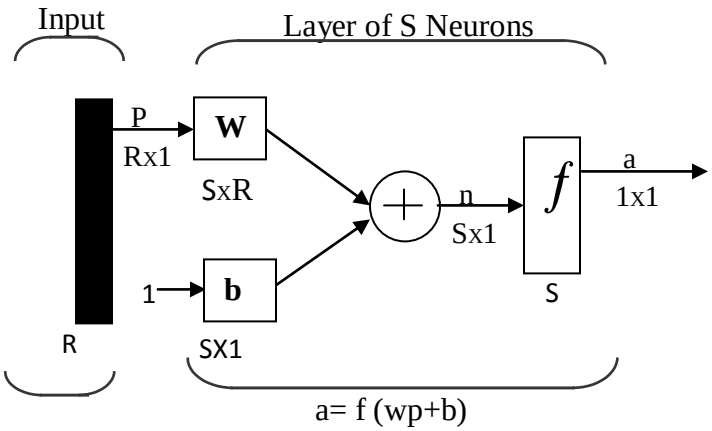
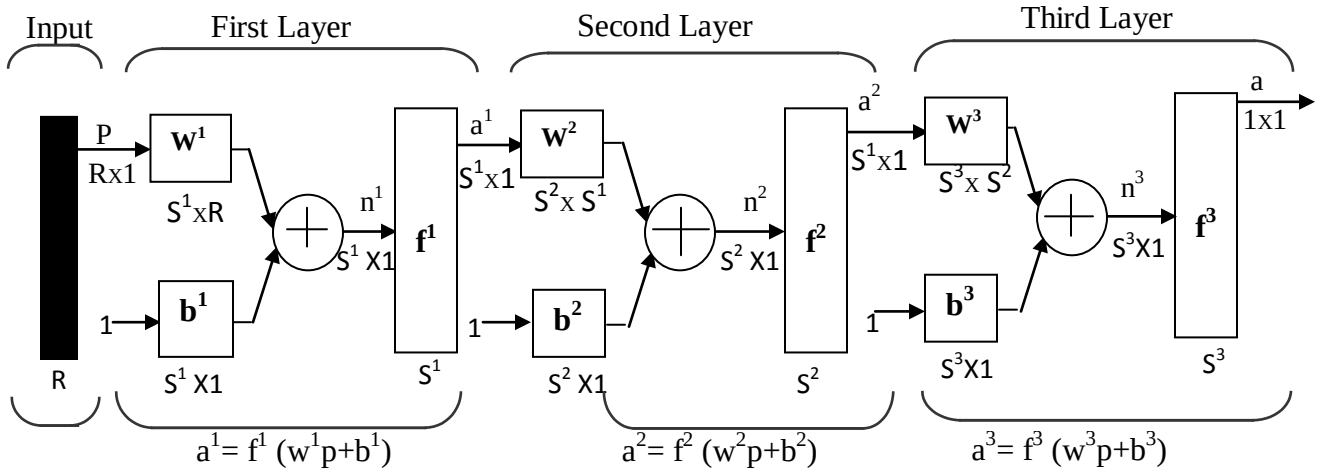


Figure 2.9 Layer of S Neurons, Matrix Notation

2.3.3 Multiple Layers of Neurons

Now consider a network with several layers. Each layer has its own weight matrix W, its own bias vector b , a net input vector n and an output vector a . We need to introduce some additional notation to distinguish between these layers. We will use superscripts to identify the layers. Thus, the weight matrix for the first layer is written as W^1 , and the weight matrix for the second layer is written as W^2 . This notation is used in the three-layer network shown in Figure 2.10. As shown, there are R inputs, S^1 neurons in the first layer, S^2 neurons in the second layer, etc. As noted, different layers can have different numbers of neurons.

The outputs of layers one and two are the inputs for layers two and three. Thus layer 2 can be viewed as a one-layer network with $R = S^1$ inputs $S=S^2$ neurons, and an S^2 ,S^1 weight matrix W^2. The input to layer 2 is a^1, and the output is a^2. A layer whose output is the network output is called an output layer. The other layers are called hidden layers. The network shown in Figure2.15 has an output layer (layer 3) and two hidden layers (layers 1 and 2).



$$a^3= f^3 (w^3f^2 (w^2f^1 (w^1p+b^1)+ b^2)+ b^3)$$

Figure 2.10 Three-Layer Networks

2.3.4 Training of Multilayer Networks

We know multilayer networks are universal approximators; the next step is to determine a procedure for selecting the network parameters (weights and biases) that will best approximate a given function. The procedure for selecting the parameters for a given problem is called training the network. In this section we will outline a training procedure called backpropagation, which is based on gradient descent. (More efficient algorithms than gradient descent are often used in neural network training.)

As we discussed earlier, for multilayer networks the output of one layer becomes the input to the following layer (see Figure2.10). The equations that describe this operation are

$$a^{m+1} = f^{m+1}(W^{m+1}a^m + b^{m+1}), \text{ For m=0, 1, 2… M-1} \quad \ldots\ldots\ldots \quad (2.6)$$

$$a^0 = p \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2.7)$$

This provides the starting point for Eq. (2.7). The outputs of the neurons in the last layer are considered the network outputs:

$$a = a^m \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2.8)$$

The backpropagation algorithm for multilayer networks is a gradient descent optimization procedure in which we minimize a mean square error performance index. The algorithm is provided with a set of examples of proper network behavior:

$$\{p_1, t_1\}, \{p_2, t_2\}, \{p_Q, t_Q\},$$

Where p_q is an input to the network, and t_q is the corresponding target output. As each input is applied to the network, the network output is compared to the target. The algorithm should adjust the network parameters in order to minimize the sum squared error:

$$F(x) = \sum_{q=1}^{Q} e_q^2 = \sum_{q=1}^{Q} (t_q - a_q)^2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2.9)$$

Where is a vector containing all network weights and biases.

CHAPTER 3

MATHEMATICAL MODELING
OF
NARMA L-2 NEURO CONTROLLER

3.1 Mathematical Modeling of NARMA-L2 Neuro Controller

In recent years, a rapidly advancing technology and a competitive market have required systems to operate in many cases in regions in the state space where linear approximations are no longer satisfactory. To cope with such nonlinear problems, research has been underway on their identification and control using neural networks based entirely on measured inputs and outputs. It is now known that under certain conditions, an exact input–output representation of the nonlinear system is given by the nonlinear autoregressive moving average (NARMA) model in a neighborhood of the equilibrium state.

In [18] the two approximations to the nonlinear autoregressive moving average (NARMA) model called the NARMA-L1 and the NARMA-L2 are proposed. From a practical stand-point, the NARMA-L2 model is found to be simpler to realize than the NARMA-L1 model. The controllers used in this section are based only on the NARMA-L2 approximate model.

Multilayer neural networks have been applied successfully in the identification and control of dynamic systems. Rather than attempt to survey the many ways in which multilayer networks have been used in control systems, we will concentrate on three typical neural network controllers: model predictive control [19], NARMA-L2 control, and model reference control [20].These controllers are representative of the variety of common ways in which multilayer networks are used in control systems. As with most neural controllers, they are based on standard linear control architectures.

In spite of the fact that the NARMA model is an exact representation of the system, the approximate nonlinear models are found to be at least as good as the exact model for control purposes (which is quite often much better than a linear model). This can be attributed to the fact

that only approximate methods are used in practice for controlling a plant represented by a NARMA model. The Fig.3.1 shows the block diagram representation of NARMA L-2 controlled SCDM.

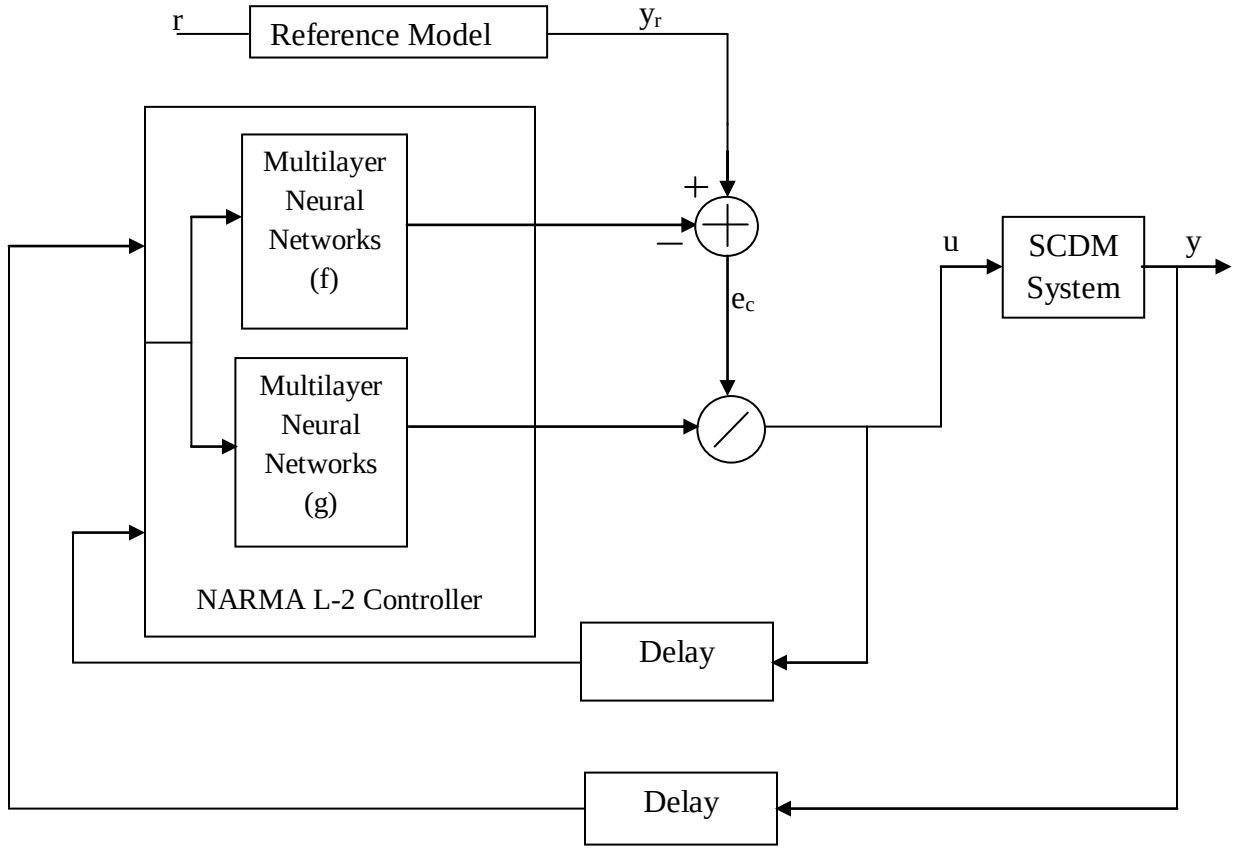


Fig. 3.1 NARMA-L2 controlled SCDM

3.1.1. NARMA-L2 (Feedback Linearization) Control

The Neuro controller described in this section is referred to by two different name such as feedback linearization control and NARMA-L2 control [21]. It is referred to as feedback linearization when the plant model has a particular form (companion form). It is referred to as NARMA-L2 control when the plant model can be approximated by the same form. The central idea of this type of control is to transform nonlinear system dynamics into linear dynamics by canceling the nonlinearities.

3.1.2 Identification of the NARMA-L2 Model

In model predictive control, the first step is feedback linearization to identify the system to be controlled. For training of a neural network to represent the forward dynamics of the system, the first step is to choose a model structure. One standard model to represent general discrete-time nonlinear systems is the nonlinear autoregressive-moving average (NARMA) model expressed by equation (3.1).

$$y(k+d) = N[y(k), y(k-1), \ldots., y(k-n+1), u(k), u(k-1), \ldots., u(k-n+1)] \quad \ldots\ldots.. (3.1)$$

Where u(k) is the system input, and y(k) is the system output and d is the system delay. For the predictive control the system delay is taken as 1. For the identification phase, to train the neural network the nonlinear function N is approximated.

If the system output followed some reference trajectory of $y(k + d) = y_r(k + d)$, to develop a nonlinear controller by using equation(3.2)

$$u(k) = G[y(k), y(k-1), \ldots. y(k-n+1), y_r(k+d), u(k-1), \ldots., u(k-m+1)] \ldots\ldots.. (3.2)$$

The problem to exploit NARMA L-2 Controller is quite slow due to dynamic back-propagation training and it will create a function G to minimize the mean square error. The proposed solution is to approximate model by equation (3.3).

$$\hat{y}(k+d) = f[y(k), y(k-1), \ldots. y(k-n+1), u(k-1), \ldots., u(k-m+1)] + g[y(k), y(k-1), \ldots, y(k-n+1), u(k-1), \ldots., u(k-m+1)].u(k) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.3)$$

This approximated model is in companion form, here the next controller input u(k) is not contained nonlinearity, the advantage of this form is to solve for the control input that causes the system output following the reference $y(k + d) = y_r(k + d)$. The resulting controller represented by equation (3.4)

$$u(k) = \frac{y_r(k+d) - f[y(k), y(k-1), \ldots . y(k-n+1), u(k-1), \ldots ., u(k-n+1)]}{g[y(k), y(k-1), \ldots ., y(k-n+1), u(k-1), \ldots ., u(k-n+1)]}$$

…… (3.4)

The Use of this equation directly can cause realization problem, because it determines the control input u(k) based on the output y(k) simultaneously . So, instead of companion NARMA L-2 model, approximated NARMA L-2 model is used and represented by equation (3.5).

$$y(k+d) = f[y(k), y(k-1), \ldots . y(k-n+1), u(k), u(k-1), \ldots ., u(k-n+1)] + g[y(k), \ldots ., y(k-n+1), u(k), \ldots ., u(k-n+1)].u(k+1)$$

…… (3.5)

Where time delay line d ≥ 2.

The following figure 3.2 shows the structure of NARMA-L2 Plant Model.

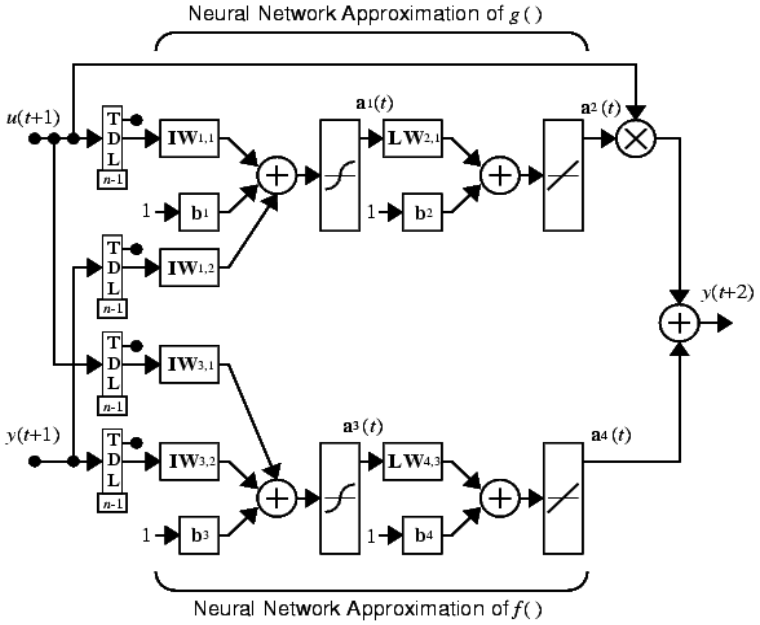


Figure 3.2 Approximated NARMA-L2 Plant Model

3.1.3 NARMA-L2 Controller

Using the approximated NARMA-L2 model, we can obtain the next input $u(k+1)$ represented by equation (3.6)

$$u(k+1) = \frac{y_r(k+d) - f[y(k), \ldots., y(k-n+1), u(k), \ldots., u(k-n+1)]}{g[y(k), \ldots., y(k-n+1), u(k), \ldots., u(k-n+1)]} \quad \ldots\ldots\ldots\ldots\ldots\ldots (3.6)$$

This is realizable for d ≥2.
The following figure 3.3 is a block diagram of the NARMA-L2 controller.

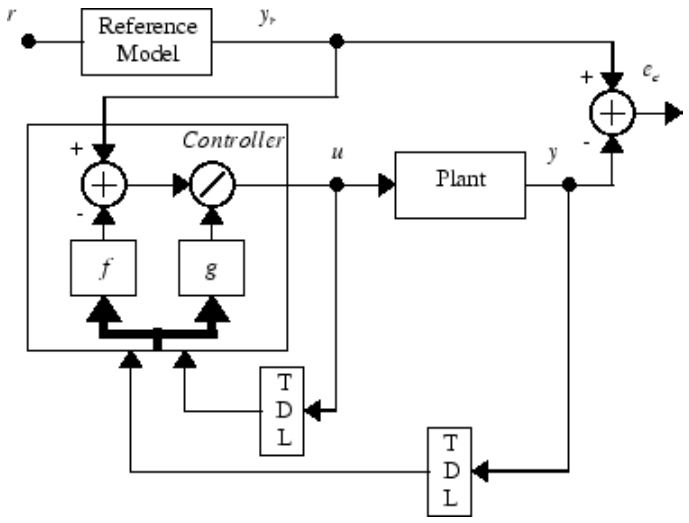


Figure 3.3 Block diagram of NARMA-L2 Controller

This controller can be implemented with the Approximated NARMA-L2 plant model, as shown in the figure3.4.

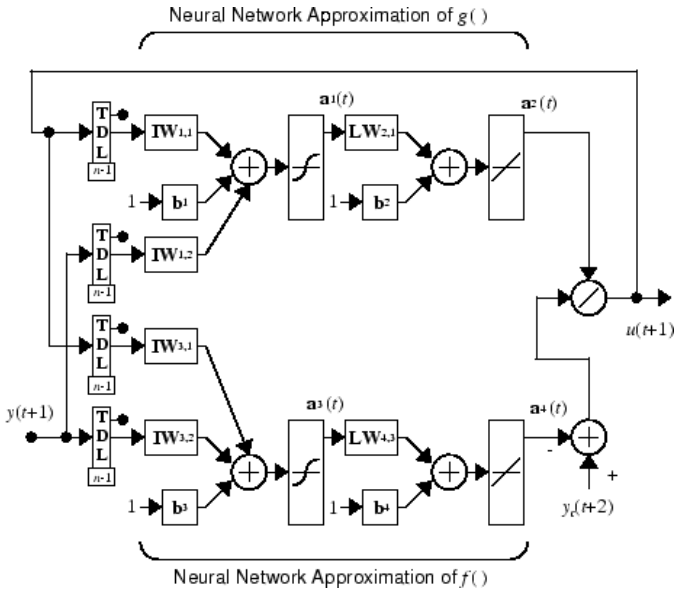


Figure 3.4 Implementation of NARMA-L2 Controller

CHAPTER 4

MODELING AND SIMULATION STUDIES OF SCDM SYSTEM

4.1 Dynamics of SCDM

A series connected dc motor is configured by simply connecting the field circuit in series with the armature circuit (A_1 and A_2), shown in Fig.4.1

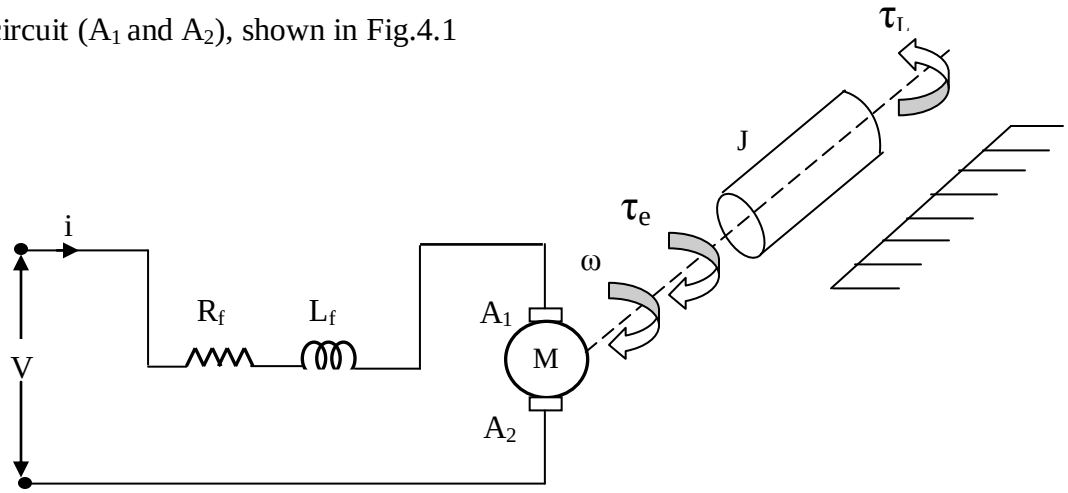


Figure 4.1Equivalent circuit of SCDM drive system

Neglecting magnetic saturation in field circuit, the dynamic equations of a SCDM with $i_a = i_f = i$, $L = L_f + L_a$ and $R = R_a + R_f$ where i_a is armature current and i_f is field current, are described by

$$L\frac{di}{dt} = -Ri - Mi\omega + V$$

$$J\frac{d\omega}{dt} = Mi^2 - \tau_L \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.1)$$

Hence $\tau_e = Mi^2$ and $E = Mi\omega$

Where,
i = Armature current or Field current
V =Terminal control voltage
E = Counter emf
L_a and R_a = Armature inductance and resistance
L_f and R_f = Field inductance and resistance
L = Total armature and field circuit inductance
R = Total armature and field circuit resistance
ω = Rotational speed of the motor
τ_e = Electromagnetic Torque
τ_L = Load Torque
J = Moment of inertia associated with both motor and the load
M= Motor constant

The dynamic model of SCDM is nonlinear due to the terms of $\tau_e = Mi^2$ and $E = Mi\omega$

4.2. Derived model of SCDM from Simulink

4.2.1. Simulation model

The SCDM simulation models is built using resources from the SimPowerSystems and Simulink libraries because there is no Series Connected DC Motor(SCDM) function block available in SimPowerSystem Library. The function block is derived from the "DC machine" function block and saved it in the working directory as SERIES_DC_motor.mdl shown in fig.4.2 and DC_SERIES_NARMA_motor.mdl shown in fig.4.3

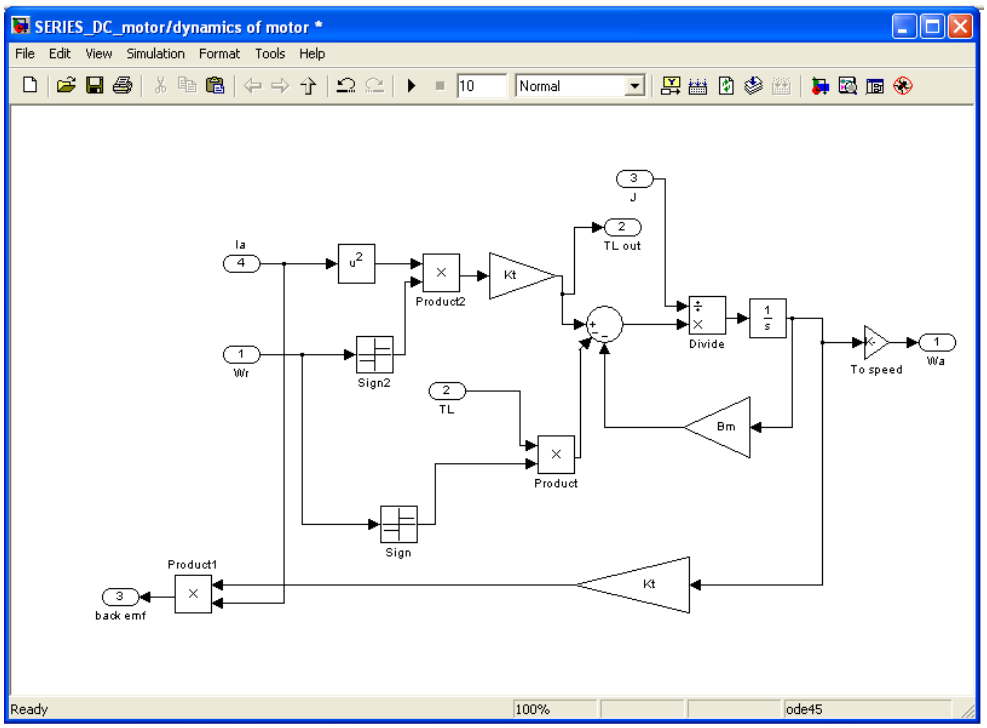


Figure.4.2 SCDM for conventional PID controller

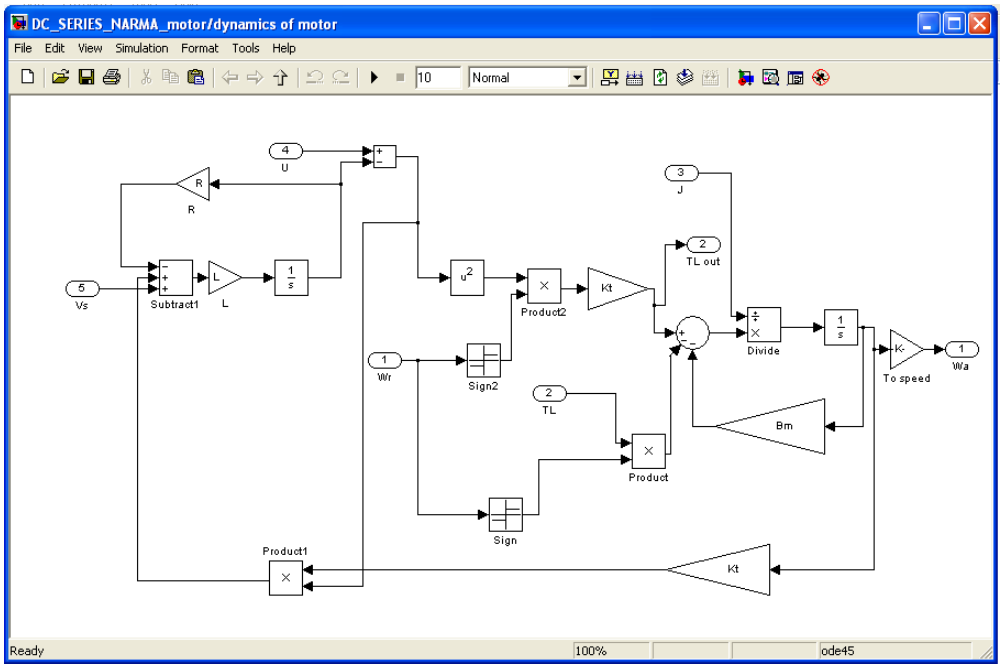


Figure. 4.3 SCDM for NARMA-L2 controller

4.2.2. Function Block with its Parameters

Select all the component of the model to develop mask subsystems for the chopper controlled SCDM shown in fig.4.4 and NARMA-L2 controlled SCDM shown in fig.4.5

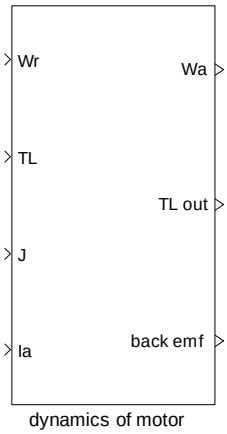


Figure 4.4 Subsystem for the chopper controlled SCDM

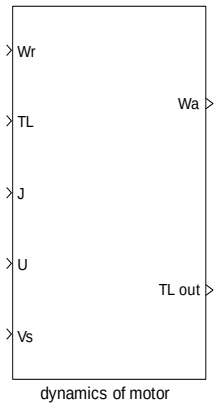


Figure 4.5Subsystem for NARMA-L2 controlled SCDM

4.3 Nonlinear Speed Controllers

Consider a nonlinear PI speed controller by extended linearization technique for the criteria of optimal linear regulator settings [7] is used in the determination of the linearized PI controller gains on the basis of parameterized transfer function of the SCDM system.

For constant input $u(t) = U$, the equilibrium state vector for the following nonlinear system is

$$\dot{x} = f(x) + g(x)u$$

$$y = h(x) \quad \text{..... (4.2)}$$

It can be expressed as a function of U by means of a function x (U). By considering nonlinear system equation (4.2) for the equilibrium operating point (U, x (U)) is

$$\dot{x}_\delta = A(U)X_\delta + B(U)u_\delta$$

$$y_\delta = C(U)x_\delta \quad \text{..... (4.3)}$$

Where, $x_\delta = x(t) - x(U)$

$$u_\delta = u(t) - u(U)$$

$$y_\delta = y(t) - y(U)$$

$$A(U) = \frac{\partial f(x(U))}{\partial x(U)} + \frac{\partial g(x(U))}{\partial x(U)} u$$

$$B(U) = g\big(x(U)\big), c(U) = \frac{\partial h(x(U))}{\partial x(U)}$$

The linearized model (4.3) actually constitutes a family of linearization of model (4.2) which is parameterized by the constant input equilibrium points. Taking Laplace transform in (4.3), under the invertible matrix A(U) and zero initial conditions, the parameterized transfer function equation is:

$$G_U(s) = \frac{y_\delta(s)}{u_\delta(s)} = C(U)\big(sI - A(U)\big)B(U) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.4)$$

Based on this parameterized transfer function, a parameterized PI controller can be chosen in the form of

$$C_U(s) = k_1(U) + \frac{k_2(U)}{s} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.5)$$

Where $k_1(U)$ and $k_2(U)$ can be determined via optimal linear design technique [22].

Now, for constant input $u(t) = U$, the equilibrium state vector for SCDM system (4.1) can be expressed as a function of U by means of a function X (U).

$$i(U) = \sqrt{\frac{\tau_L}{M}}$$

$$\omega(U) = \frac{U - Ri}{\sqrt{MT_L}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.6)$$

About any such point, the linearized model of (4.1) from assuming the output being ω is

$$\begin{bmatrix} \frac{d\Delta i}{dt} \\ \frac{d\Delta\omega}{dt} \end{bmatrix} = \begin{bmatrix} -\frac{R + M\omega(U)}{L} & -\frac{Mi(U)}{L} \\ \frac{2Mi(U)}{J} & 0 \end{bmatrix} \begin{bmatrix} \Delta i \\ \Delta\omega \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} \Delta V_t \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.7)$$

$$\Delta y = \Delta\omega$$

Where $\Delta i = i - i(U), \Delta y = \omega - \omega(U), \Delta u = u = U$

The parameterized transfer function relating the incremental motor speed ω to the incremental terminal voltage U for the linearized model (4.7) can be written in the form

$$G_U(s) = \frac{\Delta\omega(s)}{\Delta u(s)} = \frac{k_d}{(T_1 s + 1)(T_2 s + 1)} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.8)$$

$$\text{Where, } k_d = \frac{1}{Mi(U)}$$

$$T_1 = \frac{J\big(R + M\omega(U)\big) - \sqrt{J^2(R + M\omega(U))^2 - 8M^2 i^2(U) JL}}{4M^2 i^2}$$

$$T_2 = \frac{J(R+M\omega(U))}{2M^2 i^2(U)} - T_1 \qquad \text{..... (4.9)}$$

Based on the design of linear regulator setting, the values of proportional and integral gains k_1 (U) and k_2 (U) are given

$$k_1(U) = \frac{Mi(U)T_2}{2T_1}$$

$$k_2(U) = \frac{Mi(U)}{2T_1} \qquad \text{..... (4.10)}$$

The PI controller represented by (4.10) is such that it would stabilize the output response of the whole family of linearized model (4.7) to zero. The block diagram of the parameterized linear closed-loop system is shown in Fig. 4.6.

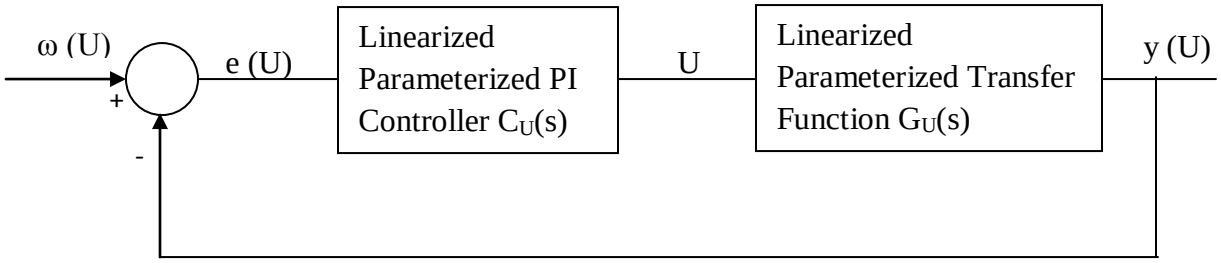


Figure 4.6 Parameterized linear closed- loop system

To introduce a PID controller in the parameterized linear closed loop system by using integral and differential corrector in parallel with the proportional gain and then start manual correction in the proportional, integral and differential gain parameter to get stable PID controller.

4.4 Speed control of SCDM using 4-Quadrant chopper

Figure 4.7 shows the speed control simulation model of SCDM using 4-Quadrant chopper. It consists of a SCDM fed by a DC source through 4-Quadrant chopper as shown in figure 4.8 and its subsystem as shown in figure 4.9 and PID controllers as shown in figure 4.10. The motor drives a mechanical load characterized by inertia J, friction coefficient D, and a load torque τ_L. The control circuit consists of a speed control loop and a current control loop. A proportional-integral derivative (PID) controlled speed control loop senses the actual speed of the motor and compares it with the reference speed to determine the reference armature current required by the motor. One may note that any variation in the actual speed is a measure of the armature current required by the motor. The current control loop consists of a 4-Quadrant chopper. The 4-Quadrant chopper[23] consists of a relay, which is used to generate switching patterns required for the chopper circuit. The speed control loop consists of PID Controller and a current limiter. Current limiter limit the speed control output so that an input current is generated which follows a reference current waveform. The difference between the desired current, and the current being injected is used to control the switching of the chopper circuit. When the error reaches an upper limit namely upper hysteresis limit, IGBT is switched to force the current down. On other hand when the error reaches the lower hysteresis limit, a positive pulse is produced to increase the current. The minimum and maximum values of the error signal are e_{min} and e_{max} , and the range of the error signal, $(e_{max} - e_{min})$, directly controls the amount of the ripple in the output current and is called the hysteresis band. Thus the armature current is forced to stay within the hysteresis band determined by the upper and lower hysteresis limits.

4.4.1 Simulink model

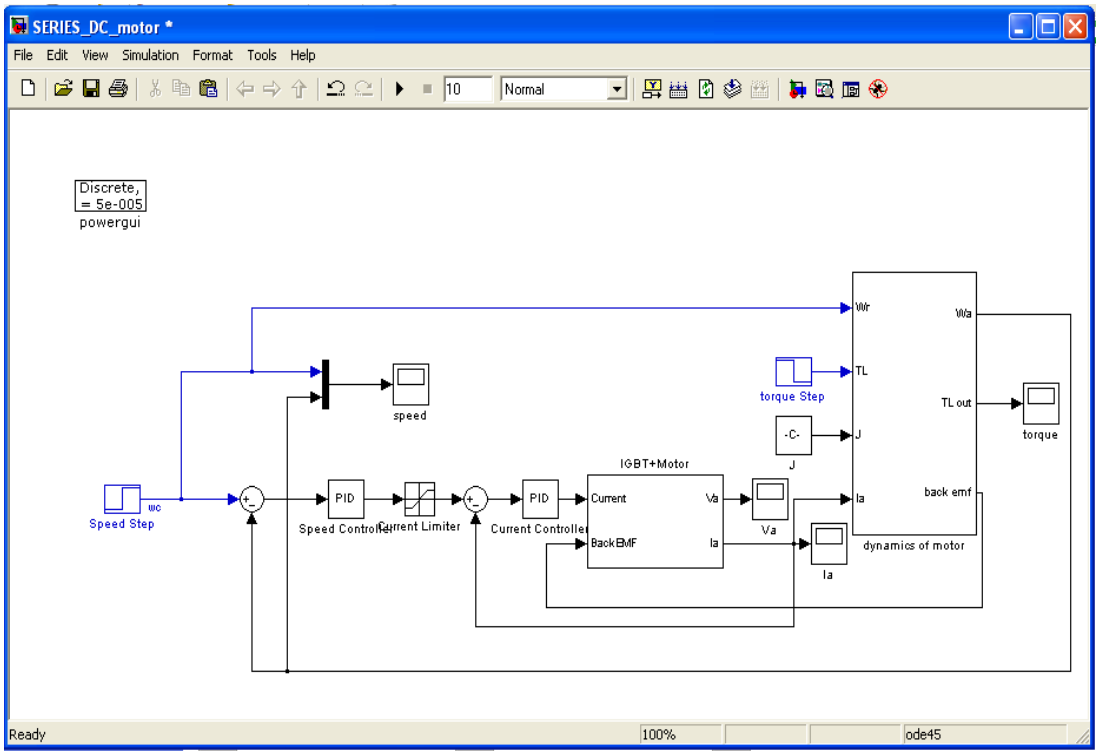


Figure 4.7 Simulink model for Speed control of DCSM using 4-Quadrant chopper

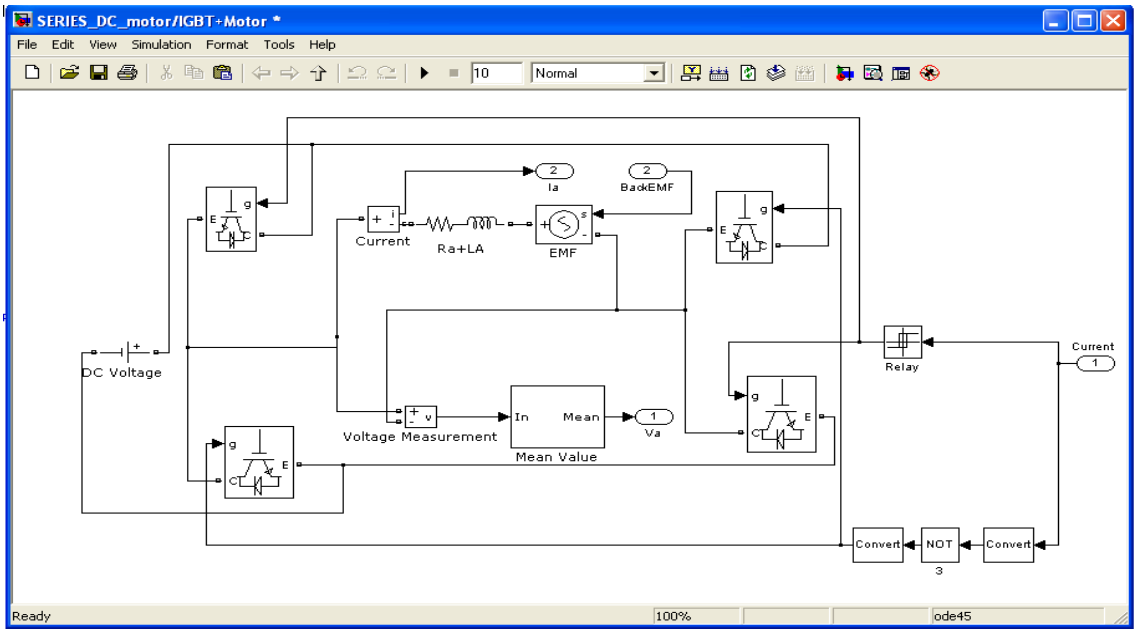


Figure 4.8 Simulink model of Four Quadrant chopper model

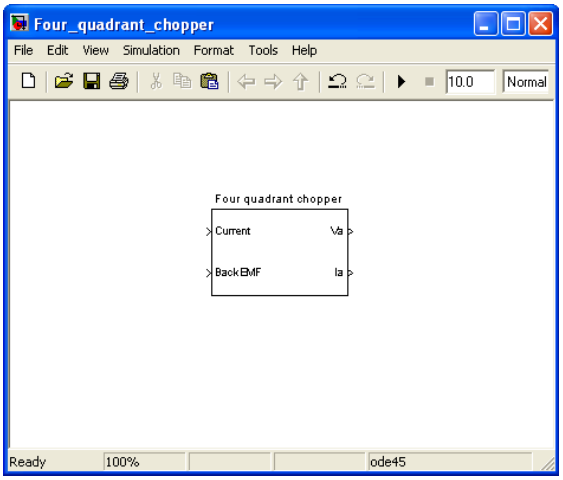


Figure 4.9 Subsystem of Four Quadrant chopper model

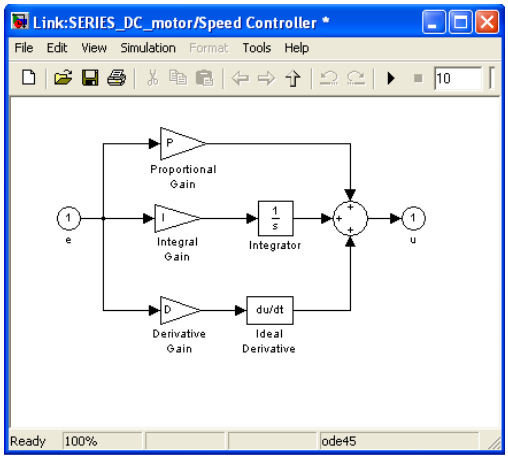


Figure 4.10 Simulink model for PID Controller

4.5 Speed control of SCDM using NARMA-L2 Controller

4.5.1 NARMA-L2 Controller

The learning ability, self-adapting, and super-fast computing features of ANN make it well suited for the control of DC motor. In learning process, neural network adjusts its structure such that it will be able to follow the supervisor. The learning is repeated until the difference between network output and the supervisor is low. MATLAB based Subsystem for NARMA L-2 Controller as shown in figure 4.11

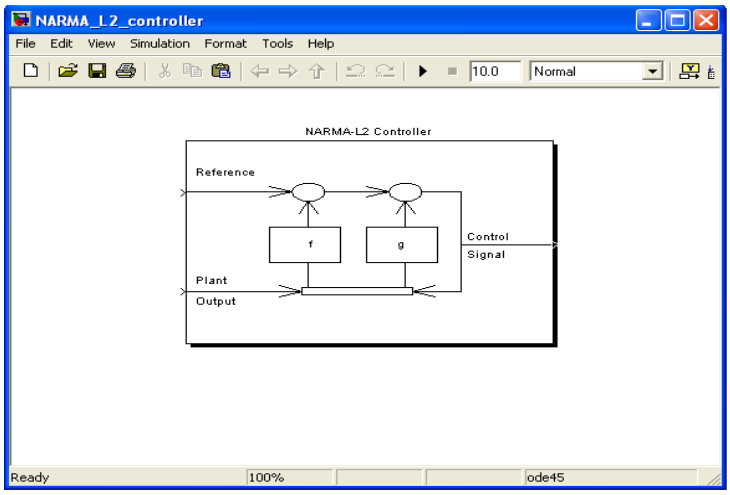


Figure 4.11. MATLAB based Subsystem for NARMA L-2 Controller

4.5.2 System Identification and controller design Stage

In NARMA-L2 controller, a multilayer neural network has been successfully applied for the identification and control of dynamic systems [24]. There are two steps involved in NARMA-L2 controller which are System identification and control design. The system identification stage developed a neural network model of the plant to be controlled. The control design stage, use the neural network plant model to train the controller [25], figure4.12 shows the neural network training. In the system identification stage a neural network plant model must be developed before the controller is used. The plant model predicts future plant outputs. The specifications of the plant model are given in figure 4.13

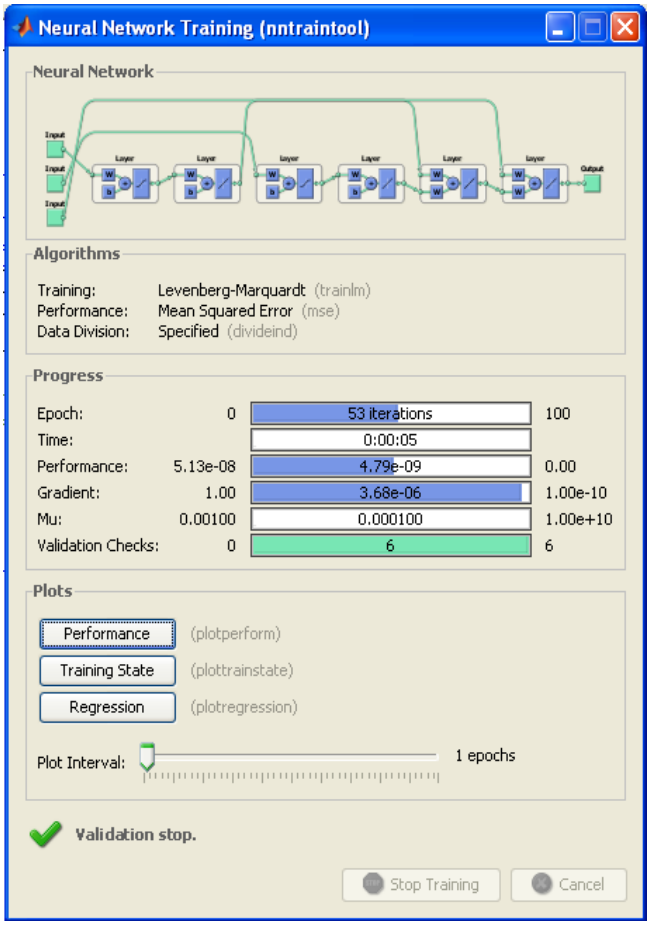


Figure 4.12 neural network training

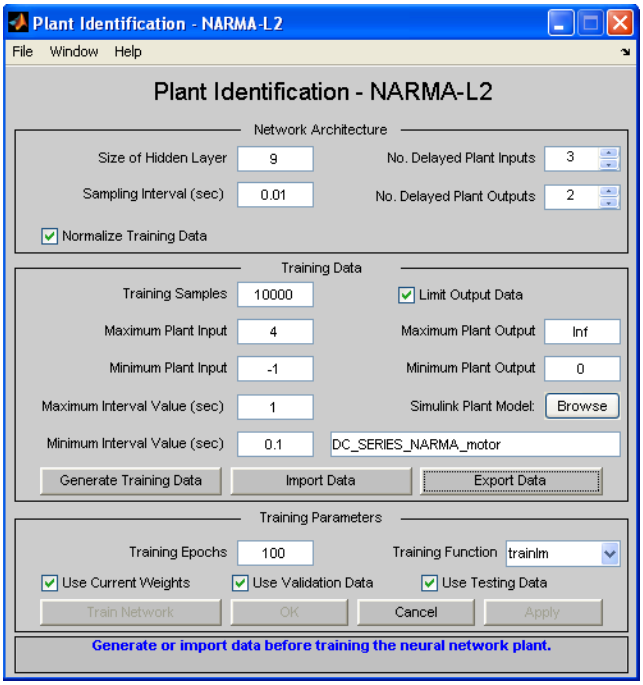


Figure 4.13 specifications of the plant model

Figure 4.14 shows the Plant input-output data of NARMA-L2 controller. Sample performance, training state and regression graph as shown in figure 4.15, 4.16 and 4.17 respectively and testing, training and validation data obtained from a NARMA-L2 controller are illustrated in Figure 4.18,4.19 and 4.20 respectively.

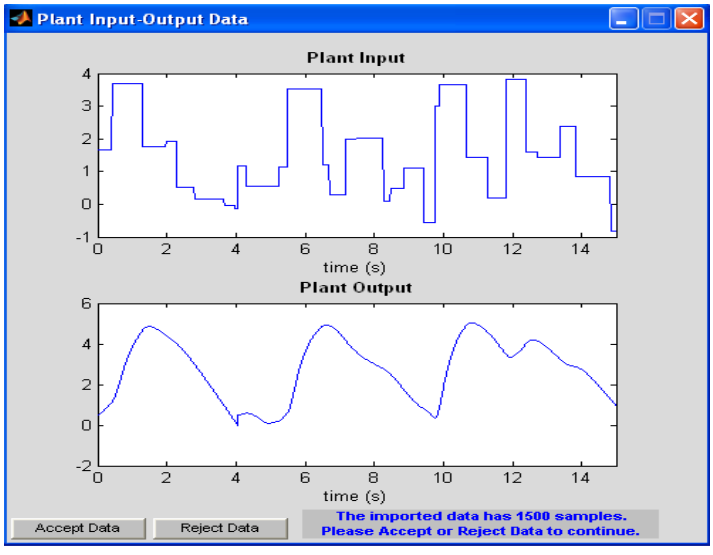


Figure 4.14 SCDM input-output data

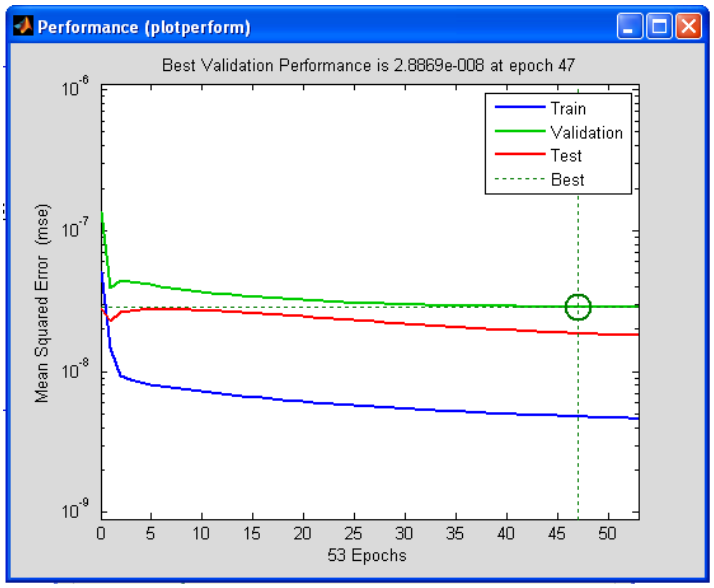


Figure 4.15 Performance of NARMA L-2 controller

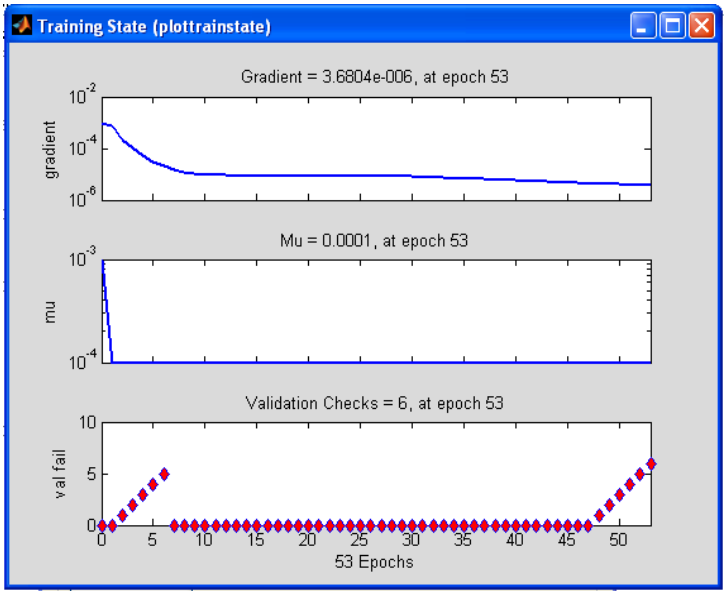


Figure 4.16 Training state of NARMA L-2 controller

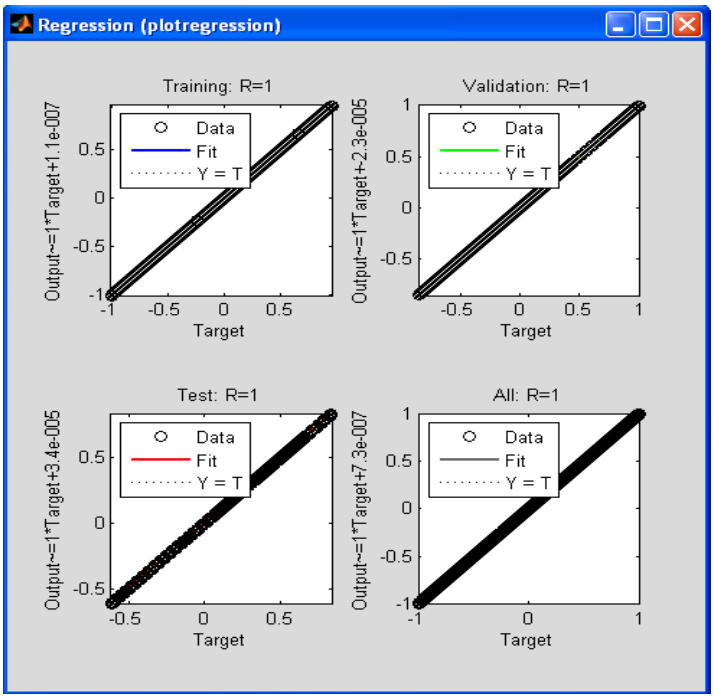


Figure 4.17 Regression of NARMA L-2 controller

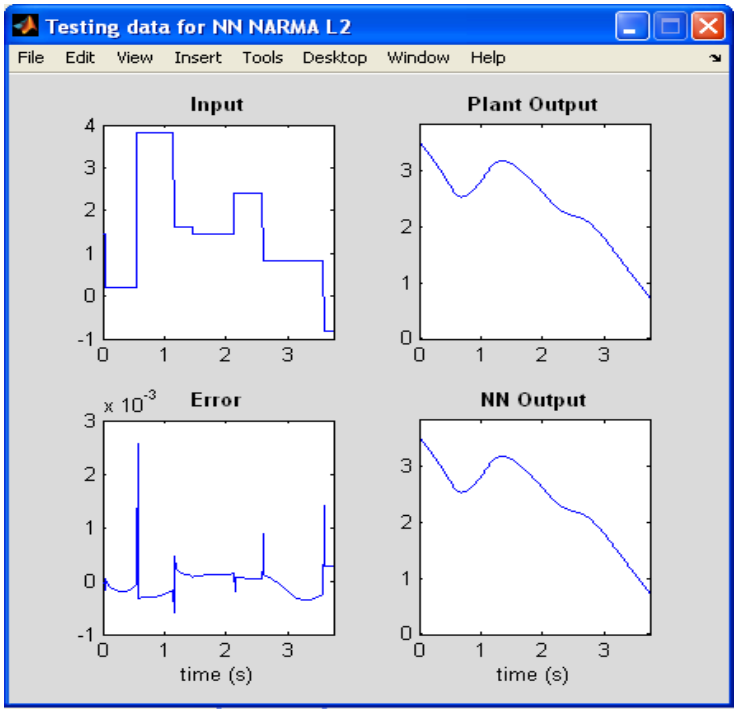


Figure 4.18 Testing data of NARMA L-2 controller

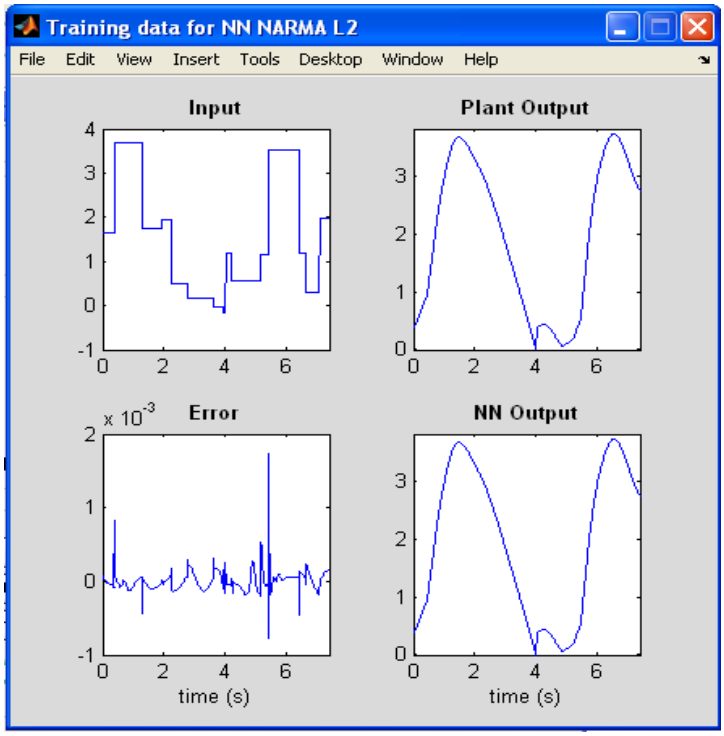


Figure 4.19 Training data of NARMA L-2 controller

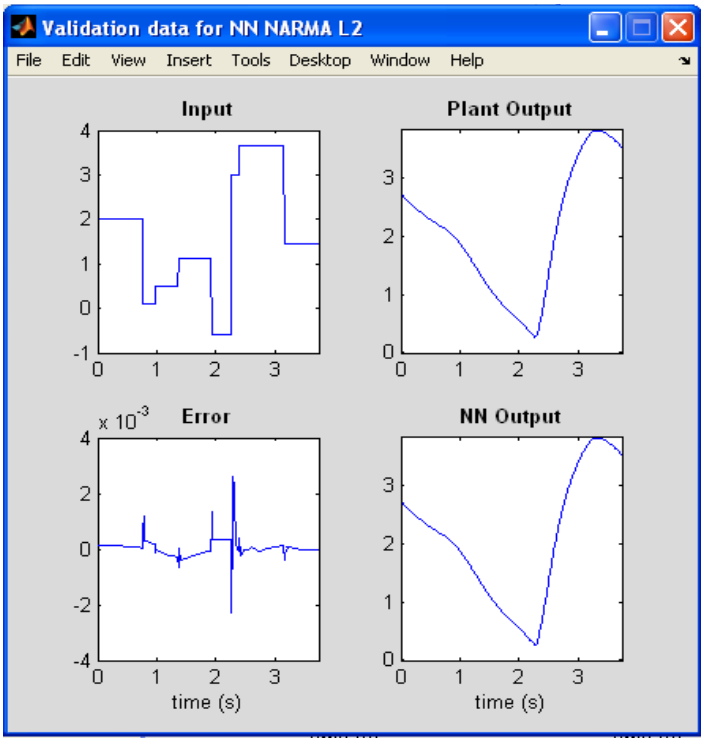


Figure 4.20 Validation data of NARMA L-2 controller

4.5.3 Implementation of NARMA L-2 Neuro controller

The Implementation of NARMA L2 Neuro Controller for dynamic model of SCDM equation (4.1).Consider the nonlinear change of coordinates

$$x_1 = \omega$$

$$x_2 = \frac{2M}{J} i^2 - \frac{\tau_L}{J} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.11)$$

In these new coordinates, the system is represented by

$$\frac{dx_1}{dt} = x_2$$

$$\frac{dx_2}{dt} = \frac{2M}{J} i \frac{di}{dt}$$

$$= \frac{2M}{JL} i(-Ri - Mi\omega + V) \quad \ldots\ldots\ldots\ldots\ldots\ldots (4.12)$$

Where, the load torque is assumed to be constant.

To linearize the system using feedback, the input voltage is V set as

$$V = \frac{JL}{2Mi} u + Ri + Mi\omega \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.13)$$

Resulting in the linear system

$$\frac{dx_1}{dt} = x_2$$

$$\frac{dx_2}{dt} = u$$

Where u is a new control input

The linearized controller equation (4.13) is singular, which is simply a consequence of the fact that the motor cannot produce torque without current. Now from equation (4.12)

$$\frac{dx_2}{dt} = \frac{2M}{J} i \frac{di}{dt} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.14)$$

On integrating equation 4.14 the square current is,

$$i^2 = \frac{J}{2M} x_2$$

Again,

$$\frac{dx_2}{dt} = \frac{2M}{JL} i(-Ri - Mi\omega + V)$$

$$\frac{dx_2}{dt} = -\frac{2M}{JL}i^2 - \frac{2M^2 i^2 \omega}{JL} + \frac{2M}{JL}iV \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4.15)$$

Substituting the value of $i^2 = \frac{J}{2M}x_2$ in equation (4.15)
consider $x_1 = y, \dot{x} = \dot{y} = x_2, \dot{x}_2 = \ddot{x}_1 = \dot{y}$. The model of the nonlinear plant can be written as

$$\ddot{y}(t) + \frac{R}{L}\dot{y}(t) + \frac{M}{L}y(t)\dot{y}(t) = u(t) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.16)$$

This is a second order nonlinear plant which can be modeled as *n*-dimensional discrete time as

$$y(k+2) = f[y(k+1), y(k)] + g[y(k+1), y(k)]u(k) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4.17)$$

The control input to the plant is in the form of:

$$u(k+1) = \frac{y_r(k+d) - f[y(k), y(k-1)]}{g[y(k), y(k-1)]} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.18)$$

4.5.4 Simulink model of NARMA L-2 Controlled SCDM

The simulink model of a NARMA-L2 controlled SCDM is shown in Fig.4.21. The inputs of the controller are the reference speed and the output is actual speed and the control input is driving current to the motor.

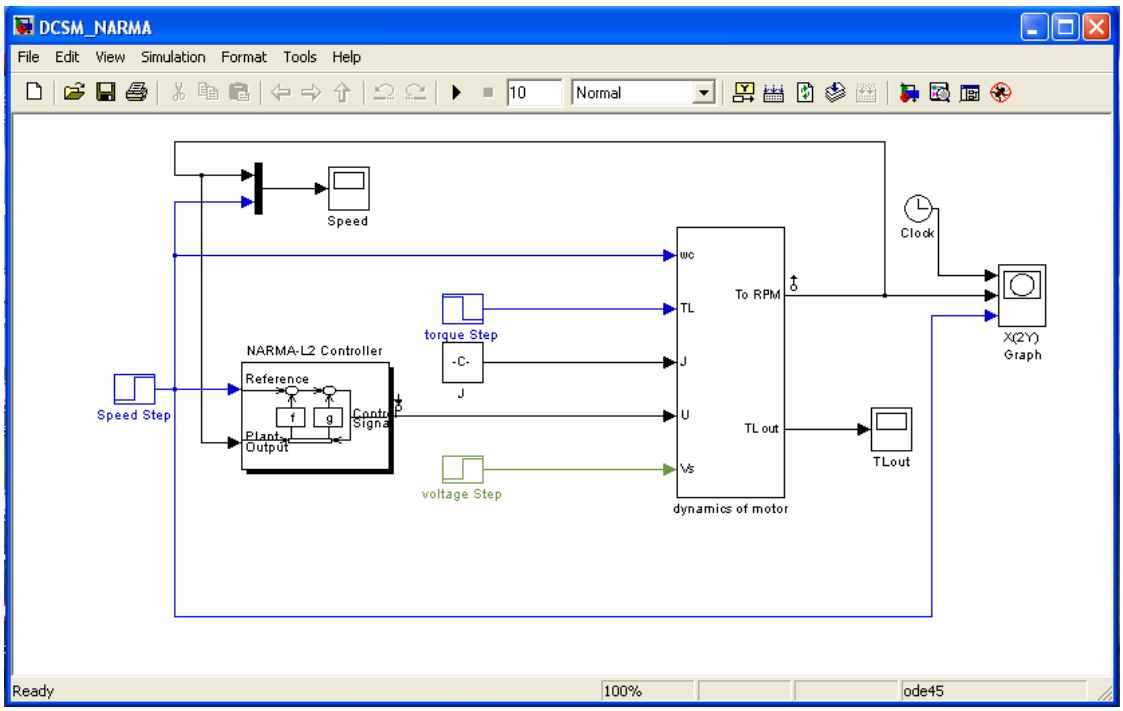


Figure 4.21 NARMA-L2 controlled SCDM

CHAPTER 5

RESULTS AND DISCUSSION

5.1 SCDM system parameters

To demonstrate the effectiveness of the proposed two nonlinear speed controllers such as conventional PID controller and NARMA L-2 Neuro controller, the computer simulations have been carried under step changes in speed reference and load torque. The parameters used for simulation of SCDM system are given in table 5.1

Table 5.1 Series connected DC Motor: list of parameters

Rated power	1.5 kW
Rated Armature current (I)	5A
Definition speed (ω)	600 rev/min
Maximum speed (ω_{max})	5000rev/min
Minimum Supply voltage(V_s)	40V
Maximum supply voltage (V_{max})	200V
Total armature and field circuit inductance(L)	0.0917 H
Total armature and field circuit resistance (R)	7.2Ω
Viscous damping torque constant(D)	0.0004N.m/rad/s
Motor constant (M)	0.1236Nm/Wb.A
Moment of inertia associated with both motor and the load(J)	0.0007046 kg-m^2

5.2 Speed responses of SCDM system

5.2.1 PID controller

The speed response of PID controlled SCDM under operating conditions of 200V and 2 kN with increase of speed reference from 340rad/sec to 500rad/sec. The PID parameters are calculated from the equations (4.6), (4.9) and (4.10), given in Table 5.2. The corresponding speed response is shown in figure 5.1, which observes more overshoot but usually overshoot is undesirable phenomenon in precise system.

Table 5.2 gains for PID Controller

i(U)	ω(U)	T_1	T_2	$k_1(U)$	$k_2(U)$	k_D
4.02A	342.4 rad/sec	0.0057 sec	0.17 sec	1.236	20.6	0.15

To get the stable speed regulation of SCDM the modified gains of PID controller given in table 5.3, which observes the less overshoot in figure 5.3

Table 5.3 gains for PID Controller with modification

i(U)	ω(U)	T_1	T_2	$k_1(U)$	$k_2(U)$	k_D
4.02A	342.4 rad/sec	0.0057 sec	0.17 sec	12.3	21	0.15

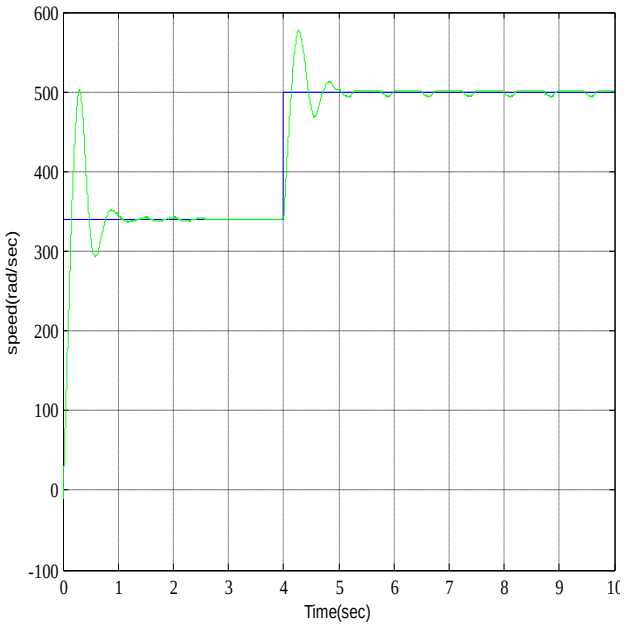


Figure 5.1 Speed response of SCDM

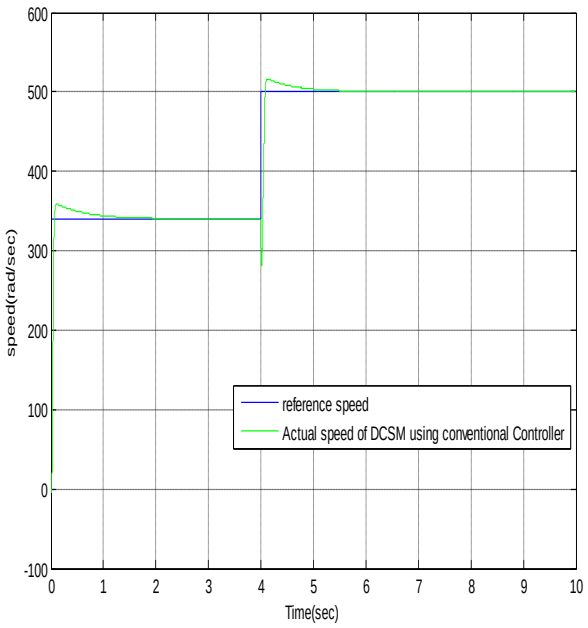


Figure 5.2 Speed response of SCDM with modified PID gains

The corresponding Armature current and load torque response is shown in figure 5.3 and figure 5.4 respectively.

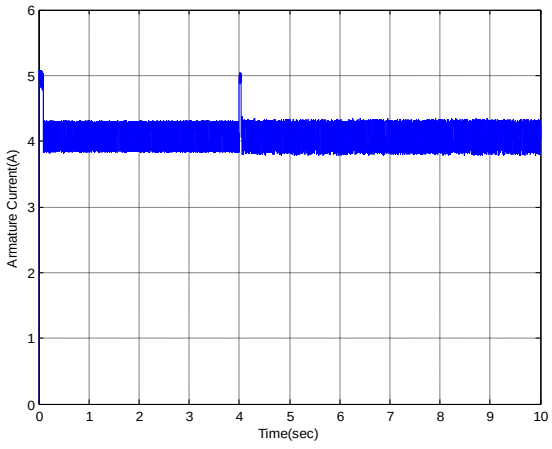


Figure5.3 Armature current

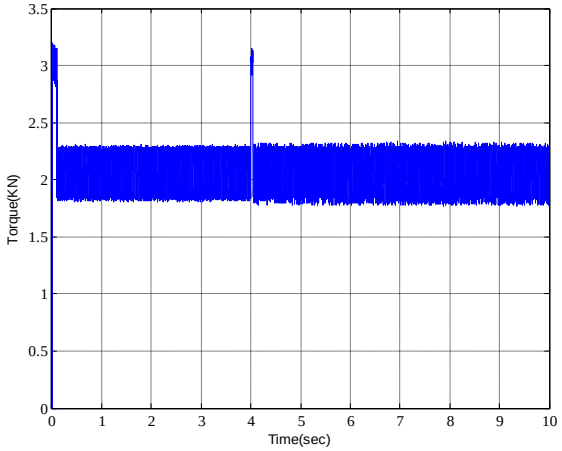


Figure 5.4 Load Torque with PID controller

5.2.2 NARMA-L2 controller

The speed response of NARMA L-2 controlled SCDM under operating conditions of 200V and load torque of 0.260 kN to 5.45 kN with increase of speed reference from 340rad/sec to 500rad/sec as shown in figure 5.5 which observes no overshoot and corresponding load torque as shown in figure 5.6

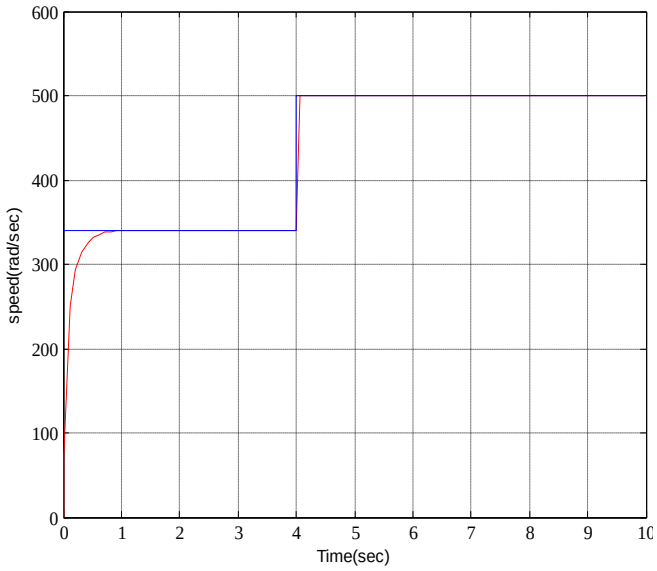


Figure 5.5 Speed response of SCDM with NARMA L-2 controller

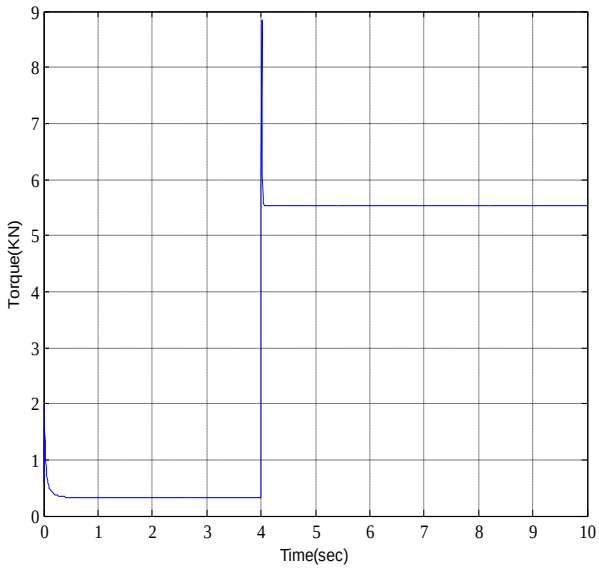


Figure 5.6 Load Torque with NARMA L-2 controller

5.3 Comparison of Speed responses of SCDM by PID and NARMA L-2 controller

Figure 5.7 displays the comparative speed response of SCDM with increase of speed reference from 340rad/sec to 500rad/sec and the operating voltage is 200V under the steady load torque of 2 kN for conventional PID controller and 0.260 kN to 5.45 kN for NARMA L-2 controlled SCDM. The corresponding parameter values for both the controller as given in table 5.4 which shows the comparison of parametric values of controllers.

Table 5.4 Parametric values of both the controllers with 200V

Parameters	rise time(sec)		peak overshoot		steady state error	
Speed(rad/sec)	340	500	340	500	340	500
PID Controller	0.11	4.16	0.44	0.23	0.005	0.01
NARMAL-2 Controller	0.75	4.55	0.29	0.06	1.7	0.05

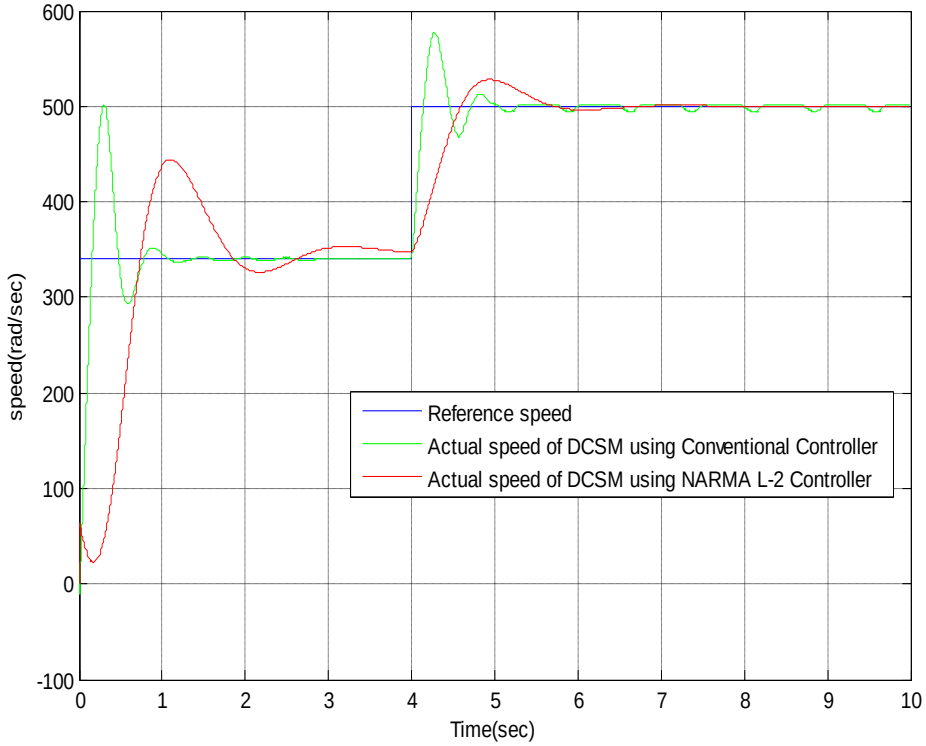


Figure5.7 Comparison of speed responses for SCDM with 200V

Figure 5.8 displays the comparative speed response of SCDM with increase of speed reference from 340rad/sec to 500rad/sec and the operating voltage is 40V under the load torque of 0.250 kN to 0.099 kN for conventional PID controller and 0.260 kN to 0.210 kN for NARMA L-2 controller. The corresponding parameter values for both the controllers are given in table 5.5 which shows the comparison of parametric values of controllers.

Table 5.5 Parametric values of both the controllers with 40V

Parameters	rise time(sec)		peak overshoot		steady state error	
Speed(rad/sec)	340	500	340	500	340	500
PID Controller	0.28	0.5	0.09	0.06	0.734	0.165
NARMAL-2 Controller	0.72	1.01	no overshoot	no overshoot	0.74	0.16

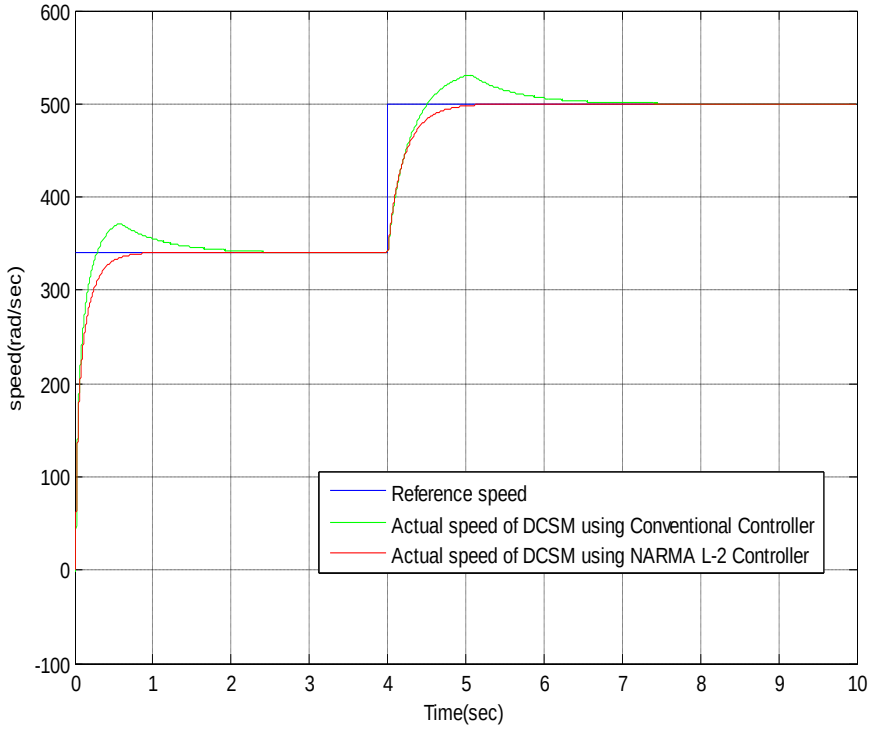


Figure 5.8 Comparison of speed responses for SCDM with 40V

Figure 5.9 displays the comparative speed response of SCDM with increase of speed reference from 340rad/sec to 500rad/sec and the operating voltage is 200V under the load torque 0.260 to 2 kN for conventional PID controller and the load torque 0.260 KN to 5.45KN for NARMA L-2 controller. The corresponding parameter values for both the controller as given in table 5.6 which shows the comparison of desired parametric values of controllers.

Table 5.6 Desired parametric values of both the controllers with 200V

Parameters	rise time(sec)		peak overshoot		steady state error	
Speed(rad/sec)	340	500	340	500	340	500
PID Controller	0.02	0.08	0.05	0.03	0.81	0.35
NARMAL-2 Controller	0.7	0.04	no overshoot	no overshoot	0.7	0.34

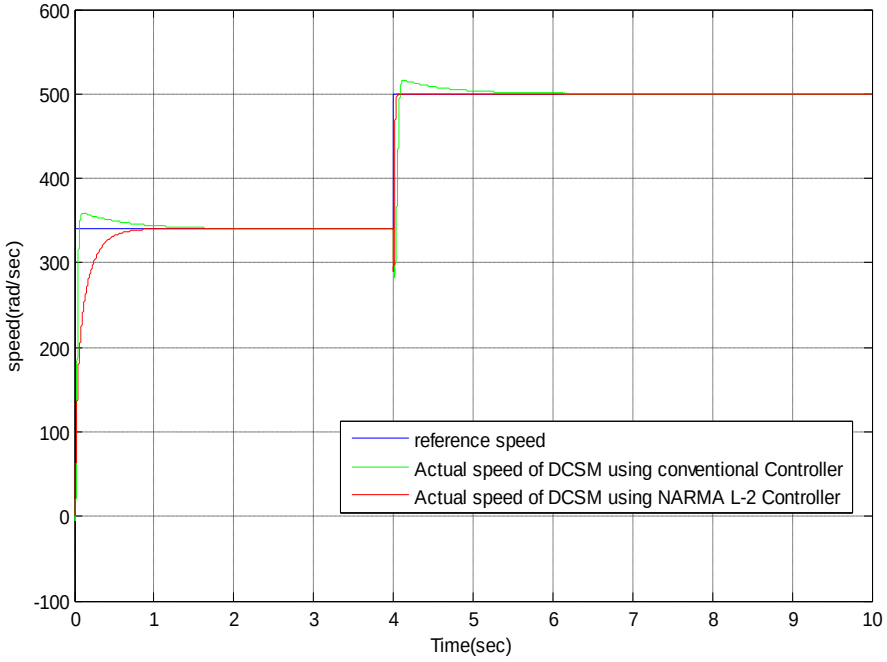


Figure 5.9 Desired Speed responses of both the controller for SCDM

CONCLUSION

The conventional PID and NARMA-L2 controllers have been successfully developed and simulated using MATLAB to control the speed of a Series connected DC Motor. Simulation results show effectiveness of these two controllers for dealing with the motor system with nonlinearity under wide dynamic operation regimes. In comparison with the conventional PID controller, the proposed NARMA L-2 controller has the advantages of no overshoot and excellent speed tracking performance.

FUTURE SCOPE

Possible future work in this area that shows promise of further improvement in the performance of series connected DC drive by developing hardware model.

REFERENCES

[1] Peter Vas, "Artificial-intelligence-based electrical machines and drives: application of fuzzy, neural,fuzzy-neural and genetic-algorithm-based techniques" Oxford university press,1999.

[2] Sudarshan K Valluru and T.N. Rao, "Introduction to Neural Networks, Fuzzy Logic and Genetic Algorithms" Jaico Publishing House, Mumbai, 2011.

[3] Wang. J.L., and W**.** J. Rugh, "Feedback Linearization Families for Nonlinear Systems," IEEE Trans. Automatic Control, Vol. 32. NO. IO, 1987**,** pp. 935-940.

[4] Darren M.Dawson,Jun Hu,and Timothy C.Burg " Nonlinear control of electric machinery"Marcel Dekker, Inc, NY,1998.

[5] H. Sira-Ramirez. "Design of P-I Controllers for DC-to-DC Power Supplies via Extended Linearization." International Journal of Control. Vol. 51, NO. 3, 1990, pp. 601-620.

[6] M.H.Rashid, "Dynamic Responses of DC Chopper-Controlled Series Motor" IEEE Transactions on industrial electronics and control instrumentation, Vol.-28,No.4 pp.323-330, 1981.

[7] Zuo Z. Liu and Fang L. Luo, M. H. Rashid,' Nonlinear Speed Controllers for Series DC Motor,' IEEE International Conference on Power Electronics and Drive System, PEDS'99, July 1999, Hong Kong.

[8] Mehta. S and John Chiasson "Nonlinear Control of a Series DC Motor: Theory and Experiment", IEEE Tran. Indu. Electron, Vol. 45, pp.134-141, 1998.

[9] Dongbo Zho Ning Zhang "An Improved Nonlinear Speed Controller for Series DC Motors" Proceedings of the 17th world congress, IFAC, Seoul, Korea, pp.11047-11052. July 6-11,2008.

[10] P. Chevrel, L. Sicot, and S. Siala, "Switched LQ controllers for DC motor speed and current control", a comparison with cascade control," 27th Annual IEEE Power Electronics Specialists Conf. 1996,vol.1, pp. 906-912, June 23-27,1996.

[11] K. S. Narendra and S.Mukhopadhyay, "Adaptive control using neural networks and approximate models," IEEE Trans. Neural Networks, vol.8, pp. 475-485, May 1997.

[12] A. Pukrittayakame, O. De Jesus and M.T. Hagan."Smoothing the control action for NARMA-L2 controllers," Midwest Symposiom on Circuits and System, vol.3, pp: III 37-40, 2002.

[13] O. Adetona, S. Sathananthan, and L. H. Keel, "Robust Adaptive Control of Non affine Nonlinear Plants With Small Input Signal Changes", IEEE Transactions on neural networks, vol. 15, no. 2, pp.408-416,2004.

[14] Jeffrey T. Spooner,Manfredi Maggiore,Raul Ordonez, Kevin M. Passino "Stable adaptive control and estimation for nonlinear systems: neural and fuzzy approximator techniques."Wiley interscience,2002

[15] Sergey E. Lyshevski. "Electromechanical Systems and Devices", CRC Press, New York, 2008.

[16] Hagan, M. T. and H.B. Demuth, 'Neural Networks for Control,' Proceedings of the 1999 American Control Conference, San Diego, CA, pp1642-1656,1999.

[17] Hagan, M. T. and H.B. Demuth and Mark Beale, "Neural network design", PWS publishing company,USA,1996.

[18] O. Adetona, S. Sathananthan, and L. H. Keel, " Approximations the NARMA model of Non affine Plants" Proceeding of the American Control Conference Boston, Massachusetts, 30th June -2nd July,pp.5502-5507,2004

[19] Soloway, D. and P.J. Haley, 'Neural Generalized Predictive Control,' Proceedings of the IEEE International Symposium on Intelligent Control, pp. 277-281,1996.

[20] Narendra, K.S. and K. Parthasarathy, 'Identification and Control of Dynamical Systems Using Neural Networks,' IEEE Transactions on Neural Networks, vol. 1, pp.4-27 ,1990.

[21] M. T. Hagan, H. B. Demuth, and O. D. Jesús, "An introduction to the use of neural networks in control systems," International Journal of Robust and Nonlinear Control, John Wiley & Sons, Vol. 12, no. 11, pp. 959-985, September 2002

[22] B.D.O. Anderson & J.B. Moore, "Optimal Control: Linear Quadratic Methods", Prentice-Hall, NY, 1990.

[23] J. Santana, J. L. Naredo, F. Sandoval, I. Grout, and O. J. Argueta, "Simulation and construction of a speed control for a DC series motor," Mechatronics, vol. 12, issues 9-10,. pp. 1145-1156, 2002.

[24] M. H. Beale, M. T. Hagan, H. B. Demuth "Neural Network Toolbox™ User's Guide" The MathWorks Inc, 2012.

[25] M. D. Minkova, D. Minkov, J. L. Rodgerson, and R. G. Harley, "Adaptive neural speed controller of a dc motor," Electric Power Systems Research, vol. 47, issue 2, 15, pp. 123-132, 1998.