SLIDING MODE CONTROL WITH RBF NEURAL NETWORK FOR TWO LINK ROBOT MANIPULATOR AND AN INVERTED PENDULUM SYSTEM

DISSERTATION
SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY
IN
CONTROL AND INSTRUMENTATION

Submitted by:

ANKITA YADAV

2K17/C&I/04

Under the supervision of

PROF. BHARAT BHUSHAN

DEPARTMENT OF ELECTRICAL ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

2019

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I, ANKITA YADAV, Roll No. 2K17/C&I/04 student of M.Tech (Control and Instrumentation), hereby declare that the thesis titled "SLIDING MODE CONTROL WITH RBF NEURAL NETWORK FOR TWO LINK ROBOT MANIPULATOR AND AN INVERTED PENDULUM SYSTEM" which is submitted by me to the Department of Electrical Engineering, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associate ship, Fellowship or other similar title or recognition.

Place: Delhi **ANKITA YADAV**

Date:

DEPARTMENT OF ELECTRICAL ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

CERTIFICATE

I, ANKITA YADAV, Roll No. 2K17/C&I/04 student of M. Tech. (CONTROL AND INSTRUMENTATION), hereby declare that the dissertation/project titled “SLIDING MODE CONTROL WITH RBF NEURAL NETWORK FOR TWO LINK ROBOT MANIPULATOR AND AN INVERTED PENDULUM SYSTEM” under the supervision of PROF. BHARAT BHUSHAN of Electrical Engineering Department Delhi Technological University in partial fulfillment of the requirement for the award of the degree of Master of Technology has not been submitted elsewhere for the award of any Degree.

Place: Delhi **ANKITA YADAV**

Date:

PROF. BHARAT BHUSHAN

SUPERVISOR

ACKNOWLEDGEMENT

I would like to express my gratitude towards all the people who have contributed their precious time and effort to help me without whom it would not have been possible for me to understand and complete the project.

I would like to thank Prof. Bharat Bhushan, Department of Electrical Engineering, my Project guide, support, motivation and encouragement throughout the period this work was carried out. His readiness for consultation at all times, his educative comments, his concern and assistance even with practical things have been invaluable.

Date: ANKITA YADAV

ABSTRACT

Nonlinear control techniques are applied on two different mechanical systems namely two link robot manipulator and an inverted pendulum system to study the effect of the controllers on the tracking performance of the two system. A design of sliding mode control(SMC) for the position tracking of two link robot manipulator based on the sliding mode control technique and the Lyapunov stability theory is carried out to eliminate the perturbation and asymptotical stability can be achieved when the system is subjected to the sliding mode. A sliding mode control method based on RBF(radial basis function) neural network is addressed which has the capability of learning uncertain control actions shown by the several industrial robots. In RBFNN-SMC method the algorithm for tuning the parameters are extracted from the RBF function. The comparative study is done based on the evaluated parameters for the system.

There are several practical applications like self-balancing robot, rocket propeller, Segway which are based on inverted pendulum. The control of inverted pendulum is carried out and a comparative analysis is made for the linear as well as the nonlinear model depending upon the two control techniques which are based on investigating the time, tracking error for obtaining the best performance for the inverted pendulum system. The implemented control techniques are the sliding mode control and the RBF neural network based adaptive sliding mode. Keeping the cart horizontal position and pendulum angle, to obtain the tracking performance the designed control law is subjected to different test signals and the results are shown for reduced chattering effect by using the above given controllers. A nonlinear SMC for position tracking control based on RBF neural network is also presented and the stability is given by the Lyapunov theorem.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATION

VSC	Variable Structure Control
SMC	Sliding Mode Control
MIMO	Multi Input Multi Output
IPS	Inverted Pendulum System
NLC	Non Linear Control
DOF	Degree of Freedom
RBF	Radial Basis Function
ANN	Artificial Neural Network
FBD	Free Body Diagram

CHAPTER 1
INTRODUCTION

1.1 NON LINEAR CONTROL SYSTEM

Non-linear control systems consist of at least one nonlinear component. All physical systems are inherently nonlinear in nature. The need to analyse the nonlinear systems arises from the fact that the principle of superposition on which linear control analysis is based, fails in the nonlinear case. The primary reason behind growing interest to analyse the nonlinear systems includes: improvement of linear control systems, analysis of non-linearities and need to deal with model designs.

Some common nonlinear system behaviour are listed below. It is important to know the main characteristics of nonlinear behaviour to just acknowledge whether these characteristics can be evaluated experimentally or in system simulations. The nonlinear systems are quite complex in nature. Since it does not follow the superposition principle, nonlinear systems reacts very differently to the external inputs and initial conditions from linear systems. The properties are given below:

(a) These systems oftenly have more than one equilibrium point. In linear system stability is evaluated by looking that for any initial condition, the response/motion of a stable system should always converges to the equilibrium point. On the other hand nonlinear system can converge to one of the equilibrium point by considering one set of equation and then goes to infinity with another set of initial conditions. This represents that the nonlinear system's stability depends on the initial condition but it also depends on the input value in the presence of a bounded external input.

(b) Without external input, these system display some oscillations of fixed amplitude and fixed period. Such oscillations are called as limit cycle.

(c) Nonlinear system may give a periodic output for a periodic input whose frequency can be harmonic or subharmonic of the input frequency. These nonlinear behaviour should be eliminated.

(d) Nonlinear systems can show the jump resonance in its frequency response which is a form of hysteresis.

1.2 COMMON NONLINEARITIES IN CONTROL SYSTEM

Some of the common nonlinearities are listed below:

(a) Saturation
(b) Deadzone
(c) Backlash
(d) On-off

(a) SATURATION

This is most commonly exist in amplifiers and actuators where the output changes linearly with the input but only valid for small range of inputs, if the amplitude of the input gets out of that range, the output shows the least variation and will settle around the maximum value.

(b) DEADZONE

In this nonlinearity, the system does not give response to the input until the input reaches a certain level and it also represents the situation in which output reaches zero after input exceeds a particular limiting level. The deadzone drastically affects the control system as it degrades the system performance as well as the accuracy of the system and this nonlinearity also destabilizes the system.

(c) BACKLASH

This finds its presence mainly in mechanical parts of control system. It possess multi valued nature. There can be two values present corresponding to each input. For example in gear trains, if the driving gear rotates at smaller angle than the gap, it can't make driver gear to move. Also if it moves in reverse direction , then also the driver gear remain stationary. When the contact establishes between the two then only it starts moving.

(d) ON-OFF NON-LINEARITY

It is an intentional nonlinearity and is also an extreme case of saturation nonlinearity and basically occurs when the linearity range converges to almost zero. This type of nonlinearity can lead to chattering phenomena.

1.3 ADVANTAGE OF STUDYING NON-LINEAR CONTROL THEORY

There are several reasons cited for studying the nonlinear control theory like we know that the linear system only works for very small range when one needs to operate in a wide range, linear controllers failed to perform because it is difficult to compensate the nonlinearities present. It also helps the designer to account for hard nonlinearities and uncertainties. There is also one more advantage to study such systems as this theory provides the design simplicity.

1.4 NON LINEAR SYSTEM ANALYSIS

It is difficult to analyse nonlinear systems because there is no universal technique that works for the analysis of all nonlinear systems. Since the direct solutions of nonlinear differential equations are generally difficult to find and frequency domain transformations do not apply so serious efforts have been made to develop appropriate tools for finding its solutions. There are many methods of nonlinear control system analysis as discussed below:

1. PHASE PLANE ANALYSIS

It is the method for studying the second order nonlinear systems. It solves the second order differential equation and display the result graphically as a family of motion trajectories on a two dimensional plan, called the phase plane, which allows to visually observe the motion patterns of the system. The disadvantage of this analysis is that it is only applicable to the systems which can be well approximated by a second order dynamics.

2. LYAPUNOV THEORY

This method is applicable to nonlinear control systems. The analysis of the stability of a nonlinear system using Lyapunov theory incorporates the idea to construct a scalar energy-like function(a Lyapunov function) and evaluates whether its decreases. The disadvantage of this method is to find the Lyapunov function for a given system. One of its application is the design of nonlinear controllers. It involves the formulation of a scalar positive definite function of the system state, and then choose a control law to make this function decrease. Thus the system designed by this will be guaranteed to be stable. This design approach has been used to solve complex design problems e.g., in adaptive control and in sliding mode control.

3. DESCRIBING FUNCTIONS

This method approximate the nonlinear components in nonlinear control system by linear equivalents and then use frequency-domain techniques to analyse the final systems. The accuracy of describing function analysis improves with the order of the system. Unlike Lyapunov technique in which trial and error search for Lyapunov function helps to analyse the stability, describing functions application is straightforward for a specific class of nonlinear system.

1.5 NONLINEAR CONTROL DESIGN

There is no general method for designing nonlinear controllers. There are several alternative and complimentary techniques, each applicable to particular classes of nonlinear control systems.

1. TRIAL AND ERROR

It uses the analysis tools to guide the search for the controllers. The phase plane method, describing function method, and the Lyapunov analysis can be used for this purpose. It fails for complex systems.

2. FEEDBACK LINEARIZATION

In this method, we transform a nonlinear system into linear system using feedback, and then use linear design techniques to complete the control design. It is applicable to many nonlinear systems.

3. VARIABLE STRUCTURE SLIDING MODE CONTROL

In model based nonlinear control, the control law is designed based on the nominal model of the physical system. In variable structure sliding mode control, the controller is designed based on both the nominal model and its uncertainties.

4. ADAPTIVE CONTROL

It deals with time varying systems. Adaptive control technique mainly applies to system with known dynamic structure but unknown constant. Adaptive controllers are inherently nonlinear in nature.

5. GAIN SCHEDULING

This involves application of linear control methodology to the control of nonlinear systems. Gain scheduling selects number of operating points and then at each operating point, a linear time invariant approximation to the plant dynamics is done. The parameters of the compensators are interpolated between these operating points and results in a global compensator.

6. INTELLIGENT CONTROL

Intelligent control takes advantage of computational structures like fuzzy systems and neural networks. Intelligent control is largely rule based because dependencies involved are complex in nature. Fuzzy system has the ability to quantify linguistic inputs and give approximation of complex system input output rules. Intelligent control incorporates the crucial aspects of biological intelligence. On the other hand, the neural network has the ability to learn from its data. Thus the intelligent control is a promising method for future control of many nonlinear systems.

1.6 NON LINEAR MECHANICAL SYSTEMS

1.6.1 ROBOT MANIPULATOR

A robot manipulator is an electronically controlled mechanism that performs tasks by interacting with its environment. They are also stated as robotic arms. An industrial robot consists of a robot manipulator, power supply, and controllers. They have sections like robot arm and body which have different functions. Robot manipulators are extensively used in industries. It involves dealing with the places and alignments of its numerous segments that make up the manipulators. Industrial robots perform several tasks like picking and dropping objects at desired locations and can imitate the human like tasks if programmed properly and do the work with the help of fully functional robotic arm. These robotic arms are called as the robotic manipulators. Manipulators are made up of an assembly of links and joints. Links are defined as the rigid units that make up the mechanism and joints are defined as the connection between two links as shown in figure 1.1.

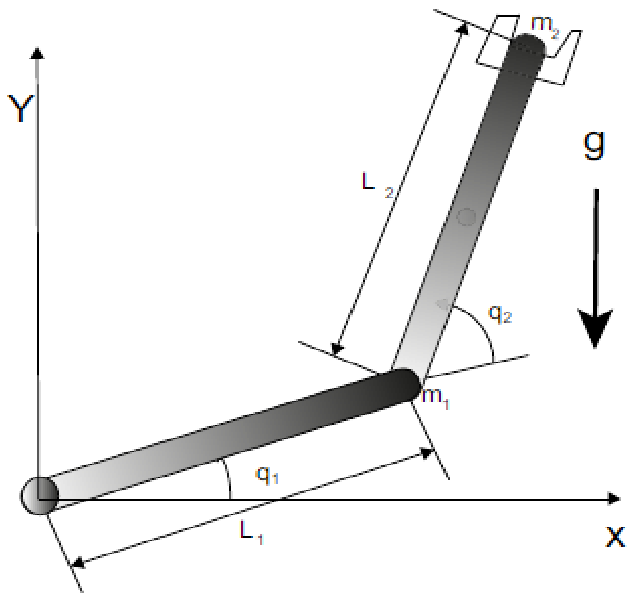


Figure 1.1 A two link robot manipulator

Robot manipulators are driven by electric, hydraulic, or pneumatic actuators, which apply torques at their joints. The dynamics of a robot manipulator describes the robot movement in response to the actuator forces. If we assume that the actuators do not have the dynamics of their own then we command arbitrary torques at the joints of the robot. This helps us to study the mechanics of robot manipulators without knowing about the details of how the joints are actuated on a specific robot. A set of nonlinear, second-order, ordinary differential equations gives the dynamic behaviour of robot which relies on the kinematic and inertial properties of the robot. These equations can be derived by summing all the forces acting on the robot. This technique has a great advantage of requiring only the kinetic and potential energies of the system to be calculated, and hence tends to be less susceptible to error than summing together the inertial, centrifugal, actuator, and other forces acting on the robot's links. It also allows us to determine and exploit the structural properties of the robot dynamics. Once the equations of motion for a manipulator are known, the inverse problem can be preserved: the control of a robot manipulator requires finding actuator forces which makes the manipulator to move along a specified trajectory. If we have a perfect model of the dynamics of the manipulator, we can find the appropriate joint torques directly from its model. Practically, we must strategize a feedback control law that can updates the applied forces in response to deviations from the chosen trajectory. To make the system to converge towards the desired trajectory under given initial condition error, modelling faults and sensor clatter There are ways to solve the problem of robot dynamics. The first method known as joint space control and it involves the technique to convert a given task into an anticipated path for the joints of the robot. A control law is used to evaluate the joint torques which makes the manipulator to trail the specified trajectory. A different method is to transform the dynamics into the task space and then the control law is written in relation with the end effector location and alignment. This approach is termed as the workspace control. If the robot is in touch with the surroundings then it is really difficult to determine its position, speed and orientations. So we have to control not only the location of the end-effector but also the forces it put on against the environment.

Industrial robots are programmable electronic machine. There main usage are in industries for manufacturing. These robots are automated and are highly capable of moving on three or more axis. It has few human-like attributes that look like the human bodily structure. The robots also retort to sensory signals as human responds to them. Anthropomorphic characteristics of robot manipulator like mechanical arms are used for numerous industrial tasks. Sensors permits the robots to communicate and interact with machines and to yield simple decisions. Some of the advantages of robots are given below:

(a)Robots are the best alternative to be used at hazardous places where human can't reach.

(b)The performance of robots is consistent and also it possesses repeatability which is difficult for human beings.

(c)We can reprogram the robots so that it can perform a different task when equipped with proper tools.

(d)The property of robots interconnection and wireless control technologies in robots increases the efficiency and productivity of robot industry.

1.6.1.1 ROBOT COMPOSITION AND RELATED ATTRIBUTES

Joint and link

An industrial robot consist of a series of joints and links. Robot anatomy is the study of diverse joints and links and various other features of the manipulator's physical buildings. A robotic joint provides the relative motion between two links of the robot. Each joint provides a given degree-of-freedom (dof) of motion. Mainly, only one degree-of-freedom is linked with each joint. Therefore the complexity of a robot is determined by the total number of degree of freedom it possesses. Each joint is associated to two links, an input link and an output link. Joint delivers controlled relative motion between the input link and output link. A robotic link is the rigid section of the robot manipulator. Most of the robots are straddling upon a motionless base, such as the floor. From this stationary base, a joint-link numbering scheme may be documented and is shown in Figure 1.2. The robotic base and its joining with the first joint are termed as link-0. The very first joint in the sequence is joint-1. Link-0 will be the input link for joint-1, while the output link from joint-1 will be link-1—which tends to joint-2. hence link 1 eventually becomes the output link for joint-1 and the input link for joint-2. The above given joint-link-numbering scheme is further trailed for all joints and links in the robotic systems. The joints in the robots are really helpful in the robot movement and basically provides a relative motion between the connected two parts. This schemes gives the assembling of the outer components of the robot like arm, wrist and the body of the robot. The schematic diagram of the given joint link scheme is given in the figure 1.2 which clearly shows the interconnections of one link to the another.

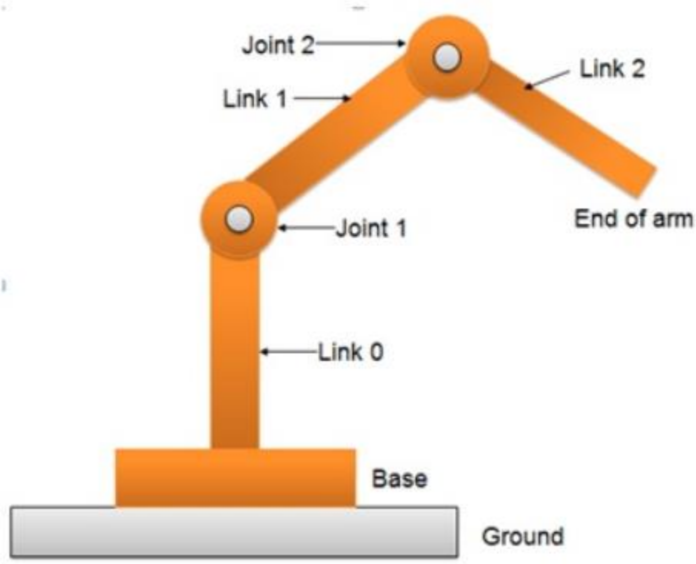
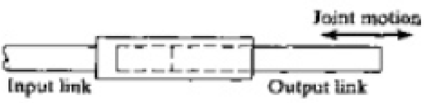
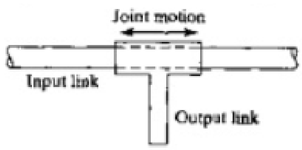


Figure 1.2 Joint-link scheme for robot manipulator

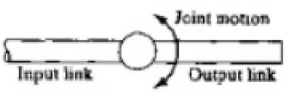
All industrial machine robots have numerous mechanical joints that can be classified into following types:



(a) Linear joint

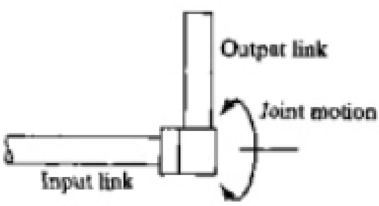


(b) Orthogonal joint



(c) Rotational joint

(d) Twisting joint



(e) Revolving joint

Figure 1.3 Types of joints

Robotic joints are classified into five types given in the figure 1.3

(a) Linear joint (type L joint)

The relative movement between the input and the output link will be translational motion and the axes of both the links should be parallel.

(b) Orthogonal joint (type U joint)

This also depicts a translational sliding motion, but the input and output links lie perpendicular to each other during the movement.

(c) Rotational joint (type R joint)

This provides relative rotational motion, with the rotational axes be perpendicular to the axes of the input link and output links.

(d) Twisting joint (type T joint)

This joint also includes rotational motion, but the axis of rotation is parallel to the axes of the both links.

(e) Revolving joint (type V-joint)

In this type, input link axis is parallel to the rotational axis of the joint. And the output link axis is perpendicular to the axis of rotation.

1.6.1.2 DRIVE SYSTEMS

Basically there are three types of drive systems that are usually used to actuate robotic joints which are electric, hydraulic, and pneumatic drives. Electric motors act as the prime movers in robots. Servo-motors or steeper motors are extensively used in robotics. Hydraulic and pneumatic drive systems like piston-cylinder, rotary vane actuators are castoff to achieve linear motions, and rotary motions of the joints respectively.

Pneumatic drive is frequently used for smaller, simpler robotic application while electric and hydraulic drives have applications for more learned industrial robots. Electric drives are usually preferred in commercial applications nowadays due to the advancement in technologies. They are also having compatibility with computing systems. Hydraulic systems apart from having less flexibility as compared with electrical drives, are commonly used when larger speeds are desired. These system are generally working to carry out heavy duty operations using robots.

The combination of this drive system, sensors used, and feedback control system governs the dynamic response characteristics of the robot manipulator. In robots, speed usually referred to as the absolute velocity of the robot manipulator at their arm. The system can be programmed into different sections so that different portions of this work cycle are carried out at different velocities. Acceleration and deceleration controls are also crucial factors, particularly in a confined work space. Manipulator's capabilities of switching control between the velocities is beneficial. Other key factors involves the weight (mass) of the article being manipulated, and the precision that is essential to locate and position the article suitably. All of these factors are collected under the concept 'speed of response', which is well-defined as the time required for the robot manipulator to move from one point in space to the next one. Production rate can be altered by this phenomenon of speed of response as this affects the robot's work cycle time. The amount of overshoot and the oscillations happening in the robot motion at their arms gives the stability of the system to move towards the next programmed location. If the rate of oscillations are more, it will lead to greater instability and hence lead to greater response time for a robotic system.

Another important factor is load sharing capacity which can be evaluated by weight of the gripper used to hold the items. A heavy gripper sets a greater load on the robotic manipulator along with the object mass. Commercial robots have the capacity to carry loads of up to 1000 kg, while medium-sized industrial robots can lift up to 45kg.

1.6.1.3 DEGREE OF FREEDOM

The number of degree of freedom is the number of independent variables required to completely identify its configuration in space.

The number of degree of freedom is calculated as:

$$n = a(n-1) - \sum_{i=1}^{k}(a-f)$$

where n is the number of links, k is the number of joints, f is the number of degree of freedom of ith joint and a is 3 for planar mechanism and 6 for spatial mechanism.

1.6.1.4 ROBOT CONTROL SYSTEM

Robotic system are extensively used nowadays in smart manufacturing industries. Earlier, there was a stiff mechanism of controlling the robots joint by a SISO system. The control of manipulators can be classified as follows:

1. Linear control
2. Feedback or closed loop control

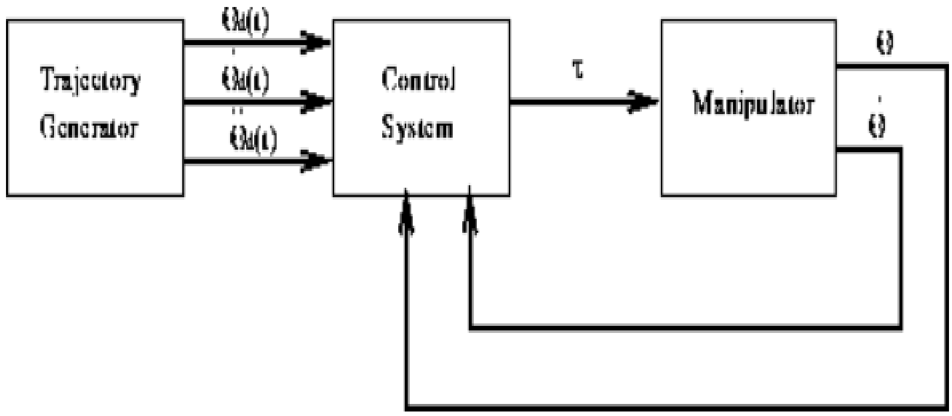


Figure 1.4 Block diagram of robot control system

The figure 1.4 shows the block diagram for the robot control system. In linear control of manipulator, the initial and the final location of the end effector will serve as the input which one can determine through the transformation matrices. In joint space, inverse kinematics computes the desired location. The control system commands the actuator based on the servo error produced by the reference value and the sensor measurements. Linear techniques can only be useful if the system can be easily modelled by the differential equations mathematically. These methods are the approximate methods and hence the use of such linear controllers are not reasonable.

While in feedback or closed loop control system, the servo error is evaluated between the actual and required locations and velocities respectively. The torque is then computed to reduce this error. There are few requirements before implementing this technique such as system should be stable and also the performance should be satisfactory.

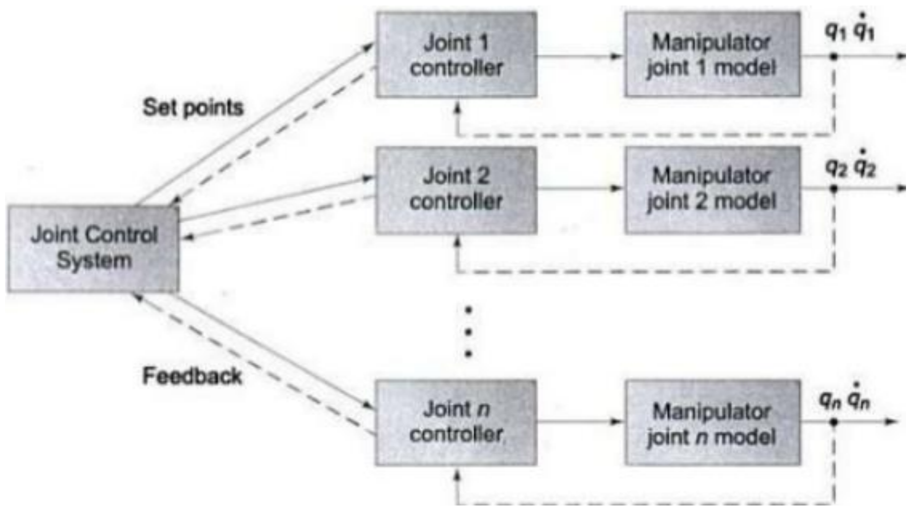


Figure 1.5 Robot control architecture

The above figure 1.5 shows the robot control architecture which consist of controllers for each joint as well as the manipulator joints model. The manipulator problem is multi input multi output problem with each joint controlled independently. Nonlinear control strategies are hence used to overcome the model uncertainties.

1.6.2 INVERTED PENDULUM

A simple pendulum having a bob which is hanging from a rigid support called as pivot is stable if it is motionless and is said to be at equilibrium position and hence there will be no torque applied on the system but if it is disturbed from its original position and given a drift, a restoring torque start applying at it which tend to make it to come to its equilibrium position. But if we consider a pendulum in inverted position supported on a rigid surface like a wagon/cart directly above it at an angle of 180 degree from its stable position is considered to be as a unstable point. There is no torque initially but if it is displaced a bit from this position, a gravitational torque start acting that will accelerate the system away from its equilibrium position and the pendulum can easily fall over. A feedback control system is required to stabilize it which records its angle and the drifts the position of the point sideways to make it stable.

An inverted pendulum can move only in one dimension. Inverted pendulum is inherently an unstable system. To keep this system intact and stable, force should be applied. Proper theory is needed to achieve this. It is one of the most difficult system to control considering the subject of control engineering. Inverted pendulum is selected as the system because of few reasons which are: the most common and available system for laboratory usage and also a nonlinear system, which can be considered as a linear system, without considerable error, for a wide range of variation. For controlling an inverted pendulum system, there are several tasks to be accomplished which are divided into stages like the inverted pendulum must be modelled first and then operations are carried out for the wide range. Inverted pendulum closed loop response is uncompensated and hence analysed with the help of root locus plot. A controller is designed for it and then it has to be simulated in MATLAB as done in this project by applying two techniques for proper tuning and verification work. The controller hence designed is applied on the physical model of inverted pendulum.

An inverted pendulum is an unstable system with only one input signal and many output signals so we need a feedback for stabilizing it. The figure 1.6 shows an unstable inverted pendulum model in which the cart is desired to move in the x direction without the pendulum mounted on it falling. A dc motor is used to move the wagon, also the position of the cart and the angle of the pendulum is given as feedback parameters to the control system.

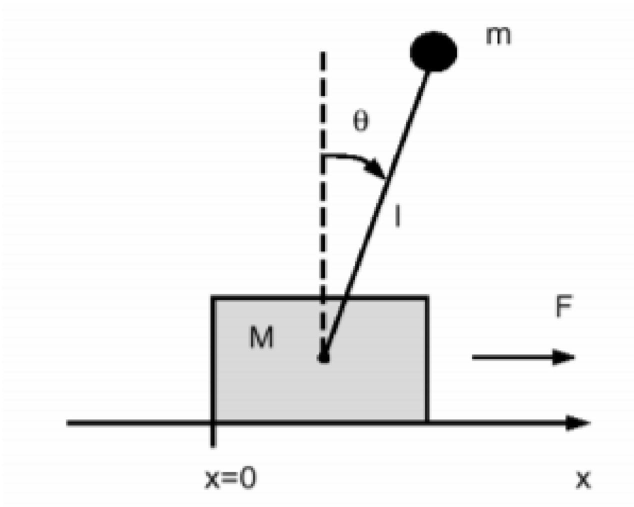


Figure 1.6 An inverted pendulum mounted on a cart

Figure 1.6 shows an inverted pendulum model mounted on a cart. A mathematical model of the system has been developed, giving the angle of the pendulum resulting from a force applied to the base. Setup Description The inverted pendulum is mounted on a moving cart. A servomotor is controlling the translation motion of the cart, through a belt/pulley mechanism. That is, the cart is coupled with a servo dc-motor through pulley and belt mechanism. The motor is derived by servo electronics, which also contains controller-circuits. A rotary-potentiometer is used to feedback the angular motion of the pendulum to servo electronics to generate actuating-signal. Controller circuits process the error signal, which then drives the cart through the servomotor and driving pulley/belt mechanism. To-or/and-fro motion of the cart applies moments on the inverted pendulum and thus it keeps the pendulum upright.

For designing this nonlinear mechanical system, we need majorly four components: (1) the cart, (2) the pendulum (3) the track, and (4) the procedure to move that cart. The above given should meet certain requirements which are given as follows:

(a)The motion of the card must be limited to one DOF which lies in the horizontal plane.

(b)The motion of the inverted pendulum needs to be limited to only 1 DOF among which one should match with one of the DOF of the cart.

(c)The friction that inhibits the motion between the cart and the pendulum should be reduced as much as possible.

1.7 MOTIVATION

There are discrepancies between the actual system and the model designed for the controller in the formulation of almost every control problem. This may be due to the variations in system’s parameters, unmodelled system dynamics or due to complex plant behaviour approximation by a simple model. So it is important to look after the real-world applicability of the designed controller so as to maintain the desired performance level. This has led my interest in developing the robust control techniques to solve the nonlinear control problems like two link robot manipulator and an inverted pendulum system. Sliding mode control method is one of the best to solve such problems. There is also a need for systematic design methodologies for stable control system using the neural network and a appropriate number of NN function can model any nonlinear function and reduce the modelling error and try to make it small. Discontinuous nonlinearities can also be modelled by using neural technique.

The sliding mode technique provides the better performance as it can be applicable to wide range of systems and is also insensitive to parameter variation so we make use of this technique to linearize two nonlinear systems and also applied neural network compensation technique as it can reduce the modelling error.

1.8 OBJECTIVE OF THESIS

(a)To give the basic idea about modelling of two nonlinear mechanical system-a two link robot manipulator and an inverted pendulum system.

(b)To understand mathematical modelling and state space representation for the manipulator and inverted system.

(c)To study the sliding mode controller and also analysis the stability of the nonlinear system through Lyapunov theory.

(d)To use two different techniques to evaluate and also improve the system’s performance which are sliding mode control technique and use of RBF as compensation technique.

(e)To give the best method for eliminating the chattering phenomena which occurs in nonlinear mechanical system by the use of appropriate controllers.

(f)To provide advanced sliding mode control design methods for two link robot and an inverted pendulum system.
(g)To provide neural network control design methods for implementing the structure of robot manipulator and inverted pendulum.
(h)To provide advanced controller design method using NN technique and deduce their stability.

1.9 LITERATURE REVIEW

Sliding mode control has been the vast topic for research in the field of control engineering. This section discuss some previous research work and development of controllers for the two link robot manipulator and an inverted pendulum. Extensive research work has been done on sliding mode control of the system as well as the integration of compensation technique with this control methodology.[1] It is shown that the adaptive sliding mode control to control a 2 link robot manipulator having uncertain parameters. An adaptive algorithm is used which is based on the sliding mode control concept to ease the chattering phenomena. Lyapunov stability is applied to achieve the stability of robot manipulator system. Controller scheme proposed in that assures robustness and attain good trajectory tracking performance.[2] A sliding mode control for a MIMO system has been given which has major uncertainties and also strong coupling features. Lagrange equations developed the mathematical model for the inverted pendulum system in this paper and the speed of the pendulum bar is estimated through the state observation and the estimation of the speed of wagon. The stability here also is determined by the Lyapunov theory. [3] The dynamic model and the terminal sliding mode control for two link robot manipulators. Lagrangian method is used to derive the closed loop dynamic equations for the robot manipulators. This technique has been used to design the controller. This technique gives the output tracking error to converge at zero in a given period of time. [4] A radial basis function network with SMC is designed for the link positions of two link robot manipulator. The learning techniques has degrades its transient performance also This neural technique uses the curve fitting mode to get nonlinear mapping. A sliding mode effectively used to eliminate the nonlinearities and has given a fast response but this control created chattering. The Lyapunov and the back propagation algorithm creates switch gain and an update law respectively.[5] The neuro SMC is designed for an inverted pendulum based on the RBF technique and stability is given by the Lyapunov theory .[6] A second order sliding mode control is given for the model of inverted pendulum and second order algorithms are applied on two systems for ensuring the robustness against variation in parameters, errors in modelling the system and external disturbances by eliminating the chattering phenomenon. This piece also deals with discrete system, the first controller controls the nonlinear single input single output systems and the second discrete controller eliminate the chattering effect. Also the controller are unaffected by the disturbances, hence this approach is applied on inverted pendulum.[7] When the input is unknown, a neural network is used for decoupled sliding mode control. The whole model of inverted pendulum is divided into two subsystem which is second order type and two sliding mode will be designed in which the first one includes a variable related to second one and then a terminal sliding mode is incorporated in both system . Finally, for both the system, a terminal mode sliding mode control is applied.[8] The designing of balance control for an inverted pendulum system which consist of a wagon, linear motor and a pole. The wagon on which it is mounted is driven by linear motor. The objective of the paper was to balance it upright on the wagon for which second order sliding mode control is used to design a control law. Also, LQR is used for comparison and first order sliding mode is there for stabilising the control law for the given system.

[23]Computational abilities of networks are made of models of neurons and was developed in the 1940s, neural network techniques saw great improvements and developments. [25] Neural Networks are being applied in many fields such as learning, pattern recognition, signal processing and system control. The major advantages this parallel structure, learning ability, non- linear function approximation, fault tolerance. This benefits make a wide range of applications. [26]] The major applications falls into three major categories Pattern classification, Prediction and financial analysis, and Control and Optimisation. [27]Real-world applications have many nonlinearities, un-modeled dynamics, unmeasurable noise, multi-loop, etc., which creates problems to implement control strategies. [28] Past decades have seen development of new control strategies and has been based on modern and classical control theories. Control theory nowadays have shifted from classical control theory.[29] Conventional design methods for controllers for a MIMO plant like a multi-joint robot require minimum knowledge of the structure and accurate mathematical model of the plant. [20] The values of many parameters of the model also need to be precisely known to accurate implementation.[30] Neural networks can learn the forward and inverse dynamic behaviors of complex plants online, offer alternative methods of realizing MIMO controllers capable of adapting to

environmental changes. [31] The design of a neural network-based control system does not require any prior knowledge about the plant. The concept of back-propagation (BP) neural network and RBF neural network have boosted the development of neural control [32] For example, there are many neural control approaches have been developed with BP neural network. The study of adaptive nonlinear control systems with RBF universal function approximation has been widely used.[33] The RBF network adaptation improves the control performance against large uncertainty of the system. The adaptation law can be derived by Lyapunov method which guarantees the stability of the system. RBF neural networks were recognized in 1988 [35], and have drawn much attention due to their optimum generalization ability and simple network structure that avoids unnecessary and lengthy calculation. [36] . Research from the past shows that universal approximation theorems on RBF has shown that any nonlinear function over a compact set with arbitrary accuracy can be approximated by RBF neural network.[38] RBF neural network has three layers: the input layer, the hidden layer, and the output layer. [39] Neurones forms the hidden layer and are activated by a radial basis function. [40]The control of multi-input multi-output (MIMO) plant is a peculiar problem when the plant is nonlinear and time varying . [41]MIMO plant posses dynamic interactions between the plant variables. A robot manipulator is an example of one such system.[42]Robot manipulators are very important in the field of flexible automation as they have been widely used for increasing productivity and reduce losses. A lot of research effort is been made to design their controller. [43]To achieve desired trajectory tracking and good control performance, a number of control schemes have been developed. [44]Computed torque control is one of such schemes, which is based on the exact cancellation of the nonlinear dynamics of the manipulator system. [45]The payload of the robot manipulator varies during its operation and is not known advance. To face such problems, adaptive control strategies for robot manipulators have been developed.

Now, we are going to implement adaptive sliding mode control on two underactuated mechanical systems. RBF neural network with adaptive sliding mode control technique has also been applied on two link robot manipulator and one link inverted pendulum system to improve their tracking performances as well as these techniques will help to reduce model uncertainties and external disturbances.

CHAPTER 2

SLIDING MODE CONTROL AND NEURAL NETWORK ANALYSIS

2.1 SLIDING MODE CONTROL

Sliding mode control deals with the problem of model uncertainties as these uncertainties can have strong adverse effects on nonlinear control systems. A major approach to deal with the model uncertainly is adaptive control and another approaches which can be used to solve the control problems includes the sliding mode techniques. These techniques are generating greater interest nowadays. Discrepancies may occur between the actual plant and the mathematical model established for the controller design. Various factors may be responsible for this mismatch. The engineer's role is to ensure required performance levels for the system instead of such mismatches. A set of robust control methods have been developed to eliminate any error. One such method is sliding mode control methodology(SMC). This is a specific type of variable structure control system (VSCS).

SMC has been used for several systems including nonlinear system, multi-input multi-output(MIMO) systems, discrete-time models, large-scale and infinite-dimension systems, and stochastic systems. SMC is completely insensitive to parametric imprecisions and external disturbances during sliding mode. VSC uses a high-speed switching control law to accomplish two objectives. Firstly, the nonlinear plant's state trajectory is taken onto a specified and user-chosen surface in the state space which is called the sliding or switching surface. This is called as the switching surface because a control path has a unique gain if the state trajectory of the plant is "above" the surface and a different gain if the trajectory falls "below" the surface. Secondly, it keeps the plant's state trajectory on this surface for all consequent times. During this process, the control system's structure changes from one to another and thus given the name variable structure control(VSC). The control is also named as the sliding mode control to accentuate the importance of the sliding mode. Sliding mode controller can stabilize the trajectory of a system.

In sliding mode control technique, the system is designed to drive and then compel the system state to stay within a neighbourhood of the switching function. Its main advantages includes (1) the dynamic behaviour of the system may be well fitted by the specific choice of switching function, and (2) the closed-loop response becomes totally impervious to a specific class of uncertainty. Also, the capability to stipulate the performance directly makes sliding mode control attractive from the design viewpoint. System's trajectory can be stabilized by a sliding mode controller. The system states "slides" along the line s=0 after the initial reaching phase. Particular s=0 surface is selected because it has desirable reduced-order dynamics.

Now since,

$$s=cx_1+\dot{x}_1,\ c>0 \tag{2.1}$$

the above surface corresponds to the first order linear time invariant system given by:

$$\dot{x}_1 = -cx_1 \tag{2.2}$$

In practice there will be high frequency switching and this high frequency motion is called as the chattering effect which means that the states are continuously crosses the surface and ideally no sliding mode occurs. This effect can be reduced by using various techniques like nonlinear gain, dynamic extensions, higher order sliding mode control.

Sliding mode control is a nonlinear control method which changes the dynamics of a nonlinear system by the multiple control structures that are designed so as to guarantee that trajectories always move towards a switching condition. Therefore, the final trajectory will not exist entirely within one control structure. The state-feedback control law is not a continuous function of time. Instead, it switches from one continuous structure to another based on the current position in the state space. Hence, sliding mode control is a variable structure control method(VSC).

Control structures are designed so as to ensure that trajectories will always move towards a switching condition. Therefore, the ultimate trajectory will not exist completely within one control structure. Instead, the ultimate trajectory will slide along the boundaries of the control structures. The motion of the system as it slides along these boundaries is termed as a sliding mode and the geometrical locus consisting of the boundaries is called the sliding (hyper) surface.

To force the trajectories of a dynamic system to slide along the constrained sliding mode subspace the sliding mode control technique uses practically infinite gain. Control trajectories which are obtained from this reduced-order sliding mode have some desirable properties (e.g., the system naturally slides along it until it comes to rest at a desired equilibrium). The main strength of sliding mode control is its robustness. It need not be precise and will not be sensitive to parameter imprecisions because the control can be as simple as a switching between two states. Since the control law is not a continuous function, the sliding mode can be reached in finite time (i.e., better than asymptotic behaviour).

There are two steps in the SMC design. The first step is to design a sliding surface so that the plant restricted to the sliding surface has a desired system response. The second step is constructing a switched feedback gains necessary to drive the plant's state trajectory to the sliding surface. These constructions are built on the generalized Lyapunov stability theory.

2.2 SLIDING MODE CONTROL ALGORITHM

Our main objective is to regulate a dynamic system which is subjected to the plant parametric uncertainties and nonlinearities. A controller helps a system to reach and subsequently remain on a predefined surface called the sliding surface within the state space. The motion of the system while it is confined to the surface is called the sliding motion. The benefit of obtaining this motion is that there will be reduction in the order of the system and also the sliding motion is not affected by the variations in the parameter of plant. The latter property makes this an attractive methodology for designing the robust control for several uncertain systems.

The design approach consist of mainly two steps:

(a) Designing of sliding surface in the state space so that the reduced order sliding motion satisfies the specifications applied on the system.

(b) Synthesis of control law such that the trajectories of closed-loop motion are directed towards the sliding surface.

A variable structure control law used for the closed loop dynamical behaviour obtained has two distinct types of motion. The initial phase is termed as the reaching phase which occurs while the states are driven towards the sliding surface. This type of motion is affected by the external perturbations. This is eliminated only when the states reaches the surface and become insensitive to the uncertainties. If in the neighbourhood of sliding surface, the state velocity vectors are directed towards the surface then there exist a sliding surface. The sliding mode can be best understood by the figure (2.1) which shows a sliding surface, reaching phase and the sliding phase.

Sliding surface is one where the trajectory reaches and stays for the remaining period of time.

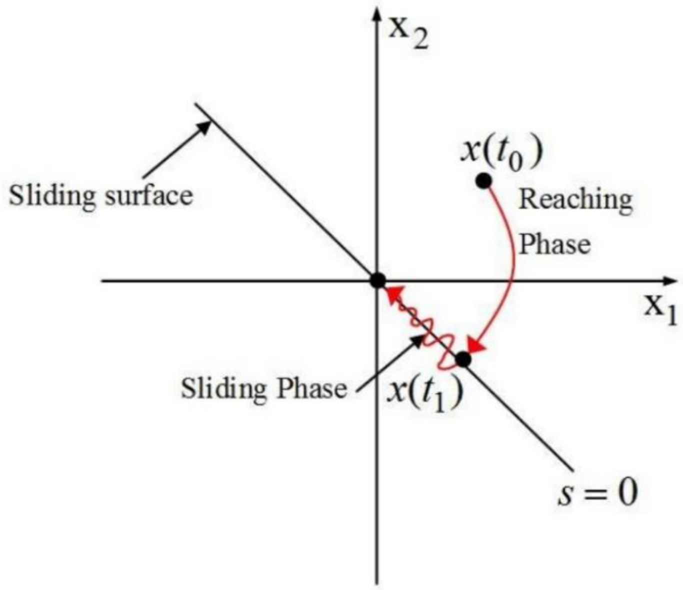


Figure 2.1 Sliding mode phases

A hyper surface

S:$\sigma(x_1, x_2, \dots \dots, x_n)$=$\sigma(x) = 0$

is attractive if

(a) Any trajectory starting on the surface stays there only

(b) and any trajectory starting from outside the surface directs at it asymptotically.

Reachability condition that ensure the motion state trajectory x(t) of single input system is given below:

$$\dot{x} = f(x, u, t) \qquad (2.3)$$

Now we know $\sigma = 0$ on either side of sliding surface so we have

$$\lim_{\sigma \to 0^+} \dot{\sigma} < 0; \lim_{\sigma \to 0^-} \dot{\sigma} > 0 \qquad (2.4)$$

The above two equations combined together to give

$$\sigma\dot{\sigma} < 0 \qquad (2.5)$$

in the neighbourhood of sliding surface i.e.

$$\lim_{\sigma \to 0} \sigma \frac{d\sigma}{dt} < 0 \qquad (2.6)$$

The trajectory will remain on the hyper surface S for all the time so we require

$$\frac{d\sigma}{dt} = \left(\frac{d\sigma}{dx}\right)^T \dot{x} = 0 \qquad (2.7)$$

So if the reachability condition gets satisfied globally and since

$$\frac{1}{2}\frac{d}{dt}\sigma^2 = \sigma\dot{\sigma} < 0 \qquad (2.8)$$

which gives

$$V(\sigma)=\frac{1}{2}\sigma^2 \tag{2.9}$$

This is the Lyapunov function for $\sigma(t)$.

2.3 LYAPUNOV FUNCTIONS FOR NONLINEAR SYSTEMS

Lyapunov's direct method centres around the choice of a positive definite function V(x), called the Lyapunov function for the determination of the stability. There is no universal method for selecting the unique Lyapunov function for a specific nonlinear system. Some Lyapunov functions provides a better result than the other ones. For the systematic construction of the Lyapunov functions, there exists several techniques and each one of them is applicable to a particular class of systems. If we are unable to find out the Lyapunov function then it does not mean that the system is unstable. Also, if a certain Lyapunov function gives stability for a specified parameter region then it does not necessarily means that if it leaves that region then it will result in system instability. There is no general method which will tell us whether the given V-function is positive definite. Despite of all its shortcoming, it is the most powerful method to determine the stability of the nonlinear system. There are several methods for building the Lyapunov functions like krasovskiis method and the variable gradient method.

2.4 LYAPUNOV STABILITY

The Lyapunov stability analysis is based upon the concept of energy and also depends on the relation of stored energy with the stability of system. But in general there is no way of relating the energy function with the given set of equations which describes the system. There is no uniqueness about the total energy of the system which associates with the system stability. There are scalar functions present which describe this relation and this ides is given by the mathematician A.M Lyapunov. The scalar functions are known as the Lyapunov Function and the method used for evaluating the stability of the control system is known as the Lyapunov's direct method. Following are some of the scalar functions which are described as the follows by the Lyapunov direct method. Based on these functions, stability can be evaluated.

(a)Positive Definiteness of Scalar Functions :V(x) which is a scalar function is said to be positive definite in the region $||x||\leq k$ if $V(x)>0$ at all the points of the region except origin since it is zero at origin.

(b)Negative Definiteness of Scalar Functions: V(x) which is a scalar function is said to be negative definite if [-V(x)] is positive definite.

(c)Positive Semi definiteness of Scalar Functions: V(x) which is a scalar function is said to be positive semidefinite in the region $\|x\| \leq k$ if its value is positive at all points of the region and at origin too.

(d)Negative Semi definiteness of Scalar Functions: V(x) which is a scalar function is said to be negative definite if [-V(x)] is positive semidefinite.

2.5 INTELLIGENT CONTROL

There are machines having human like capabilities that can perform human functions with better efficiency like robots are used in manufacturing, mining, space and agriculture etc. In recent times, intelligent systems that matches human intelligence is the biggest topic of research. Human mind comprises of many robust attributes with complex sensory, emotional and complex thought processes. Hundred billions of biological neurons in our central nervous system plays an important role in this. The central nervous system takes information from the external world through our sensory organs and accumulate all the information and deduce through cognitive computing. But these processes are extremely complex in nature. So the subject of intelligent system provides an amazing platform to incorporate these information and provides the best mathematical solution for the emulation of higher order cognitive functions.

2.6 SOFT COMPUTING TECHNIQUES

The most commonly used method for control design is based on the mathematical models which are derived from physical laws. It provides a set of equations that represents the interrelation between the quantities and their relation with the external inputs. There are several equations like algebraic, differential, difference, integral etc. When these equations can be solved easily within a certain time limit at a reasonable cost and with a certain accuracy then there is no need for any alternative technique. But if any one of the condition fails, we need other methods to solve the control problems. The principle of soft computing techniques based on the idea of exploiting the tolerance for imprecision, uncertainty to achieve the robustness. There are many control problems like recognition, computer graphics, robot coordinates etc that does not uses the classical hard computing since they don't not need precise solutions. The core methodologies of soft computing are: fuzzy logic, neural network etc. Neural network has no need for the availability of human solutions but the network can be trained through exemplification. It involves the use of an algorithm for obtaining best solution in a huge solution space. No single technique is enough for computing the best results i.e. they work in synergy to have the best possible solution for a control problem. There are terms used for intelligent system like machine learning which is the key to machine intelligence. For multi input multi output controllers, the neural network having the knowledge of the dynamic behaviour of complex plants provides an approach for realizing this system which is capable of adapting environmental changes.

2.6.1 NEURAL NETWORK

There are several nonlinearities like noise effect, unmodeled dynamics, multi-loops etc. the classical control theories and the modern control theories such as adaptive control and optimal control techniques are mainly based on the system linearization. For all these, mathematical modelling of system is a necessity. The capability of neural network is that it can be trained to learn any function so it eliminates the need for evaluating the complex mathematical analysis. Vast parallelism offered by the neural network offers very fast multiprocessing technique. Due to this architecture of neural network, the overall performance does not get affected if some parts of the neural network get damaged.

Warren McCulloch and Walter suggested a simple neuron model known as the artificial neural model. The artificial neural network has few major applications like function approximation. These network can generate the input output maps which can approximate any continuous function with the desired degree of accuracy. The results are based on firm analytical foundations when the neural network are used for the control system. Based on the analogy of biological neural network, an efficient computing method is devised called as the artificial neural network (ANN) which comprises of a large collection of units that are interconnected in some patterns to allow communications between these units. These units are called as neurons. Artificial neuron evaluates the input signals and compare the total to some threshold level and finally determines the output as shown in figure 2.2. The inputs are given at the input nodes and weights are assigned after which an activation function works on that to produce the desired output.

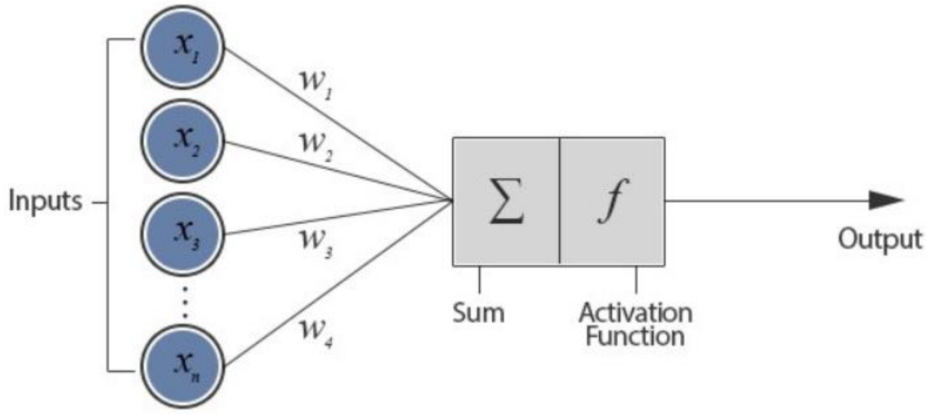


Figure 2.2 Artificial neuron

Each input will be given a relative weighting that affects the impact of that input. Weights are the adaptive coefficient that determines the intensity of the input and it provides neural network to learn and store information. Artificial neurons use an activation function also called as the transfer function. After building a neuron model, when the stimulus surpasses a certain threshold value θ that the transfer function possesses, a constant output is produced. The threshold value is subtracted from the weighted sum and the resulting value is compared to zero; if the sum is positive, then the output is one, else output is zero. An offset is added to the weighted sum by the threshold effectively. If the threshold is taken out of the body and connect to the extra input value , we can get the same effect. In this the extra input is multiplied with the weight and added in a manner similar to other inputs. This is known as biasing the neuron.

2.6.1.1 NETWORK ARCHITECTURE

The classification of the organization of neural network can be done in two types: a feedforward net and a recurrent net. The feedforward has a hierarchical structure that comprises of several layers and signal flows in one direction and in recurrent net, multiple neuron in a layer are interconnected.

The feedforward network feeds the input patterns to a layer through a set of input terminals. Each layer performs different computations based on the weighted sum of its input and pass on the result to the next layer and finally one or more neuron evaluates the output. The last layer of the network is known as the output layer. The layers which exist between the input and the output layer are called as the hidden layer. Neural networks with multiple layer are called as multilayer perceptron. There is no involvement of the initial and past states of neurons in this type network and hence limiting its use as no dynamics are involved. Recurrent neural network uses feedback from the output neuron to the input terminals. This network has a rich range of architectural layouts. Figure 2.3 shows deep neural network architecture.

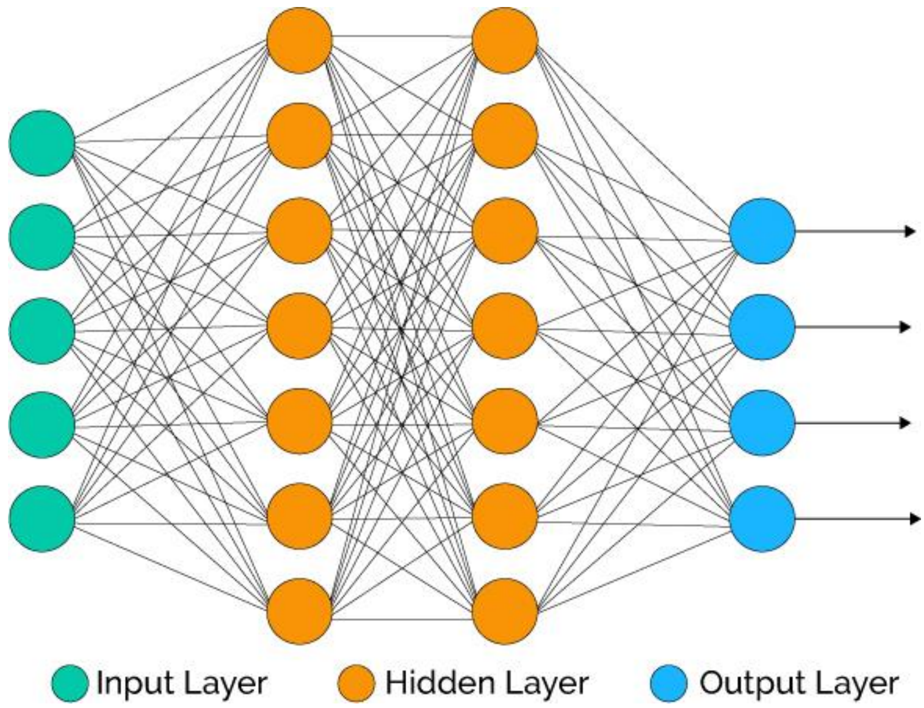


Figure 2.3 Deep neural network architecture

We can see three layers in figure 2.3 in which layer 1 represents the input layer at which any input data can be given to the network. The second layer i.e. the layer 2 is called the hidden layer because of the fact that this layer is not a part of input as well as output. Although neural network can have many hidden layers but we showed three layered network here. Each layer 1 node is connected to some node of layer 2. And then nodes of layer 2 is connected to the output. Each connection is having a certain weight.

2.6.1.2 TRAINING OF NEURAL NETWORK

Starting with initializing the weight of the neural network randomly. This does not give the accurate results so we need to train the neural network data. After changing the functions by adjusting weights, it can be possible to improve the network. There are several algorithms present that optimize he function. Those algorithms are based on gradient.

2.7 RADIAL BASIS FUNCTION NETWORKS

This is an alternative approach for multilayer perceptron which is trained by the backpropagation algorithm. It is the basic component of feedforward network. A gaussian function has two parameters: center for defining its position and for determining its shape, parameter called spread is there. In one dimensional gaussian function spread is same as the standard deviation σ.

The input vector is given by

$$x = [x_1 \quad x_2 \ldots \ldots \ldots . . x_n] \tag{2.10}$$

and the output of the RBF network is given below

$$x(x, c, \sigma)=\exp\left(\frac{-||x-c||^2}{2\sigma^2}\right) \tag{2.11}$$

there is no weights between the input and the RBF unit. c and σ represents the weight which is shown in figure 2.4.

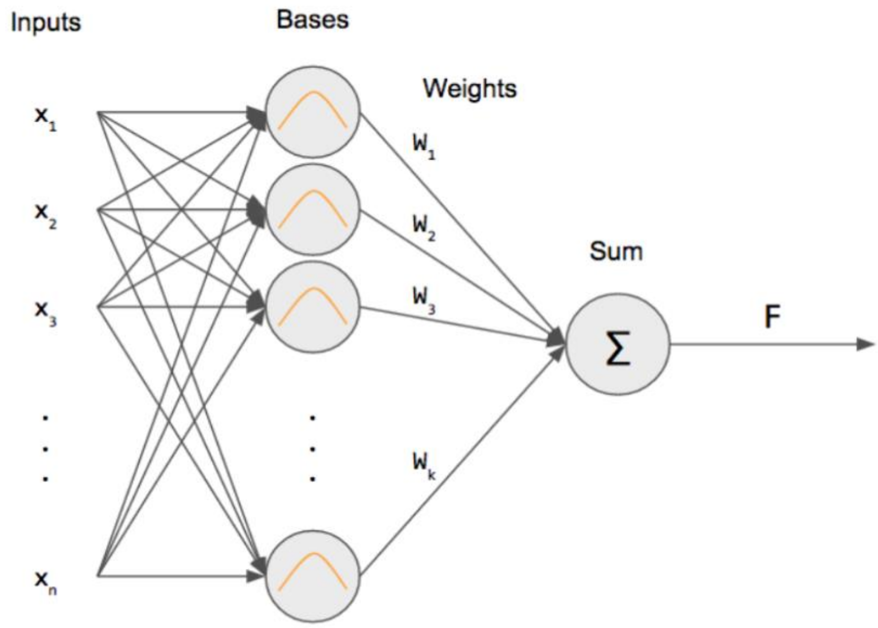


Figure 2.4 Gaussian function in RBF neuron model

2.7.1 TRAINING OF RBF NETWORK

The mapping properties of the RBF network is governed by the set of parameters given by the weight w_{jl}, j=1,2,…..,q; l=1,2,….,m, in the output layer

By selecting an appropriate cost function

$$E=\frac{1}{2}\sum_j^q \left(y_j - \hat{y}_j\right)^2 \tag{2.12}$$

where

$$\hat{y}_j = \sum_{l=1}^{m} w_{jl}\, \varphi_l(x, c_l, \sigma_l)$$

$\{c_l . \sigma_l\}$ are the parameters of the radial basis functions.

Radial basis function is a special class of function whose characteristic shows that the distance can monotonically increase and decrease from a mid-point. The centre , the shape of the graph and the distance are all the parameters of RBF.

A typical RBF function can be represented as:

$$h(x) = \exp\left(-\frac{(x-c)^2}{r^2}\right) \quad (2.13)$$

the parameters are the centre and the radius. Figure 2.4 shows a RBF function having c=0 and r=1. This is shown in figure 2.5.

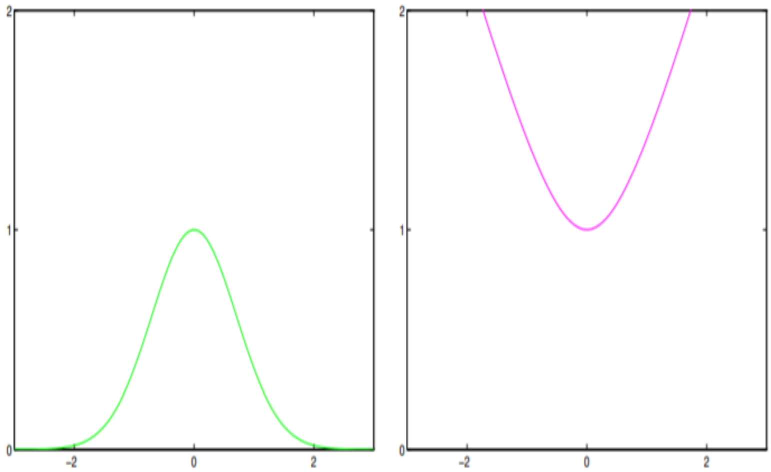


Figure 2.5 Gaussian RBF

RBFNN consist of input, hidden and output layer. RBFNN possess just one hidden layer which is called as feature vector. It is structurally same as multi-layer perceptron. It transforms the input signal into other form which is fed into the network to get linear separability.

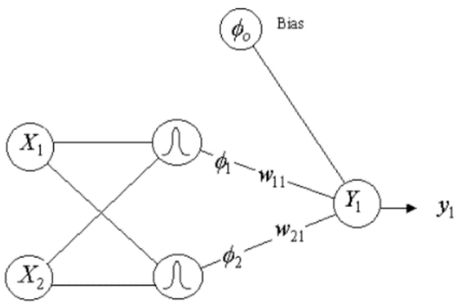


Figure 2.6 Architecture of RBFNN

The architecture of the simple RBFNN model is shown in figure 2.6. The nonlinear transfer function is applied to the feature vector. The linear separability decreases if we increase the dimension of the vector. For a radial basis function, we define a receptor and confrontal graphs are drawn around it and gaussian functions are used for its mapping and hence we define radial distance r=||x-t||.

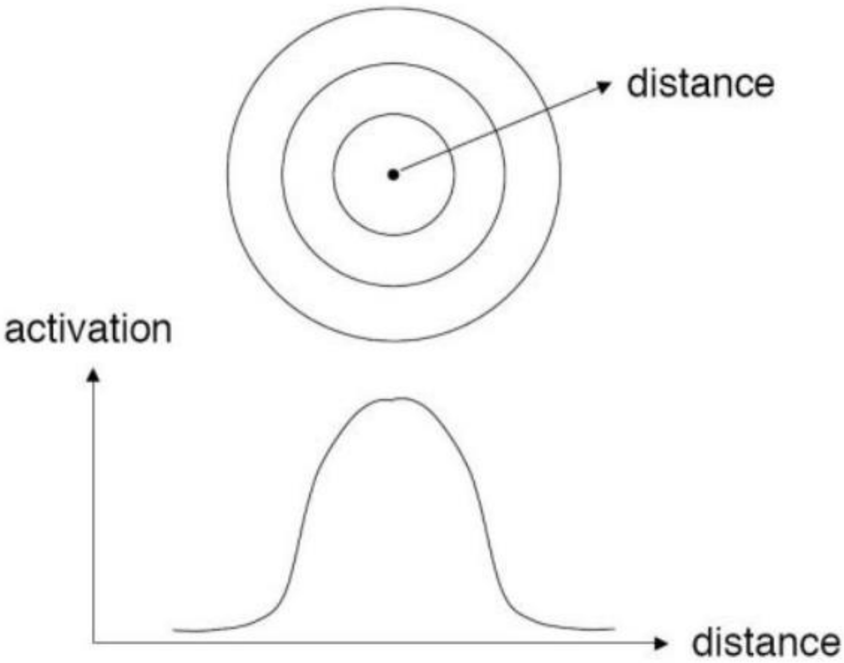


Figure 2.7 Radial distance and Radial basis function

Gaussian radial function is given

$$\varphi(x) = \exp(\frac{-r^2}{2\sigma^2}) \tag{2.14}$$

where σ>0

In RBF, first we should train the hidden layers using back propagation. It is a curve fitting method which fits the curve during the training phase which is done through stochastic approximation. Updating of the weights is done in the second training phase. Every node in the hidden layer shows the approximation. So the first phase is basically the clustering algorithm stage where cluster centres are defined which is given to hidden layer as receptors. Variance is to be calculated for each receptors then.

CHAPTER 3
DYNAMICS OF ROBOT MANIPULATOR

3.1 NON LINEAR CONTROL STRUCTURE

If the system is mildly nonlinear we can ignore the nonlinearity in designing the controller i.e. to omit the nonlinearity in the design model but consider its effect in evaluating the system performance. When the system is significantly nonlinear, it is dealt with by linearization about a selected operating point using Taylor series. A linear controller is designed using the first approximation. The disturbances in actual state about the equilibrium will be small if the controller works effectively. The higher order terms should cause no problem since the controller is designed to counteract the disturbances. It cannot work always so it is necessary to test the design for system performance and stability-analytically, by Lyapunov stability analysis and/or by simulation.

The nonlinearity inherent sometimes is so dominant that linearization approach fails to fulfil the requirement on system performance. This demands to incorporates the nonlinear dynamics into the design process. One such approach is feedback linearization. Unlike the first order approximation approach in which the higher order terms are ignored, this technique utilizes the feedback to render the given system, a linear input output dynamics. The linear control techniques can be applied on obtained linear system to address design issue.

This linearization can be viewed from two perspectives. First, it neglects all the higher order term. Second, the linear terms depend on the equilibrium point. These two uncertainties shows why this linearization approach is incapable of dealing with the system operates over wide dynamic range. Feedback linearization does not guarantees the system performance over the whole dynamic range since it often performed locally-around a specific equilibrium point. To eliminate the drawbacks of first approximation approach is to design the various control laws corresponding to several operating points that covers the whole dynamics of the system. These linear controller then pieced together to obtain a nonlinear control law. This approach is called gain scheduling. This approach accommodates the variation of the first order terms with respect to equilibrium. The plant parameter variation affects the system performance so an adaptive controller is used at a cost of increased complexity of the controller. Adaptive control is both time varying and nonlinear which increases the difficulty of stability and performance analysis. Feedback linearization transforms the dynamics of nonlinear system to that of linear system on which control designs can be applied.

3.2 FEEDBACK LINEARIZATION FOR TWO LINK ROBOT ARM

This technique helps to cancel the nonlinearities in a nonlinear system so that the closed loop dynamics converts in linear form. This is shown in the control design of two link robot as shown in figure 3.1

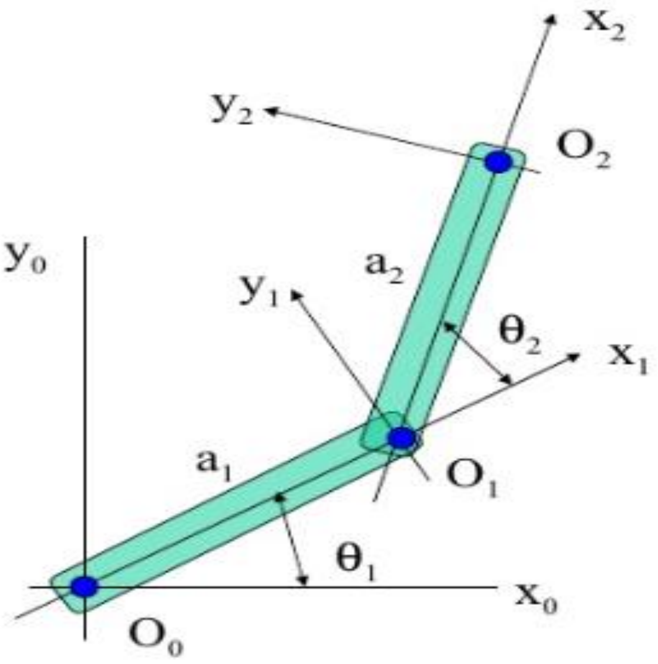


Figure 3.1 A two link robot

To determine the arm dynamics of a two link planar robot arm manipulator, we assume that the link masses m_1 and m_2 are concentrated at the ends of the links of lengths a_1 and a_2 respectively. We define the angle of the first link θ_1 with respect to the orientation of the first link as depicted in fig 3.1. The angle of the second link θ_2 is defined with respect to the orientation of the first link. Torques τ_1 and τ_2 are applied by the actuators to control the angles θ_1 and θ_2 respectively.

Let us derive the dynamics of the two link arm from first principles of Lagrange's equation of motion:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}} - \frac{\partial L}{\partial \theta} = \tau; \; \theta = [\tau_1 \quad \tau_2]^T \tag{3.1}$$

with the Lagrangian L defined in terms of the kinetic energy K and potential energy P as

$$L = K(\theta, \dot{\theta}) - P(\theta) \quad (3.2)$$

For link 1, we have the positions and velocities:

$$x_1 = a_1 \cos\theta_1 \quad (3.3)$$

$$y_1 = a_1 \sin\theta_1 \quad (3.4)$$

$$\dot{x}_1 = -a_1 \dot{\theta}_1 \sin\theta_1 \quad (3.5)$$

$$\dot{y}_1 = a_1 \dot{\theta}_1 \cos\theta_1 \quad (3.6)$$

$$v_1^2 = \dot{x}_1^2 + \dot{y}_1^2 = a_1^2 \dot{\theta}_1^2 \quad (3.7)$$

The kinetic and potential energies, for link 1, are

$$K_1 = \frac{1}{2} m_1 v_1^2 = \frac{1}{2} m_1 a_1^2 \dot{\theta}_1^2 \quad (3.8)$$

$$P_1 = m_1 g y_1 = m_1 g a_1 \sin\theta_1 \quad (3.9)$$

For link 2, we have the positions and velocities:

$$x_2 = a_1 \cos\theta_1 + a_2 \cos(\theta_1 + \theta_2) \quad (3.10)$$

$$y_2 = a_1 \sin\theta_1 + a_2 \sin(\theta_1 + \theta_2) \quad (3.11)$$

$$\dot{x}_2 = -a_1\dot{\theta}_1 \sin\theta_1 - a_2(\dot{\theta}_1 + \dot{\theta}_2)\sin(\theta_1 + \theta_2) \quad (3.12)$$

$$\dot{y}_2 = a_1\dot{\theta}_1 \cos\theta_1 + a_2(\dot{\theta}_1 + \dot{\theta}_2)\cos(\theta_1 + \theta_2) \quad (3.13)$$

$$v_2^2 = \dot{x}_2^2 + \dot{y}_2^2 = a_1^2\dot{\theta}_1^2 + a_2^2(\dot{\theta}_1 + \dot{\theta}_2)^2 + 2a_1a_2(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2)\cos\theta_2 \quad (3.14)$$

Therefore kinetic energy for link 2 is

$$K_2 = \frac{1}{2} m_2 v_2^2 = \frac{1}{2} m_2 a_1^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 a_2^2 (\dot{\theta}_1 + \dot{\theta}_2)^2 + m_2 a_1 a_2 (\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2)\cos\theta_2 \quad (3.15)$$

The potential energy for link 2 is

$$P_1 = m_1 g y_1 = m_2 g(a_1 \sin\theta_1 + a_2 \sin(\theta_1 + \theta_2) \quad (3.16)$$

The lagrangian for the entire arm is

$$L = K - P$$

$$= K_1 + K_2 - P_1 - P_2$$

$$= \frac{1}{2}(m_1 + m_2)a_1^2\dot{\theta}_1^2 + m_2 a_2^2(\dot{\theta}_1 + \dot{\theta}_2)^2 + m_2 a_1\, a_2 \left(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2\right)\cos\theta_2 - (m_1 + m_2)g a_1 \sin\theta_1 - m_2 g a_2 \sin(\theta_1 + \theta_2) \quad (3.17)$$

Equation 3.1 is the vector equation comprised of two scalar equations. The individual terms are needed to write down these two equations are

$$\frac{\partial L}{\partial \dot{\theta}_1} = (m_1 + m_2)a_1^2\dot{\theta}_1^2 + m_2 a_2^2(\dot{\theta}_1 + \dot{\theta}_2) m_2 a_1 a_2 (2\dot{\theta}_1 + \dot{\theta}_2)\cos\theta_2 \quad (3.18)$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_1} = (m_1 + m_2)a_1^2\ddot{\theta}_1 + m_2 a_1 a_2\left(2\ddot{\theta}_1 + \ddot{\theta}_2\right)\cos\theta_2 - m_2 a_1 a_2 (2\theta_1\dot{\theta}_2 + \dot{\theta}_2^2)\sin\theta_2 \quad (3.19)$$

$$\frac{\partial L}{\partial \theta_1} = -(m_1 + m_2)g a_1 \cos\theta_1 - m_2 g a_2 \cos(\theta_1 + \theta_2) \quad (3.20)$$

$$\frac{\partial L}{\partial \dot{\theta}_2} = m_2 a_2^2\left(\dot{\theta}_1 + \dot{\theta}_2\right) + m_2 a_1 a_2 \dot{\theta}_1 \cos\theta_2 \quad (3.21)$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_2} = m_2 a_2^2\left(\dot{\theta}_1 + \dot{\theta}_2\right) + m_2 a_1 a_2 \ddot{\theta}_1 \cos\theta_2 - m_2 a_1 a_2 \dot{\theta}_1 \sin\theta_2 \quad (3.22)$$

$$\frac{\partial L}{\partial \theta_2} = -m_2 a_1 a_2\left(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2\right)\sin\theta_2 - m_2 g a_2 \cos(\theta_1 + \theta_2) \quad (3.23)$$

According to Lagrange's equation (3.1), the arm dynamics are given by the two coupled nonlinear differential equations

$$\tau_1 = [(m_1 + m_2)a_1^2 + m_2 a_2^2 + 2m_2 a_1 a_2 \cos\theta_2]\ddot{\theta}_1 + [m_2 a_2^2 + m_2 a_1 a_2 \cos\theta_2]\ddot{\theta}_2 - m_2 a_1 a_2\left(2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2\right)\sin\theta_2 + (m_1 + m_2)g a_1 \cos\theta_1 + m_2 g a_2 \cos(\theta_1 + \theta_2) \quad (3.24)$$

$$\tau_2 = [m_2 a_2^2 + m_2 a_1 a_2 \cos\theta_2]\ddot{\theta}_1 + m_2 a_2^2\ddot{\theta}_2 + m_2 a_1 a_2 \dot{\theta}_1^2 \sin\theta_2 + m_2 g a_2 \cos(\theta_1 + \theta_2) \quad (3.25)$$

writing the arm dynamics in vector form, yields

$$\begin{bmatrix} (m_1 + m_2)a_1^2 + m_2 a_2^2 + 2m_2 a_1 a_2 \cos\theta_2 & m_2 a_2^2 + m_2 a_1 a_2 \cos\theta_2 \\ m_2 a_2^2 + m_2 a_1 a_2 \cos\theta_2 & m_2 a_2^2 \end{bmatrix}\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} -m_2 a_1 a_2\left(2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2\right)\sin\theta_2 \\ m_2 a_1 a_2 \dot{\theta}_1^2 \sin\theta_2 \end{bmatrix} + \begin{bmatrix} (m_1 + m_2)g a_1 \cos\theta_1 \\ m_2 g a_2 \cos(\theta_1 + \theta_2) \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad (3.26)$$

Writing the dynamics of two link arms completely as:

$$M(\theta)\dot{\theta} + V\left(\theta, \dot{\theta}\right) + G(\theta) = \tau \quad (3.27)$$

where the symmetric inertia matrix

$$M(\theta) = \begin{bmatrix} \alpha + \beta + 2\mu \cos\theta_2 & \beta + \mu \cos\theta_2 \\ \beta + \mu \cos\theta_2 & \beta \end{bmatrix} \quad (3.28)$$

And nonlinear terms

$$N(\theta, \dot{\theta}) = V(\theta, \dot{\theta}) + G(\theta) \quad (3.29)$$

are given by

$$V(\theta, \dot{\theta}) = \begin{bmatrix} \mu(2\theta_1\dot{\theta}_2 + \dot{\theta}_2^2) \\ \mu\dot{\theta}_1^2 \sin\theta_2 \end{bmatrix} \quad (3.30)$$

$$G(\theta) = \begin{bmatrix} \alpha e_1 \cos\theta_1 + \mu e_1 \cos(\theta_1 + \theta_2) \\ \mu e_1 \cos(\theta_1 + \theta_2) \end{bmatrix} \quad (3.31)$$

$\alpha = (m_1 + m_2)a_1^2; \beta = m_2 a_2^2; \mu = m_2 a_1 a_2; e_1 = \frac{g}{a_1}$

This is called as Brunovsky canonical form. Many systems like the robot arm is in this form naturally. the general state can be transformed to brunovsky form by finding a suitable state space transformation.

Defining the state vector as

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_{11} \\ x_{12} \\ x_{21} \\ x_{22} \end{bmatrix} = \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \quad (3.32)$$

we get the following state equations:

$$\dot{x_1} = \dot{\theta} = x_2 \quad (3.33)$$

$$\dot{x_2} = \ddot{\theta} = -M^{-1}(\theta)\left[V(\theta, \dot{\theta}) + G(\theta)\right] + M^{-1}(\theta)\tau = -M^{-1}(x_1)[V(x_1, x_2) + G(x_1)] + M^{-1}(x_1)\tau = f(x) + g(x)\tau \quad (3.34)$$

where

f(x)= $-M^{-1}(x_1)[V(x_1, x_2) + G(x_1)]; g(x) = M^{-1}(x_1)$

The control law

$$\tau = g^{-1}[-f(x) + u] \quad (3.35)$$

Linearizes the system (3.33) to yield

$$x_1^{\cdot} = x_2$$

$$\dot{x_2} = u \tag{3.36}$$

$$\begin{bmatrix} \dot{x_{11}} \\ \dot{x_{12}} \\ \dot{x_{21}} \\ \dot{x_{22}} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_{11} \\ x_{12} \\ x_{21} \\ x_{22} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{3.37}$$

Equation (3.37) can be expressed as

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} u \tag{3.38}$$

A two step design process follows :

Step 1 select feedback control u(t) using linear model

Step 2 compute the required arm torque from equation (3.35)

$$\tau = g^{-1}[-f(x) + u]$$

This process is termed as feedback linearization.

3.3 SLIDING MODE CONTROLLER FOR A TWO LINK ROBOT

The plant model is given by equation (3.33)

$$x_1^{\cdot} = \dot{\theta} = x_2$$

$$\dot{x_2} = \ddot{\theta} = f(x) + g(x)\tau \tag{3.39}$$

where

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_{11} \\ x_{12} \\ x_{21} \\ x_{22} \end{bmatrix} = \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}; \tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{3.40}$$

θ_1 and θ_2 are the angles of the two links defined in figure 3.1, and τ_1 and τ_2 are the torques applied by the actuators to control the angles θ_1 and θ_2 respectively.

Variable structure sliding mode controller design consists of the following two phases:

(a)Designing of sliding surface to obtain the desired system behaviour, when restricted to the surface. The system will remain at that surface for further times. The plant dynamics will always be limited to the surface and is robust towards the disturbances.

(b)Selection of the feedback gains of the controller, so that the closed loop system is stable to the sliding surface.

For a two link robot manipulator, our aim is to track the desired motion trajectory $\theta_d(t)$

The tracking error is defined as

$$e(t) = \theta_d(t) - \theta(t) \tag{3.41}$$

$$\dot{e(t)} = \dot{\theta_d}(t) - \dot{\theta}(t); \ddot{e}(t) = \ddot{\theta_d}(t) - \ddot{\theta}(t) \tag{3.42}$$

Defining $\widetilde{x_1} = e$ and $\widetilde{x_2} = \dot{e}$,robot dynamics can be written in the form:

$$\dot{\widetilde{x_1}} = \widetilde{x_2}$$

$$\dot{\widetilde{x_2}} = \ddot{\theta_d}(t) - f(x) - g(x)\tau \tag{3.43}$$

Now we use linear sliding surface which is defined by the equation

$$\sigma(\tilde{x}) = \lambda\widetilde{x_1} + I\widetilde{x_2} \tag{3.44}$$

Or

$$\begin{bmatrix} \sigma_1(\tilde{x}) \\ \sigma_2(\tilde{x}) \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} \widetilde{x_{11}} \\ \widetilde{x_{12}} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \widetilde{x_{21}} \\ \widetilde{x_{22}} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{3.45}$$

$$\sigma_1(\tilde{x}) = \lambda_1\tilde{x}_{11} + \widetilde{x_{21}} = \lambda_1(\theta_{1d} - \theta_1) + (\dot{\theta_{1d}} - \dot{\theta}_1)$$

$$\sigma_2(\tilde{x}) = \lambda_2\tilde{x}_{12} + \widetilde{x_{22}} = \lambda_2(\theta_{2d} - \theta_2) + (\dot{\theta_{2d}} - \dot{\theta}_2) \tag{3.46}$$

Assume the coefficient of $\widetilde{x_{21}}$ to be unity.

Now combine the equations of the plant and the sliding surface (Eqns (3.43) and (3.44))

$$\widetilde{x_2} = -\lambda\tilde{x}_1 \tag{3.47}$$

Therefore

$$\dot{\widetilde{x_1}} = -\lambda\widetilde{x_1} \tag{3.48}$$

The above equations describes the system dynamics in sliding. The response is specified by the selection of the parameters λ_1 and λ_2 of the switching surface. In sliding mode, the system is not affected by the model uncertainties. After designing the sliding surface, we build a feedback controller. The objective of the controller is to derive the plant state to the sliding surface and maintain it on the surface for further time. A Lyapunov approach has been used for building the controller.

The Lyapunov function is given below:

$$V = \frac{1}{2}\sigma^T\sigma = \frac{1}{2}(\sigma_1^2 + \sigma_2^2) \quad (3.49)$$

The above function is used as the Lyapunov function and the time derivative of the chosen Lyapunov function comes out to be negative definite with respect to switching surface and ensuring the motion of the state trajectory to the surface. We have to find τ so that

$$\frac{d}{dt}\frac{1}{2}\sigma^T\sigma = \sigma^T\dot{\sigma} < 0 \quad (3.50)$$

$$\sigma^T\dot{\sigma} = \sigma^T[\lambda\dot{\widetilde{x_1}} + I\dot{\widetilde{x_2}}] \quad (3.51)$$

The controller structure form is given below

$$\tau = g^{-1}(x)\left[-f(x) + \dot{x_{2d}} + \lambda(\dot{x_{1d}} - \dot{x_1}) + \begin{Bmatrix} k_1 sgn(\sigma_1) \\ k_2 sgn(\sigma_2) \end{Bmatrix}\right] \quad (3.52)$$

where k_1>0 and k_2>0 are the gains which has to be determined so that the condition $\sigma^T\dot{\sigma} < 0$ is satisfied. We substitute τ, given by (3.52) into the expression $\sigma^T\dot{\sigma}$

$$\sigma^T\dot{\sigma} = -[\sigma_1 \quad \sigma_2]\begin{bmatrix} k_1 sgn(\sigma_1) \\ k_2 sgn(\sigma_2) \end{bmatrix} \quad (3.53)$$

$$= -\sigma_1 k_1 sgn(\sigma_1) - \sigma_2 k_2 sgn(\sigma_2) = -k_1|\sigma_1| - k_2|\sigma_2| < 0 \quad (3.54)$$

The sliding surface $\sigma(\tilde{x}) = 0$ is therefore asymptotically attractive. The larger the value of gains, faster the trajectory converges to the sliding surface. The tolerance of the sliding mode control to the model imprecision and disturbances, is high and thus satisfying stability requirement. The Simulink model for the sliding mode control of the two link robot manipulator is given in figure 3.2

Simulation of this controller for a two link robot manipulator was done in MATLAB and results are attached at the end.

The specifications of the robot manipulator is taken as:

$$m_1 = 1kg; m_2 = 1kg; l_1 = 1m; l_2 = 1m; g = \frac{9.8kgm}{s^2}; \theta_{d1}(t) = \sin(\pi t); \theta_{d2}(t) = \cos(\pi t) \quad (3.55)$$

Now the objective is to construct an RBF network sliding mode control for robot manipulator. This is completely opposite to the SMC technique so far used as the unknow disturbances are approximated by RBF neural network.

3.4 RBF NETWORK ADAPTIVE SLIDING MODE CONTROL FOR ROBOT MANIPULATOR

3.4.1 CONTROLLER DESIGN

The dynamic equation of 2-joint manipulator is as follows:

$$H(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau - F(\dot{q}) - \tau_d \qquad (3.56)$$

where H(q) is a 2x2 positive definite integral matrix, G(q) is a 2x1 inertial vector, $C(q,\dot{q})$ is a 2x2 inertial matrix, τ_d is the unknown disturbance, $F(\dot{q})$ is friction force, and τ is the control input

The tracking error is denoted as:

$$e(t) = q_d(t) - q(t) \qquad (3.57)$$

Sliding variable is selected as:

$$s = \dot{e} + \dot{\Lambda} e \qquad (3.58)$$

where Λ is a symmetric positive definite matrix and $\Lambda = \Lambda^T > 0$, therefore , we have

$$\dot{q} = -s + \dot{q}_d + \Lambda e \qquad (3.59)$$

$$H\dot{s} = H(\ddot{q}_d - \ddot{q} + \Lambda\dot{e}) = H(\ddot{q}_d + \Lambda\dot{e}) - H\ddot{q}$$

$$= H(\ddot{q}_d + \Lambda\dot{e}) + C\dot{q} + G + F + \tau_d - \tau$$

$$= H(\ddot{q}_d + \Lambda\dot{e}) - Cs + C(\dot{q}_d + \Lambda e) + G + F + \tau_d - \tau$$

$$= -Cs - \tau + f + \tau_d \qquad (3.60)$$

where f(x)=$H(\ddot{q}_d + \Lambda\dot{e}) + C(q_d + \Lambda e) + G + F$

as f(x) is unknown, hence RBF network is adopted to approximate f(x). input is selected based on expression of f(x):

$$x = [e^T \quad \dot{e}^T \quad q_d^T \quad \dot{q}_d^T \quad \ddot{q}_d^T] \qquad (3.61)$$

The controller is designed as follows:

$$\tau = \hat{f}(x) + K_v s \qquad (3.62)$$

where K_v is a symmetrical positive definite constant matrix, $\hat{f}(x)$ is the output of RBF network, $\hat{f}(x)$approximatesf(x). The RBF-NN weights are adjusted using adaptive laws

From equations (3.62) and(3.60) we have

$$H\dot{s} = -Cs - \hat{f}(x) - Kvs + f(x) + \tau_d$$

$$= -(K_v + C)s + \tilde{f}(x) + \tau_d = -(K_v + C)s + \varsigma_0 \qquad (3.63)$$

where $\tilde{f}(x) = f(x) - \hat{f}(x), \varsigma_0 = \tilde{f}(x) + \tau_d$

Lyapunov function is selected as:

$$L = \frac{1}{2}s^T Hs \qquad (3.64)$$

Therefore

$$\dot{L} = s^T Hs + \frac{1}{2}s^T Hs = -s^T K_v s + \frac{1}{2}s^T (H - 2C)s + s^T \varsigma_0 \qquad (3.65)$$

$$\dot{L} = s^T \varsigma_0 - s^T K_v s \qquad (3.66)$$

It shows that with K_v, the stability of control system depends on ς_0, the magnitude of τ_d

The algorithm suggested to approximate f(x) is given below

$$\phi_i = g(\frac{||x - c_i||^2}{\sigma_i^2}) \quad \text{i= 1,2,…,n} \qquad (3.67)$$

$$y = W^{*T}\varphi(x), f(x) = W^{*T}\varphi(x) + \varepsilon \qquad (3.68)$$

Where x is the input state of network, $\varphi(x) = [\phi_1 \quad \phi_2 \quad \ldots\ldots \quad \phi_n]^T$ ε is the approximation error, W^* is the desired weight vector.

3.4.2 SLIDING MODE CONTROL WITH RESPECT TO APPROXIMATION OF f(x)

The output of the adopted RBF network for approximation of f(x) is as follows

$$\hat{f}(x) = \widehat{W}^T \varphi(x) \qquad (3.69)$$

Select

$$\widetilde{W} = W^* - \widehat{W}, ||W^*||_F \leq W_{max} \qquad (3.70)$$

Therefore we have

$$\varsigma_0 = \tilde{f}(x) + \tau_d = \widetilde{W}^T \varphi(x) + \varepsilon + \tau_d \qquad (3.71)$$

Controller is designed as

$$\tau = \hat{f}(x) + K_v s - v \tag{3.72}$$

Where v is the robust element required to overcome the network approximation error ε and the disturbance τ_d

From equations (3.72) and(3.69) we have

$$H\dot{s} = -(K_v + C)s + \widetilde{W}^T\varphi(x) + (\varepsilon + \tau_d) + v = -(K_v + C)s + \varsigma_1 \tag{3.73}$$

where $\varsigma_1 = \widetilde{W}^T\varphi(x) + (\varepsilon + \tau_d) + v$

The robust element is v is designed as

$$v = -(\varepsilon_N + b_d)sgn(s) \tag{3.74}$$

Where $||\varepsilon|| \leq \varepsilon_N, ||\tau_d|| \leq b_d$.

3.4.3 STABILITY ANALYSIS

Lyapunov function is selected as

$$L = \frac{1}{2}s^T H s + \frac{1}{2}tr(\widetilde{W}^T F_W^{-1}\widetilde{W}) \tag{3.75}$$

Where H and F_w are positive matrices, so we have

$$\dot{L} = s^T H\dot{s} + \frac{1}{2}s^T Hs + tr(\widetilde{W}^T F_W^{-1}\widetilde{W}) \tag{3.76}$$

From equation (3.73) we have

$$\dot{L} = -s^T K_v s + \frac{1}{2}s^T\left(\dot{H} - 2C\right)s + tr\widetilde{W}^T\left(F_W^{-1}\widetilde{W} + \varphi s^T\right) + s^T(\varepsilon + \tau_d + v) \tag{3.77}$$

As we know that manipulator has the characteristics of $s^T\left(\dot{H} - 2C\right)s$=0 select $\widetilde{W} = -F_W\varphi s^T$ that is adaptive rule of network is

$$\widetilde{W} = F_W\varphi s^T \tag{3.78}$$

Therefore

$$\dot{L} = -s^T K_v s + s^T(\varepsilon + \tau_d + v) \tag{3.79}$$

Because

$$s^T(\varepsilon + \tau_d + v) = s^T(\varepsilon + \tau_d) + s^T v = s^T(\varepsilon + \tau_d) - ||s||(\varepsilon_N + b_d) \leq 0 \tag{3.80}$$

We have

$\dot{L} \leq 0$

3.5 SIMULATION OF TWO LINK ROBOT MANIPULATOR

Kinetic equation of two joint robot manipulator is:

$$H(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau - F(\dot{q}) - \tau_d \quad (3.81)$$

where

$$H(q) = \begin{bmatrix} p_1 + p_2 + 2p_3 cosq_2 & p_2 + p_3 cosq_2 \\ p_2 + 2p_3 cosq_2 & p_2 \end{bmatrix} \quad (3.82)$$

$$C(q,\dot{q}) = \begin{bmatrix} -p_3 q_2 sinq_2 & -p_3(\dot{q}_1 + \dot{q}_2) sinq_2 \\ p_3 \dot{q}_1 sinq_2 & 0 \end{bmatrix} \quad (3.83)$$

$$G(q) = \begin{bmatrix} p_4 g cosq_1 & p_5 g cos(q_1 + q_2) \\ p_5 g cos(q_1 + q_2) \end{bmatrix} \quad (3.84)$$

$$F(\dot{q}) = 0.02 sgn(\dot{q}), \ \tau_d = [0.2 sint \quad 0.2 sint]^T \quad (3.85)$$

Let p=$[p_1 \quad p_2 \quad p_3 \quad p_4 \quad p_5] = [2.9 \quad 0.76 \quad 0.87 \quad 3.04 \quad 0.87]$. Gaussian function of the RBF network is selected to control the neural network. If the parameter is not suitable then mapping of the gauss function cannot be obtained and the RBF network is unavailable. The model for this is given in the figure 3.2.

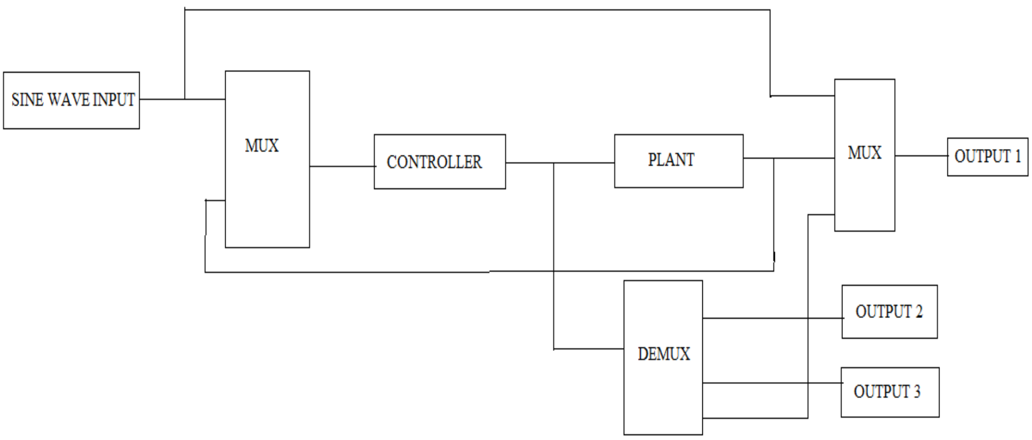


Figure 3.2 Block diagram for RBF network sliding mode control for two link robot manipulator

CHAPTER 4
INVERTED PENDULUM

4.1 SLIDING MODE CONTROL FOR AN INVERTED PENDULUM

The backstepping sliding mode approach incorporates the idea that a complex nonlinear system is decomposed into subsystems, and degree of each subsystem doesn't exceed that of the whole system. Many physical systems do not satisfy the matching condition and the backstepping approach effectively deals with the systems having multiple dynamics and with mismatched uncertainties.

In backstepping approach an appropriate functions of state variables are selected recursively for lower dimensions subsystems. At every backstepping stage there will be a new pseudo control design from preceding stages. When the process is completed a feedback design which achieves the original design objective by using a final Lyapunov function is obtained which is formed by summing the Lyapunov functions associated with individual design stage. This approach provides a systematic framework for the design of tracking and regulation strategies.

The major drawback of backstepping approach is that certain functions must be linear and tedious analysis is required to determine regression matrices. So neural networks (NN's) is preferred to estimate certain nonlinear functions. By using NN at each stage, we can design control law using backstepping approach but it is not required that the parameters have to be linear in nature and no regression matrices are needed. The neural network weights are tuned online with no learning phase required.

The backstepping method and the sliding mode control are integrated to design a backstepping sliding mode controller which provides the robust control for uncertain systems.

Consider a one link inverted pendulum as follows:

$$\dot{x_1} = x_2 \tag{4.1}$$

$$\dot{x_2} = \frac{gsinx_1 - mlx_2^2 \cos x_1 \sin x_1/(m_c+m)}{l(\frac{4}{3} - mcos^2 x_1/(m_c+m))} + \frac{\cos x_1/(m_c+m)}{l(\frac{4}{3} - mcos^2 x_1/(m_c+m))} u \tag{4.2}$$

where x_1 and x_2 are the oscillation angle and the oscillation rate respectively.

g=9.8 m/s^2, m_c is the vehicle mass, m_c =1 kg, m is the mass of the pendulum bar, m=0.1 kg, l is one half of pendulum length, l=0.5m, u is the control input. The desired trajectory

$x_d(t)=0.1\sin(\pi t)$, $c_1=35$ and $c_2=15$. The initial state of the inverted pendulum is given as [$-\pi/60$ 0].

4.2 SYSTEM EQUATIONS

The equation of motion can be derived from the free body diagram. Two free body diagrams are given in figure 4.1.

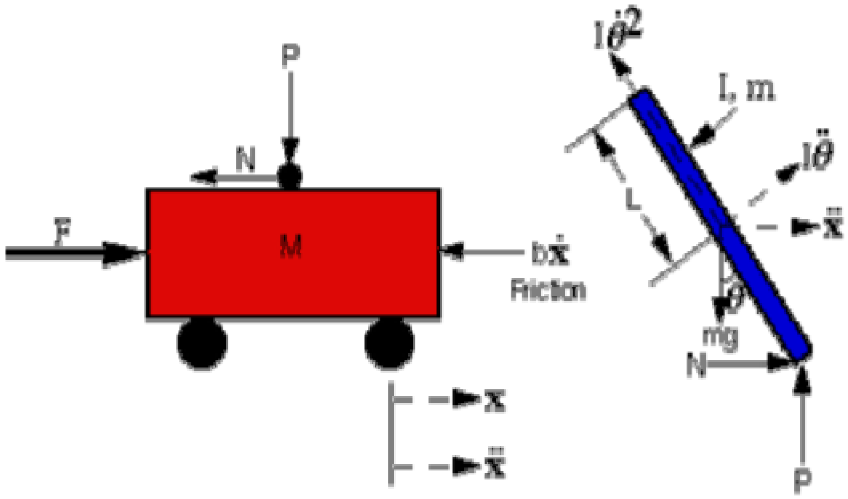


Figure 4.1 Free body diagram of inverted pendulum

Summation of forces in FBD of the wagon in x direction will give us

$$M\ddot{x} + b\dot{x} + N = F \tag{4.3}$$

There is no use of summing up forces in the vertical direction because there is no vertical motion here and also we assume that the vertical forces is balanced by the reaction force of earth. Due to the moment of the pendulum, a force is applied in the horizontal direction which can be computed as follows:

$$\tau = r * F = I\ddot{\theta} \tag{4.4}$$

$$F = \frac{I\ddot{\theta}}{r} \tag{4.5}$$

$$= \frac{ml^2\ddot{\theta}}{l}$$

$$= ml\ddot{\theta}$$

$ml\ddot{\theta}\cos\theta$ is the component of the above force in the direction of N.

So we get an equation for N by summing up forces in horizontal direction:

$$N = m\ddot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta \qquad (4.6)$$

Substituting equation () in (), we get our first equation of motion

$$(M+m)\ddot{x} + b\dot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta = F$$

(4.47)

Sum up the forces perpendicular to the pendulum and we will get the second equation of motion which is :

$$P\sin\theta + N\cos\theta - mg\sin\theta = ml\ddot{\theta} + m\ddot{\theta}\cos\theta \qquad (4.7)$$

Forces are summed up around the centroid to eliminate the terms N and P as shown below:

$$-PI\sin\theta - NI\cos\theta = I\ddot{\theta} \qquad (4.8)$$

By combining the above two equations, we get the second dynamics

$$(I+ml^2)\ddot{\theta} + mgl\sin\theta = -ml\ddot{x}\cos\theta \qquad (4.9)$$

Now we get the equations which completely define the dynamics of the inverted pendulum which are shown below:

$$(M+m)\ddot{x} + b\dot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta = F \qquad (4.10)$$

$$(I+ml^2)\ddot{\theta} + mgl\sin\theta = -ml\ddot{\theta}\cos\theta \qquad (4.11)$$

The above two equations are nonlinear in nature and hence there is a need to linearize these to make them operate in a wide range. Since we make the pendulum to get stabilized at some unstable equilibrium position which is pi radians away from equilibrium so the above equations should get stabilized at $\theta = \pi$.

After the two equations are linearized, these can be represented as:

$$(M+m)\ddot{x} + b\dot{x} - ml\ddot{\varphi} = u \qquad (4.12)$$

$$(I+ml^2)\ddot{\varphi} + mgl\varphi = ml\ddot{x} \qquad (4.13)$$

where $\theta = \pi + \varphi$ and u represent the input. Physically, it is known that the stabilization of pendulum angle occurs at a small angle from vertical so that it always points towards the centre of the path. Hence the pendulum is always seen to be falling towards the centre of the path on which the cart is moving. The only possible equilibrium position one could get in inverted pendulum model is the vertical position right at the middle of the track. If the cart is positioned at the left of this path, the pendulum can be stabilized by the control towards the right so that it should fall a little more towards the right direction. To be at par with the falling pendulum, the cart is required to move in the right direction which is the desired behaviour.

4.3 CONTROLLER DESIGN

The steps of the backstepping sliding mode control is as follows

Step1

Let

$$e_1 = x_1 - r \tag{4.14}$$

where r is the desired trajectory.

$$\dot{e} = \dot{x_1} - \dot{r} = x_2 - \dot{r} \tag{4.15}$$

Select the Lyapunov function as

$$V_1 = \frac{1}{2} e_1^2 \tag{4.16}$$

$$\dot{V_1} = e_1 \dot{e_1} = e_1(x_2 - \dot{r}) \tag{4.17}$$

To realize $\dot{V_1} \leq 0$, we let

$$s = x_2 + c_1 e_1 - \dot{r} = c_1 e_1 + \dot{e_1}, \quad c_1 > 0 \tag{4.18}$$

where s is the sliding variable. Thus we have,

$$\dot{V_1} = e_1 s - c_1 e_1^2 \tag{4.19}$$

If s=0 then $\dot{V_1} \leq 0$.

Step 2

Select the Lyapunov function

$$V_2 = V_1 + \frac{1}{2} s^2 \tag{4.20}$$

Since $\dot{s} = \dot{x_2} + c_1 \dot{e_1} - \ddot{r} = f(x,t) + b(x,t)u + d(x,t) + c_1 \dot{e_1} - \ddot{r}$, we have

$$\dot{V_2} = \dot{V_1} + s\dot{s} = e_1 s - c_1 e_1^2 + s(f(x,t) + b(x,t)u + d(x,t) + c_1 \dot{e_1} - \ddot{r}) \tag{4.21}$$

For $\dot{V_2} \leq 0$, a controller is designed as

$$u = \frac{1}{b(x,t)}(-f(x,t) - c_2 s - e_1 - c_1 \dot{e_1} + \ddot{r} - \eta|s|) \tag{4.22}$$

where $c_2 > 0, \eta \geq D$

We have

$$\dot{V}_2 = -c_1 e_1^2 - c_2 s^2 + s d(x,t) - \eta|s| \leq 0 \tag{4.23}$$

The block diagram for sliding mode control for the inverted pendulum is given in figure 4.2 below:

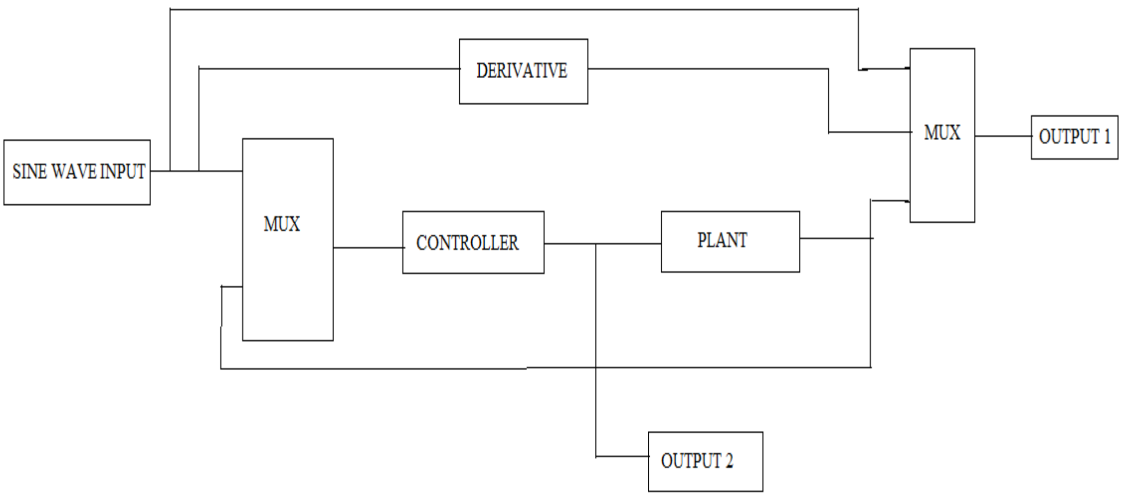


Figure 4.2 Block diagram for the sliding mode control of the inverted pendulum

4.4 BACKSTEPPING CONTROL BASED ON RBF FOR INVERTED PENDULUM

Consider one link inverted pendulum as follows

$$\dot{x_1} = x_2 \tag{4.24}$$

$$\dot{x_2} = \frac{g \sin x_1 - m l x_2^2 \cos x_1 \sin x_1/(m_c+m)}{l(\frac{4}{3} - m\cos^2 x_1/(m_c+m))} + \frac{\cos x_1/(m_c+m)}{l(\frac{4}{3} - m\cos^2 x_1/(m_c+m))} u \tag{4.25}$$

where x_1 and x_2 are the oscillation angle and the oscillation rate respectively.

g=9.8 m/s^2, m_c is the vehicle mass, m_c =1 kg, m is the mass of the pendulum bar, m=0.1 kg, l is one half of pendulum length, l=0.5m, u is the control input.

Consider the desired trajectory as $x_d(t) = 0.1 sint$

4.4.1 Backstepping Controller Design

Backstepping control is designed as follows

Step 1 $\quad \dot{e_1} = \dot{x_1} - \dot{x_{1d}}$ (4.26)

To realize $e_1 \to 0$, design lyapunov function as

$$V_1 = \frac{1}{2} e_1^2 \tag{4.27}$$

$$\dot{V_1} = e_1 \dot{e_1} = e_1(x_2 - \dot{x_{1d}}) \tag{4.28}$$

Choose $x_2 - \dot{x_{1d}} = -k_1 e_1, k_1 > 0$ then we have $\dot{V_1} = -k_1 e_1^2$

Step 2 Choose control law to realize $x_2 - \dot{x_{1d}} = -k_1 e_1$ as

$$x_{2d} = \dot{x_{1d}} - k_1 e_1 \tag{4.29}$$

We get error as

$$e_2 = x_2 - x_{2d} \tag{4.30}$$

$$\dot{e_2} = \dot{x_2} - \dot{x_{2d}} \tag{4.31}$$

Assume Lyapunov function as

$$V_2 = V_1 + \frac{1}{2}e_2^2 = \frac{1}{2}(e_1^2 + e_2^2) \tag{4.32}$$

Then

$$\begin{aligned}\dot{V_2} &= e_1(x_2 - \dot{x_{1d}}) + e_2\dot{e}_2 \\ &= e_1(x_{2d} + e_2 - \dot{x_{1d}}) + e_2\dot{e}_2 \\ &= -k_1e_1^2 + e_1e_2 + e_2(f + gu - \dot{x_{2d}}) \\ &= -k_1e_1^2 + e_1e_2 + e_2(f + \hat{g}u + (g - \hat{g})u\dot{x_{2d}}\end{aligned} \tag{4.33}$$

where $\hat{g}$ is the estimation of g.

for the estimation of $\dot{V_2} < 0$, the control law is designed as

$$u = \frac{-e_1 - \hat{f} + \dot{x_{2d}} - k_2 e_2}{\hat{g}} \tag{4.34}$$

where $\hat{f}$ is the estimation of f.

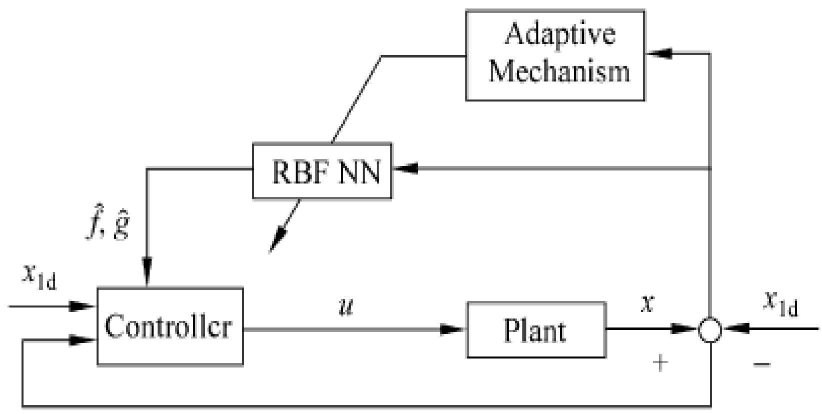


Figure 4.3 Block diagram of control scheme of inverted pendulum

The figure 4.3 shows the block diagram for the control scheme of inverted pendulum which consist of an adaptive mechanism and a RBF neural network which worked on the model of the inverted pendulum.

We finally obtain

$$\dot{V}_2 = -k_1 e_1^2 - k_2 e_2^2 + e_2(f - \hat{f}) + e_2(g - \hat{g})u \tag{4.35}$$

4.4.2 Designing of Adaptive Law

The unknown parameters $\hat{f}$ and$\hat{g}$ are evaluated by the neural network. Close loop neural based adaptive control scheme is shown in figure 4.1.

Two RBF's are used to approximate f and g respectively as given below

$$f = W_1^T h_1 + \epsilon_1 \tag{4.36}$$

$$g = W_2^T h_2 + \epsilon_2 \tag{4.37}$$

where W_i is the ideal neural network weight value, h_i is the gaussian function, i= 1,2, ϵ_i is the approximate error, and $||\epsilon||=||[\epsilon_1 \quad \epsilon_2]^T||<\epsilon_N$, $||W_i||_F \leq W_M$

By defining

$$\hat{f} = \widehat{W}_1^T h_1 \tag{4.38}$$

$$\hat{g} = \widehat{W}_2^T h_2 \tag{4.39}$$

where $\widehat{W_i}^T$ is the weight value estimation.

$Z = \begin{bmatrix} W_1 & 0 \\ 0 & W_2 \end{bmatrix}$, $||Z||_F \leq Z_M$, $\widehat{Z} = \begin{bmatrix} \widehat{W_1} & 0 \\ 0 & \widehat{W_2} \end{bmatrix}$ and $\tilde{Z} = Z - \hat{Z}$

The Lyapunov function is designed as

$$V = \frac{1}{2}\xi^T \xi + \frac{1}{2} tr(\tilde{Z}^T \Gamma^{-1} \tilde{Z}) \tag{4.40}$$

$V_2 = \frac{1}{2}\xi^T \xi, \eta > 0, \Gamma = 0$ is the positive definite matrix with proper dimension

$$\Gamma = \begin{bmatrix} \Gamma_1 & 0 \\ 0 & \Gamma_2 \end{bmatrix} \; and \; \xi = [e_1 \quad e_2]^T \tag{4.41}$$

Assume the adaptive law to be as follows:

$$\dot{\hat{Z}} = \Gamma h \xi^T - n\Gamma ||\xi|| \hat{Z} \tag{4.42}$$

where $h = [h_1 \quad h_2]^T$ and n is the positive number

From equations (4.24),(4.25) and (4.27), we get

$$\dot{V} = \xi^T\dot{\xi} + tr(\tilde{Z}^T\Gamma^{-1}\dot{Z})$$

$$= -k_1e_1^2 - k_2e_2^2 + \left(\widehat{W}_1^T h_1 + \epsilon_1\right)e_2 + \left(\widehat{W}_2^T h_2 + \epsilon_2\right)e_3 + tr(\tilde{Z}^T\Gamma^{-1}\dot{Z}) \quad (4.43)$$

Thus we have,

$$\dot{V} = -\xi^T k_e \xi + \xi^T\epsilon + \xi^T\tilde{Z}h + tr(\tilde{Z}^T\Gamma^{-1}\dot{Z}) + \widetilde{m}e_4u$$

$$= -\xi^T k_e \xi + \xi^T\epsilon\ tr\left(\tilde{Z}^T\Gamma^{-1}\dot{\tilde{Z}} + \tilde{Z}^T h\xi^T\right) + \widetilde{m}e_4u \quad (4.44)$$

where $k_e = [k_1 \quad k_2]^T$ and $\epsilon = [\epsilon_1 \quad \epsilon_2]^T$

Now we have

$$\dot{V} = -\xi^T k_e \xi + \xi^T\epsilon + n||\xi||tr(\tilde{Z}^T(Z - \tilde{Z})) \quad (4.45)$$

According to Schwartz equality we have $tr(\tilde{Z}^T(Z - \tilde{Z})) \leq \left\|\tilde{Z}\right\|_F \left\|Z\right\|_F - ||\widetilde{Z}||_F^2$

Since $k_{min}||\xi||^2 \leq \xi^T k\xi$, k_{min} is the minimum eigen value of k

Now equation 4.34 will transformed into

$$\dot{V} \leq -k_{min}\left\|\xi\right\|^2 + \xi_N\left\|\xi\right\| + n\left\|\xi\right\|(\left\|\tilde{Z}\right\|_F \left\|Z\right\|_F - ||\widetilde{Z}||_F^2)$$

$$\leq ||\xi||(-k_{min}\left\|\xi\right\| - \xi_N + n\left\|\tilde{Z}_F\right\|(||Z_F|| - Z_M)) \quad (4.46)$$

Since

$$k_{min}\left\|\xi\right\| - \xi_N + n(\left\|\widetilde{Z_F}\right\|^2 - \left\|Z_F\right\|Z_M)$$

$$= k_{min}\left\|\xi\right\| - \xi_N + n\left(\left\|\tilde{Z}_F\right\| - \frac{1}{2}Z_M\right)^2 - \frac{n}{4}Z_M^2 \quad (4.47)$$

This shows that $\dot{V} \leq 0$ as long as

$||\xi|| > \frac{\epsilon_N + \frac{n}{4}Z_M^2}{k_{min}}$ and $||Z_F|| > \frac{1}{2}Z_M + \sqrt{\frac{Z_M^2}{4} + \frac{\epsilon_N}{n}}$

From above two expressions it is evident that the tracking performance depends on the value of ϵ_N,n and k_{min}

If $e_1 \rightarrow 0$ and $e_2 \rightarrow 0$ then we get

$$\dot{e_1} = x_2 - \dot{x_{1d}} = e_2 + \dot{x_{2d}} - \dot{x_{1d}} = e_2 - k_1e_1 \rightarrow 0$$

Now considering the desired trajectory as $x_d(t)=0.1\sin(t)$ and taking the control law as given in equation (4.23) and adaptive law as given in equation (4.31) and by selecting $\Gamma_1 = 500$ and $\Gamma_2 = 0.50$ and n=0.10 and $k_1=k_2=35$. The parameters for the gaussian function c_i and b_i is taken as [-0.5 -0.25 0 0.25 0.5] and 15. The initial weight value of each neural net in the hidden layer is chosen to be 0.10. the initial state of the inverted pendulum is [π/60 , 0]. Simulation result are shown at the end of this report.

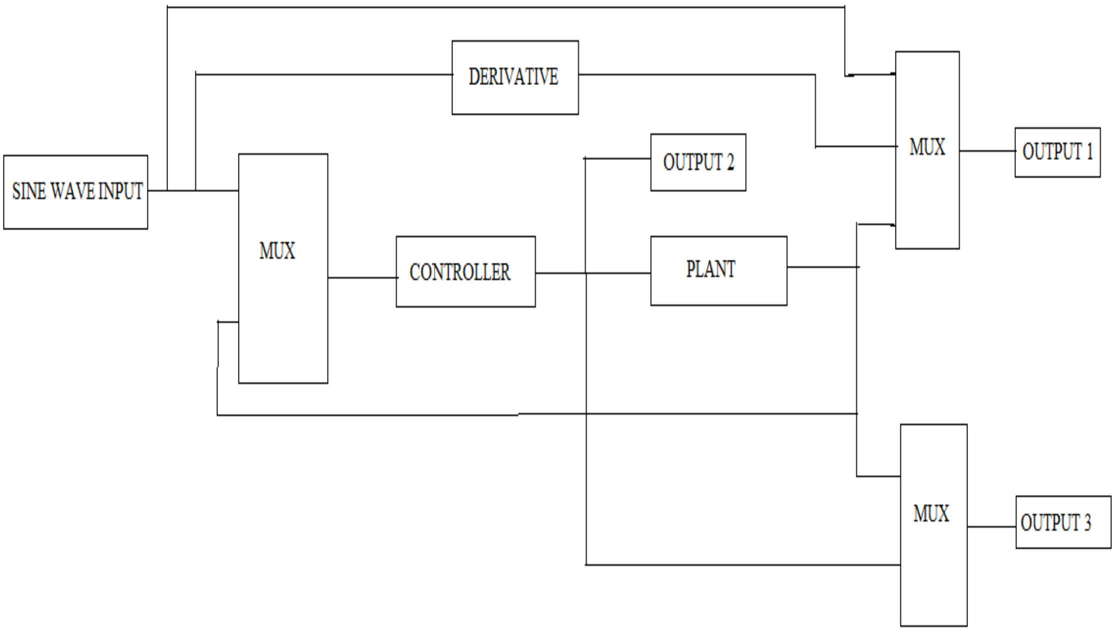


Figure 4.4 Block diagram for RBF sliding mode control for inverted pendulum

The figure 4.4 shows the block diagram of the RBF based sliding mode control for inverted pendulum system. A sine input is given and then with the application of the controller s-function and the plant s-function, the output is obtained.

CHAPTER 5

SIMULATION AND RESULTS

5.1 ROBOT MANIPULATOR

5.1.1 SLIDING MODE CONTROL FOR TWO LINK ROBOT MANIPULATOR

Simulation of sliding mode controller for the two link robot arm (m_1=1kg, m_2=1kg, l_1=1m, l_2=1m, g= 9.8kgm/s^2 , $\theta_{d1} = \sin(\pi t)$, $\theta_{d2} = cos(\pi t)$)was done using MATLAB. The below figure shows the tracking performances.

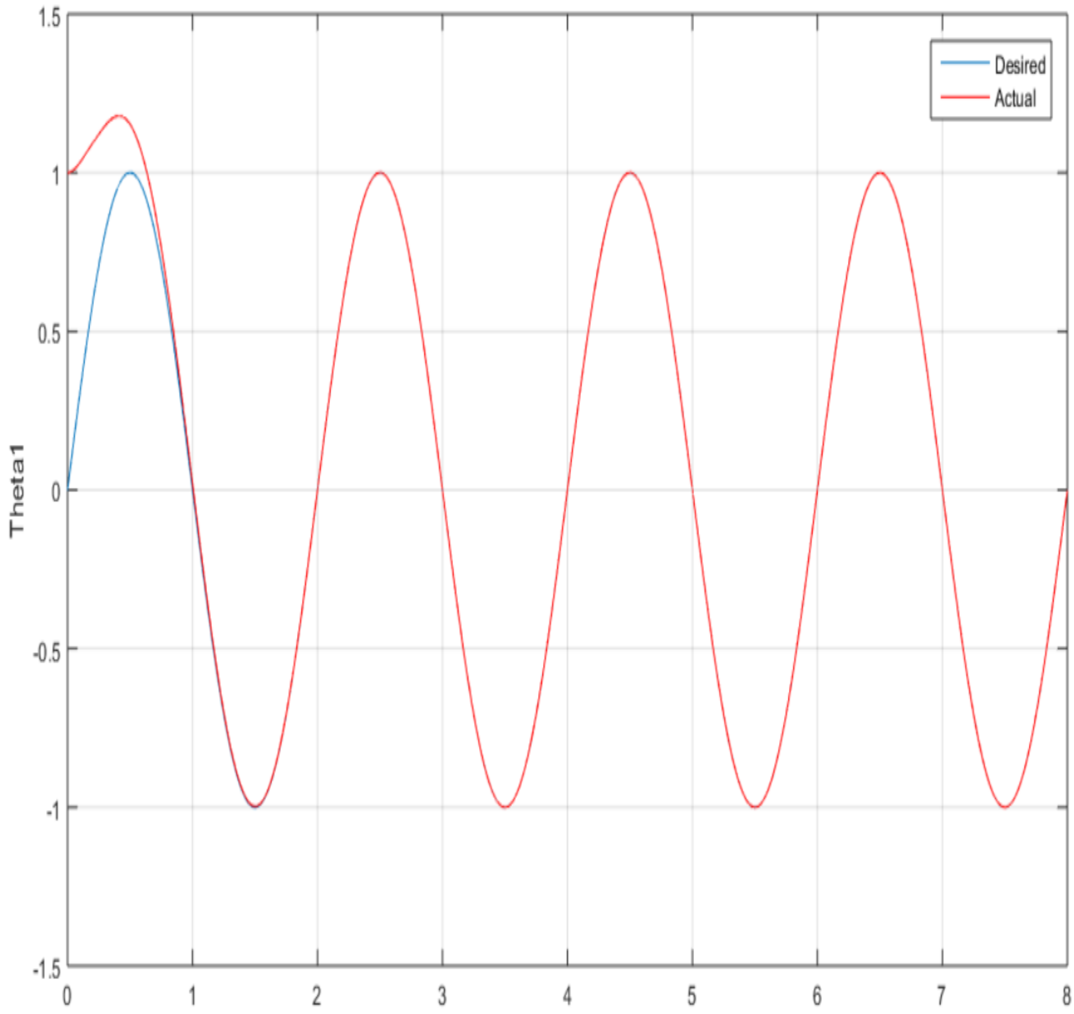


Figure 5.1 Desired and actual trajectory for link 1(θ_1) using SMC

From the figure 5.1, we can clearly see that after implementing the sliding mode control, the reference and the actual trajectory for link 1 overlap each other. The physical parameters for two link robot manipulator is given in table 5.1. the masses and the length of the link are tabulated.

Table 5.1 Physical Parameters for two link robot manipulator

Links	Masses(kg)	Length(m)
Link 1	1	1
Link 2	1	1

Parameters for position tracking of link 1 and link 2 of two link robot manipulator using SMC is given in table 5.1.

Table 5.2 Parameters for position of link 1 and link 2

LINKS		DELAY TIME(sec)	RISE TIME (sec)	PEAK TIME (sec)	PEAK OVERSHOOT %
LINK 1	ACTUAL	0.02134	0.1524	0.4113	17.9
	DESIRED	0.03665	0.3618	0.5031	0
LINK 2	ACTUAL	0.1969	0.7317	0.9231	14.53
	DESIRED	0.2018	0.8579	0.9978	0

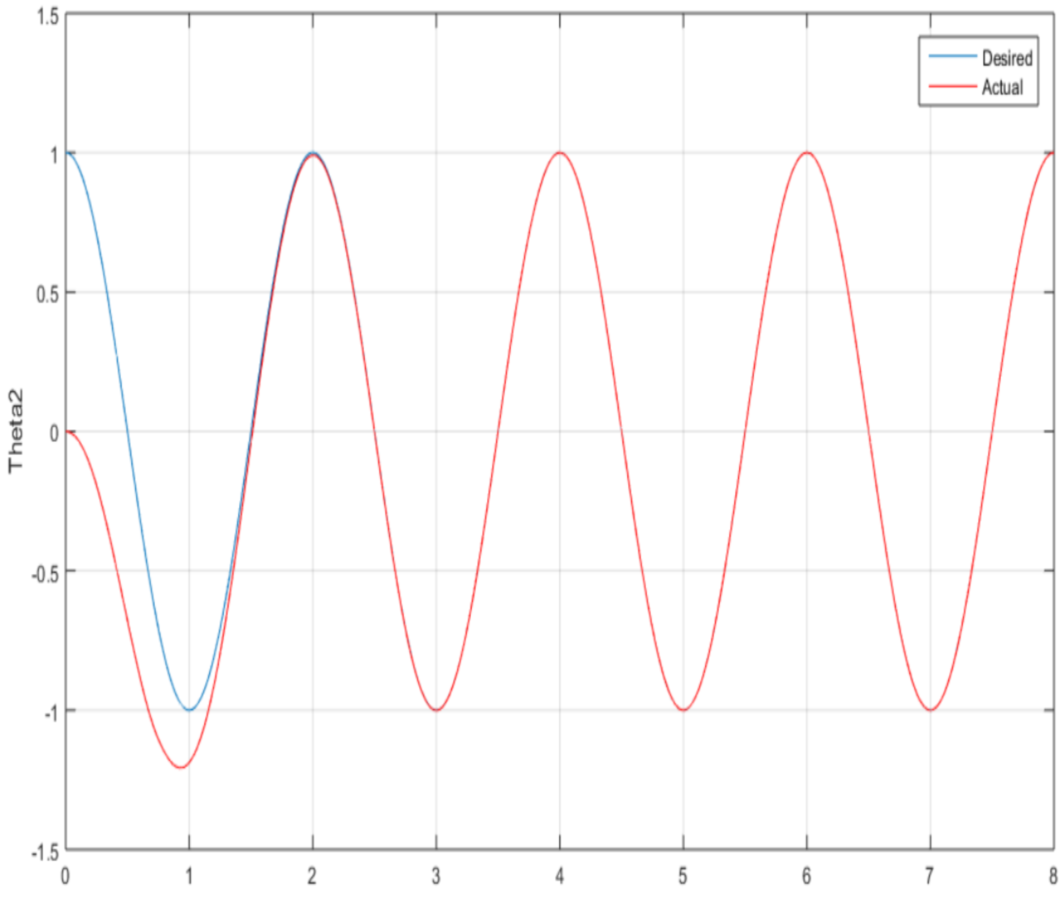


Figure 5.2 Desired and actual trajectory for link 2 (θ_2)using SMC

Figure 5.2 shows the actual and desired trajectory for the location of link 2 of robot manipulator. Here also, the actual and the desired trajectory for link 2 of robot arm coincides whose parameters are given in the table 5.2.

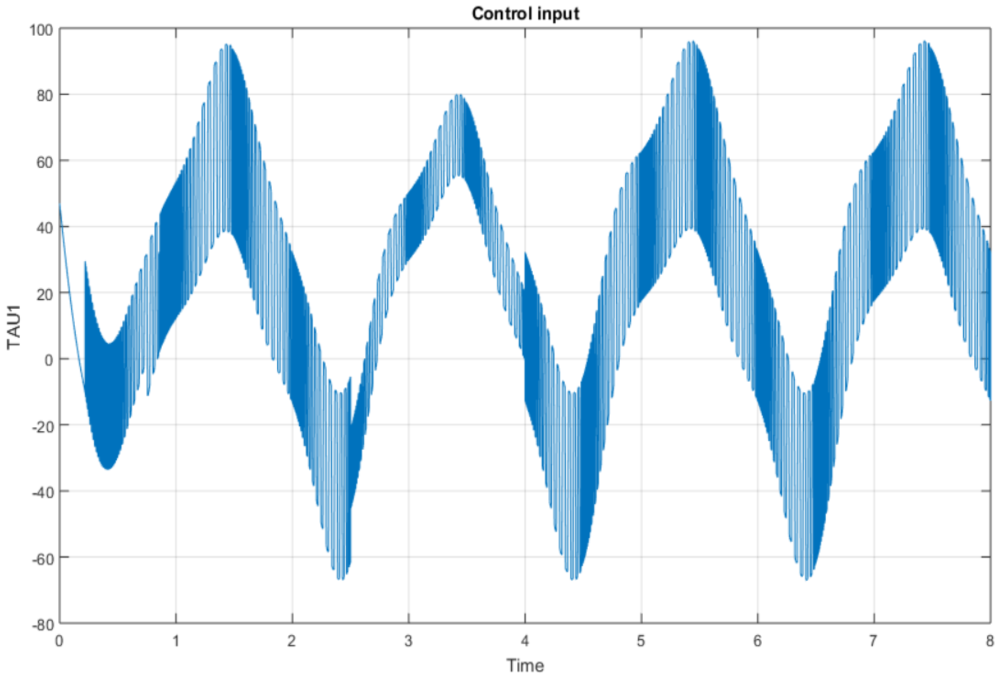


Figure 5.3 Torque of θ_1 using SMC

Torque of θ_1 using SMC for the robot manipulator is given in the figure 5.3. the control signal is depicted in this figure in the form of torque.

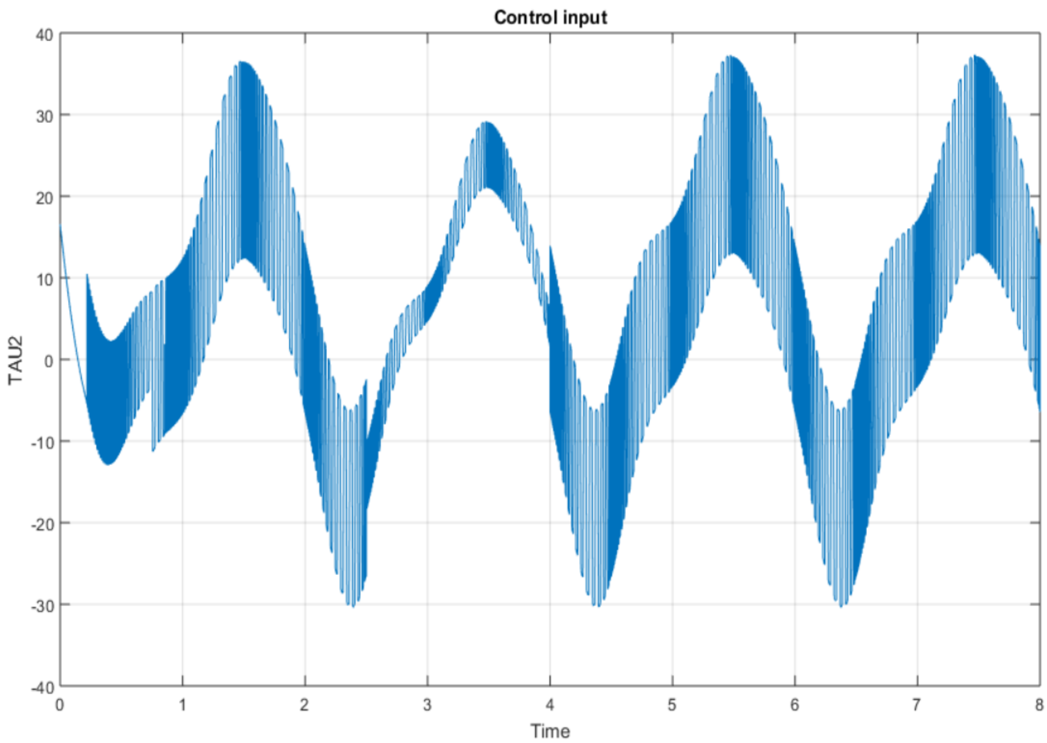


Figure 5.4 Torque of θ_2 using SMC

Figure 5.4 represents the torque of θ_2 using SMC for the link 2 of two link robot manipulator.

5.1.2 RBF NETWORK ADAPTIVE SLIDING MODE CONTROL FOR TWO LINK ROBOT MANIPULATOR

Table 5.3 Physical Parameters for two link robot manipulator

Links	Masses(kg)	Length(m)
Link 1	1	1
Link 2	1	1

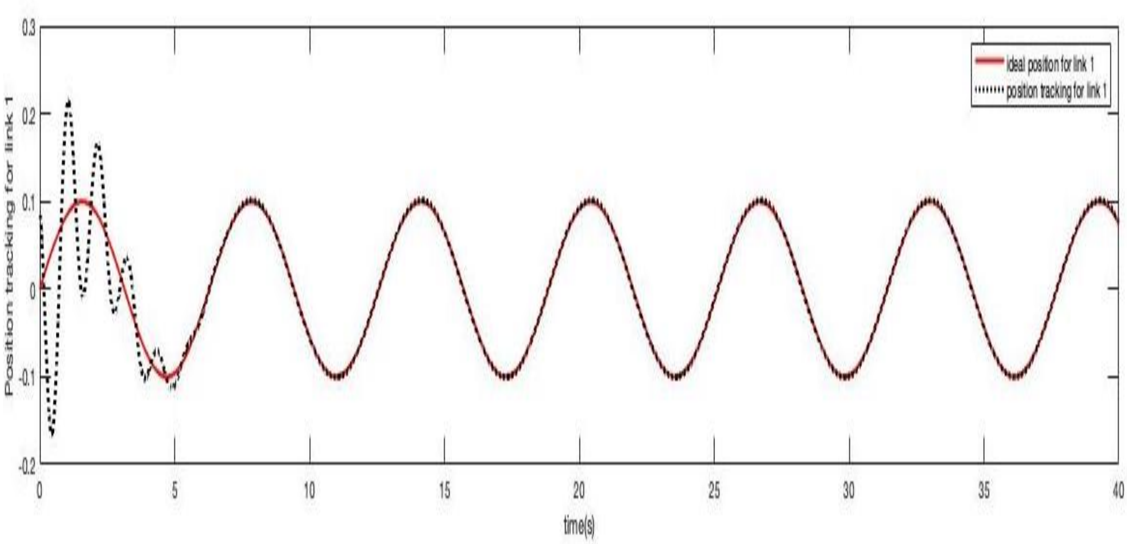


Figure 5.5 Desired and actual trajectory for link 1(θ_1) using RBFNN-SMC

From the figure 5.5, we can clearly see that after implementing the RBF sliding mode control, the reference and the actual trajectory for link 1 overlap each other. The RBF which is selected has b=0.20.

Parameters for position tracking of link 1 and link 2 of two link robot manipulator using RBFNN-SMC is given in table 5.4.

Table 5.4 Parameters for position of link 1 and link 2

LINKS		DELAY TIME(sec)	RISE TIME (sec)	PEAK TIME (sec)	PEAK OVERSHOOT %
LINK 1	ACTUAL	0.01874	0.1224	0.3216	14.18
	DESIRED	0.02265	0.3103	0.3031	0
LINK 2	ACTUAL	0.1472	0.6283	0.8163	12.44
	DESIRED	0.1524	0.7152	0.8816	0

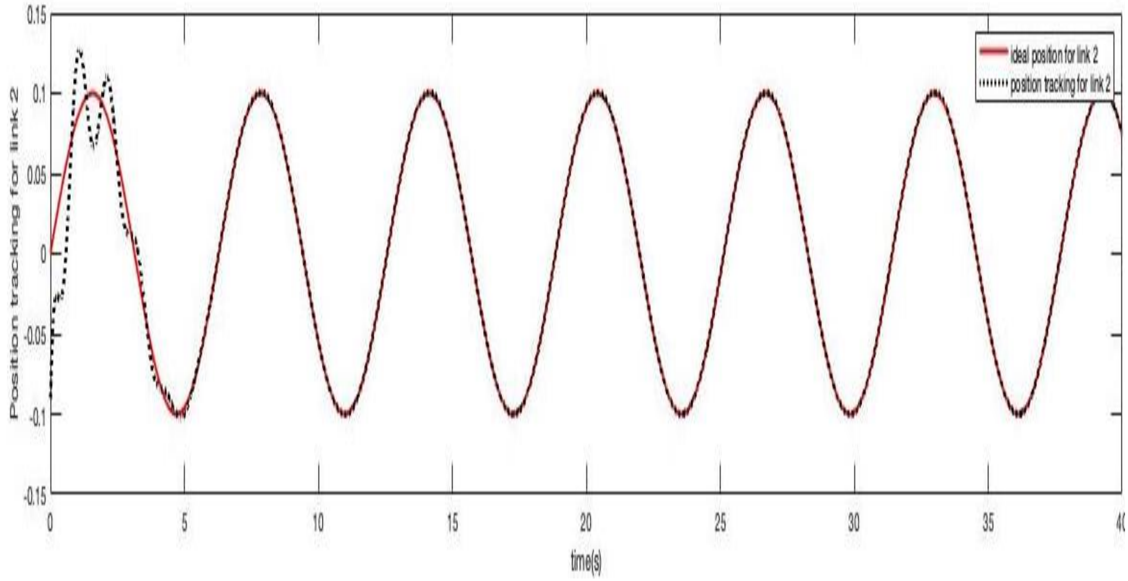


Figure 5.6 Desired and actual trajectory for link 2 (θ_2)using RBFNN-SMC

The trajectories of θ_2 i.e. the link two of robot manipulator using RBFNN and SMC is given in the figure 5.6. The desired and the actual trajectory overlaps giving the minimal tracking error.

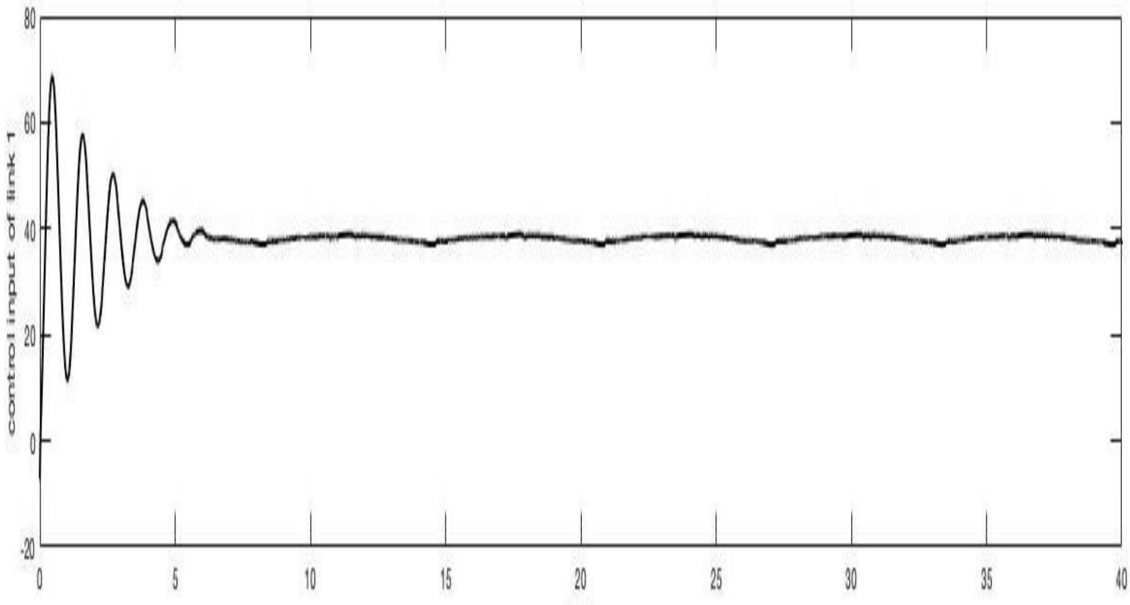


Figure 5.7 Control Input (torque) of θ_1 of link 1 using RBFNN-SMC

Figure 5.7 represents the control input in terms of torque of link 1 of two link robot manipulator. We have taken two control laws for adaptive RBF based sliding mode control and depicts it in the form of torques for both link. But for this, the parameters for the gauss function of RBF neural network has to be selected precisely otherwise it will be very difficult to map the gauss function and hence initial weight matrix is selected to be 0.

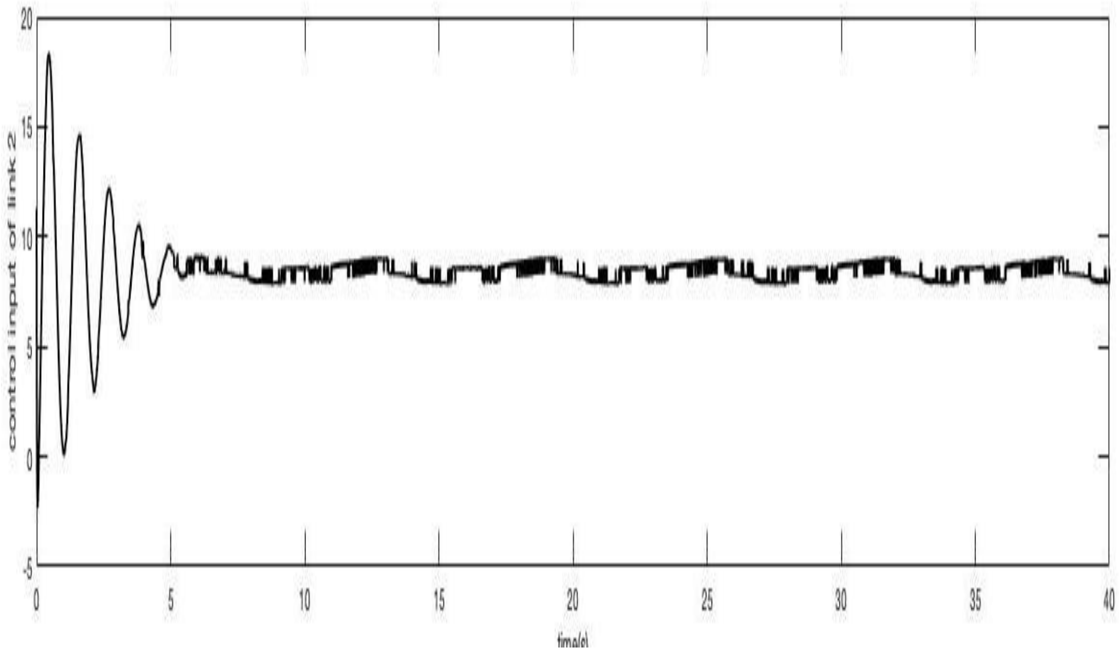


Figure 5.8 Control Input (torque) of θ_2 using RBFNN-SMC

Figure 5.8 represents the control input in terms of torque of link 1 of two link robot manipulator.

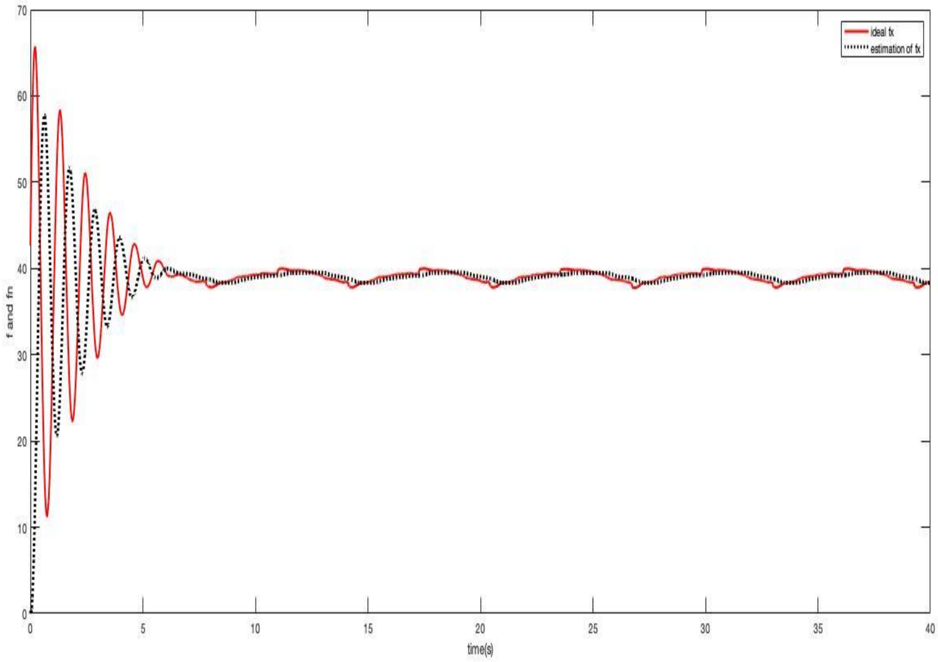


Figure 5.9 ||f(x)|| and $||\widehat{f(x)}||$ of link 1 and link 2

Figure 5.9 shows the approximation of f(x) done by the RBF neural network. $||\widehat{f(x)}||$ represents the output of the radial basis function network. This approximation is carried out by taking different weights value for RBF gaussian function and then the graph is plotted between the function and the function approximated value for the radial basis function.

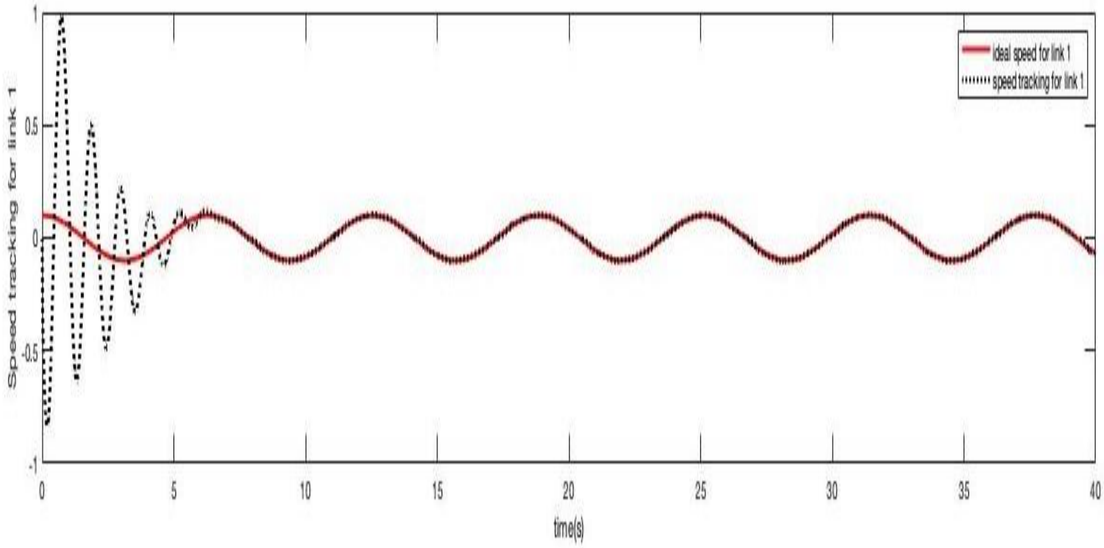


Figure 5.10 Velocity tracking of link 1

The figure 5.10 shows the speed tracking of link 1 of two link robot manipulator. This clearly shows that the link speed is much fast in RBFNN-SMC.

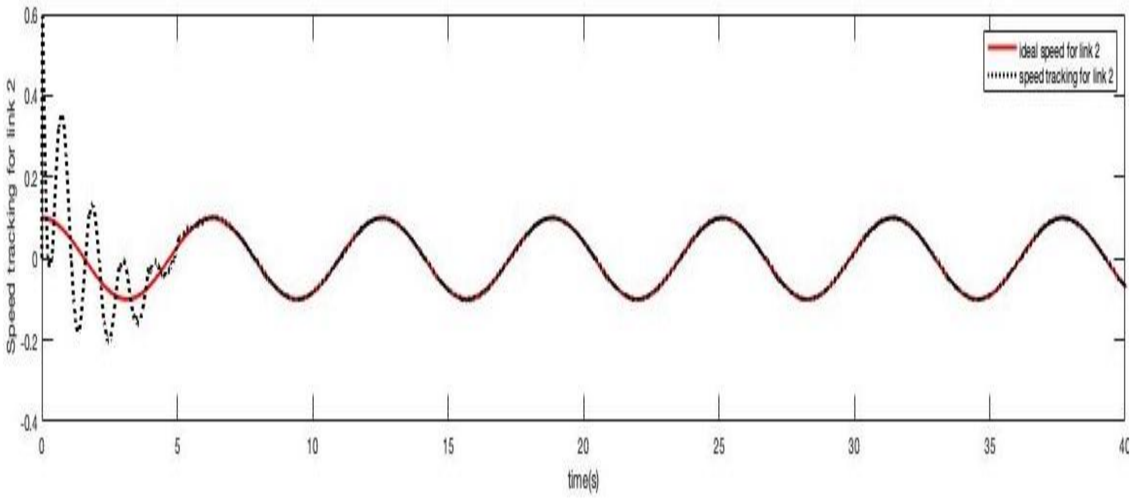


Figure 5.11 Velocity tracking of link 2

The figure 5.11 shows the speed tracking of link 2 of two link robot manipulator. This clearly shows that the link speed is much fast in RBFNN-SMC.

5.1.3 FEEDBACK LINERISATION FOR TWO LINK ROBOT MANIPULATOR

The computed torque controller is shown in Figure 5.12. Simulation of this controller for a two link robot arm (m_1=1kg, m_2=1kg, l_1=1m, l_2=1m, g= 9.8kgm/s^2 , $\theta_{d1} = \sin(\pi t)$, $\theta_{d2} = cos(\pi t)$)was done using MATLAB and the tracking performances are given below. This controller possess the inner linearization loop and the outer tracking loop. It make use of feedback loop to give the system a linear input-output dynamics. If functions were not available, we go for NN's as shown above. The results were already shown earlier for RBF-NN SMC for robot manipulator.

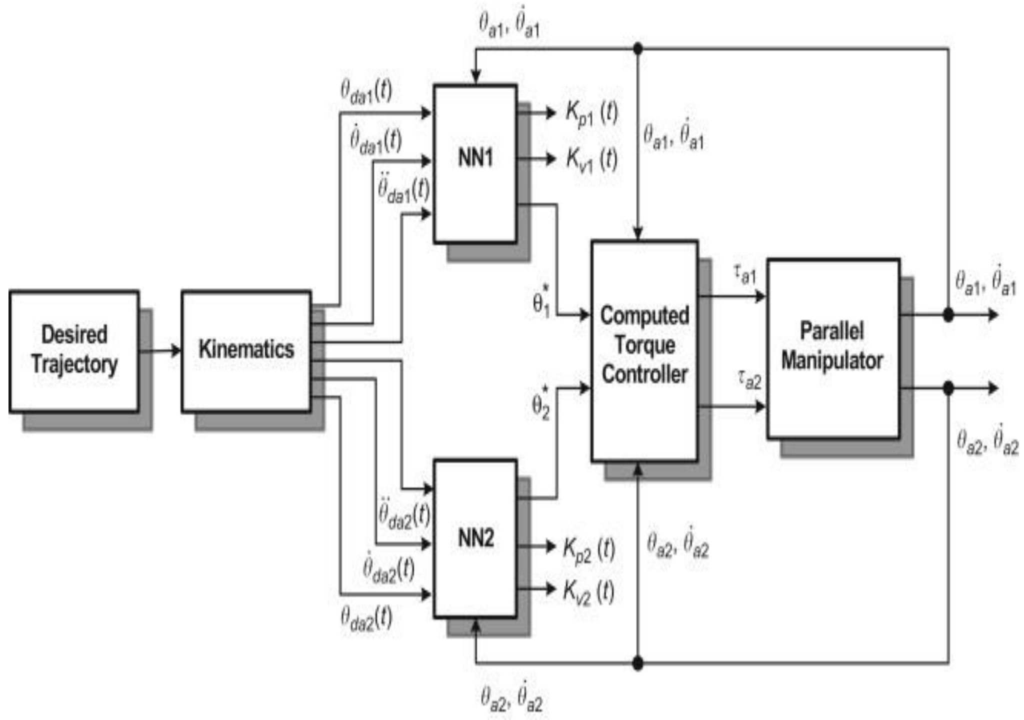


Figure 5.12 Computed torque controller

Figure 5.12 shows the computed torque controller consisting of two neural networks. Now, the desired trajectories are plotted for this controller.

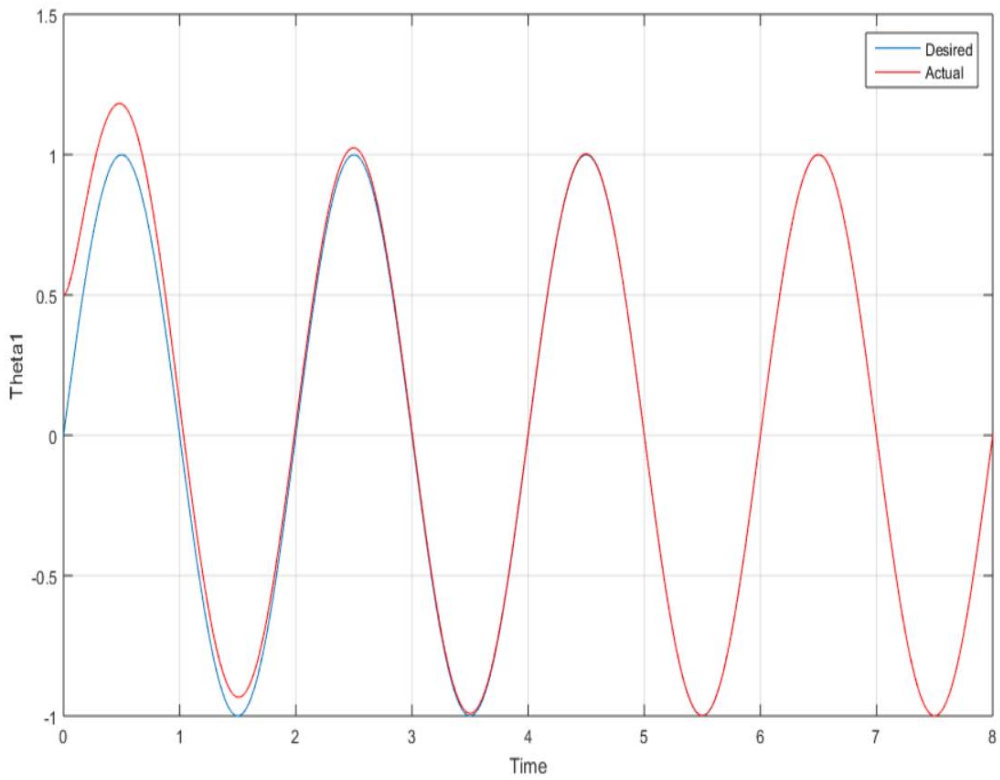


Figure 5.13 Desired and actual trajectory for link $1(\theta_1)$ using feedback linearization

From the figure 5.13, we can clearly see that after implementing the feedback linearization technique, the reference and the actual trajectory for link 1 overlap each other. The response characteristics for two link robot manipulator is given in table 5.5.

Table 5.5 Parameters for position of link 1 and link 2

LINKS		DELAY TIME(sec)	RISE TIME (sec)	PEAK TIME (sec)
LINK 1	**ACTUAL**	**0.000124**	**0.3241**	**0.4705**
	DESIRED	0.03036	0.3448	0.4901
LINK 2	ACTUAL	0.0000241	0.8105	0.9803
	DESIRED	0.2018	0.8401	1.006

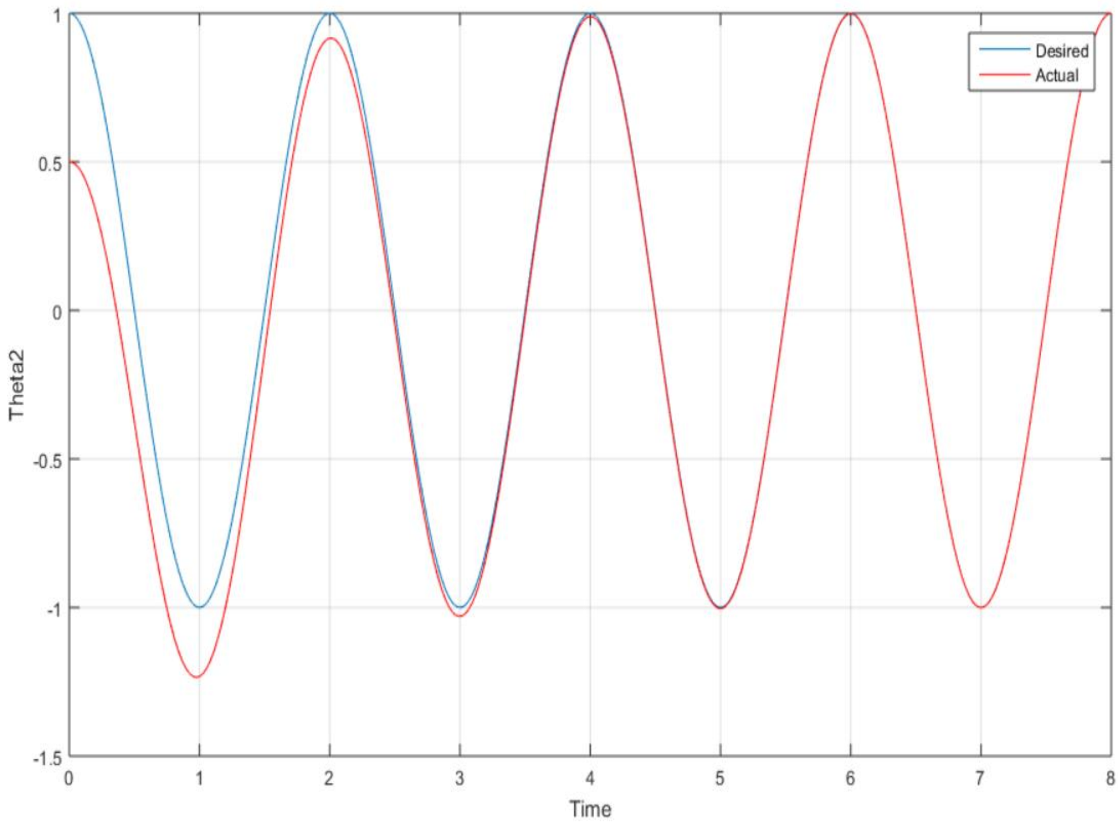


Figure 5.14 Desired and actual trajectory for link 2 (θ_2)using feedback linearization

Figure 5.14 shows the actual and desired trajectory for the location of link 2 of robot manipulator. Here also, the actual and the desired trajectory for link 2 of robot arm coincides with certain tracking error using feedback linearization technique. The response characteristics are already tabulated above. Now, the comparisons can easily be made out from these given approaches to deduce the best control techniques for tracking the trajectories of underactuated system. The error for both the link has been plotted.

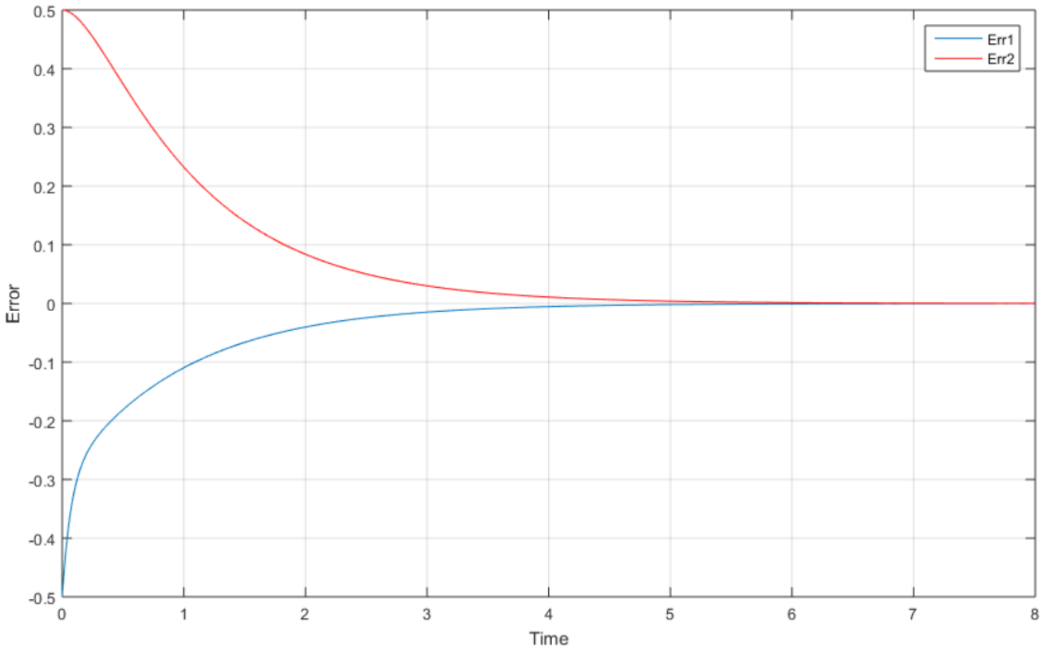


Figure 5.15 Representation of tracking error for link 1 and link 2

Figure 5.15 shows the tracking error for link 1 as well as link 2 for two link robot manipulator.

5.2 INVERTED PENDULUM

5.2.1 SLIDING MODE CONTROL FOR THE INVERTED PENDULUM WITH SMC

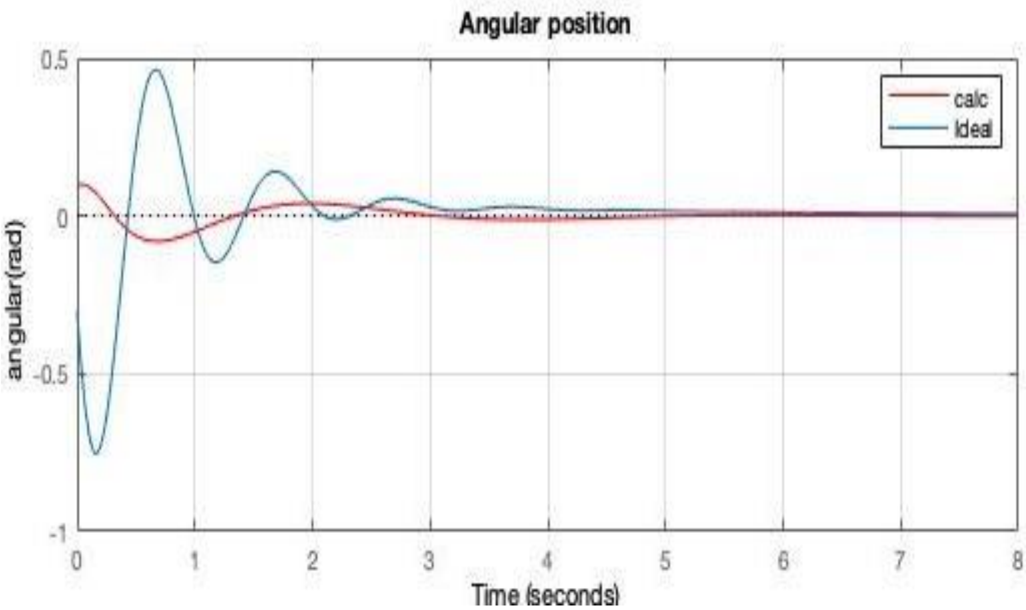


Figure 5.16 Angle tracking for one link inverted pendulum

Figure 5.16 shows the deviation of angle of one link inverted pendulum model. The controller settled the angle to the desired location.

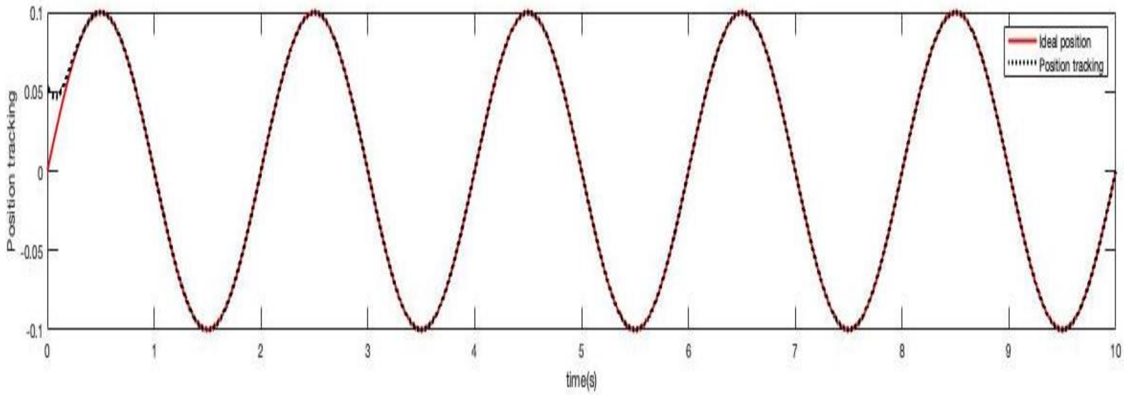


Figure 5.17 Position tracking

Figure 5.17 shows the position of the cart which is moving horizontally to make the pendulum stable.

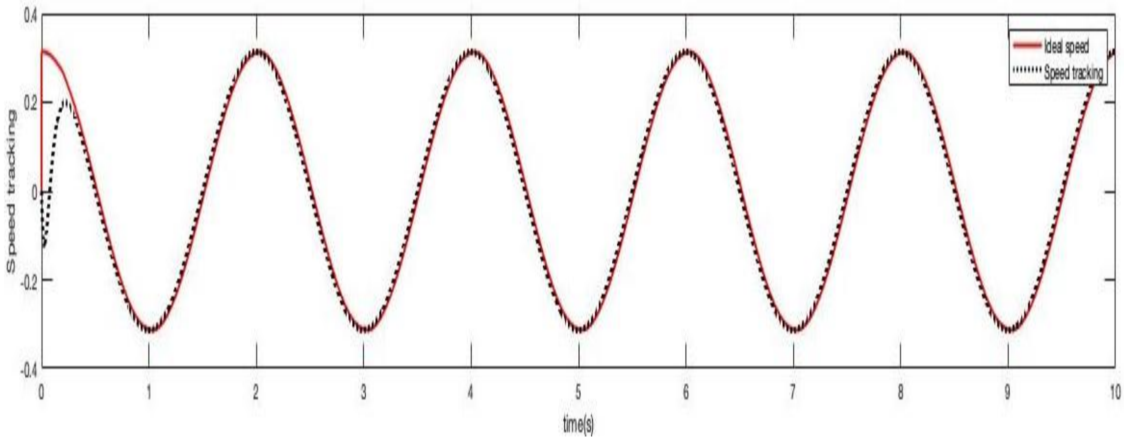


Figure 5.18 Speed tracking

Figure 5.18 shows the cart velocity, and it can be clearly seen that the ideal and the actual trajectories coincides each other with less tracking error.

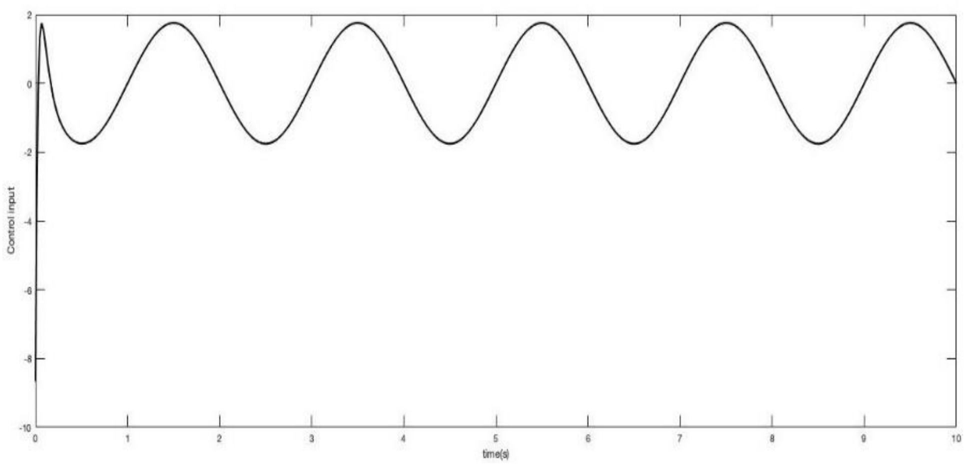


Figure 5.19 Control input

Figure 5.19 shows the control input taken for the sliding mode controller to evaluate the angle deviation of one link inverted pendulum model.

Table 5.6 Parameters for SMC for one link inverted pendulum model using sliding mode

Parameters	DELAY TIME(sec)	RISE TIME (sec)	PEAK TIME (sec)	SETTLING TIME(sec)	MAXIMUM OVERSHOOT
Position	1.37	1.71	2.0	5.8	1.1
Velocity	0.5	0.63	0.85	4.18	5.41

5.2.2 RESPONSE FOR THE NONLINEAR MODEL WITH RBF SLIDING MODE CONTROLLER

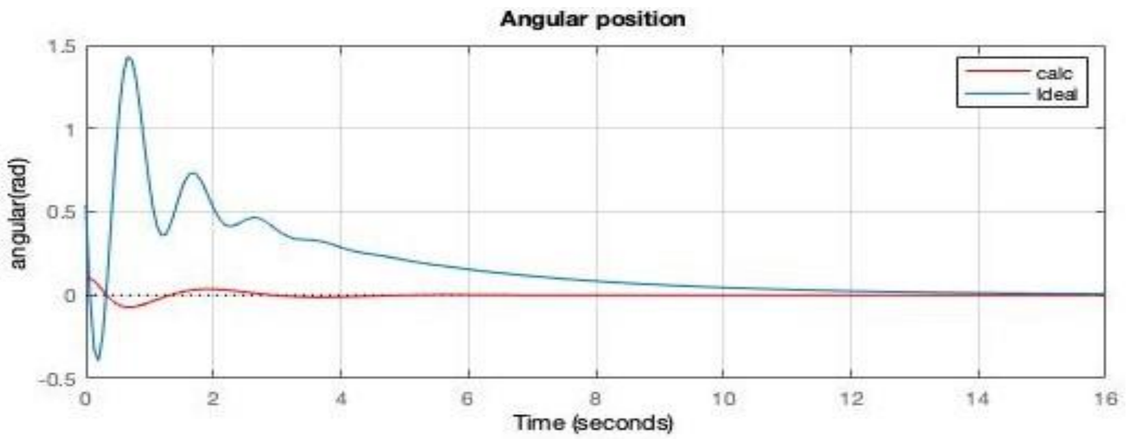


Figure 5.20 Angle tracking for one link inverted pendulum

Figure 5.20 shows the tracking of pendulum angle using RBF sliding mode control and it can be clearly seen that the angle settles at zero with very less tracking error.

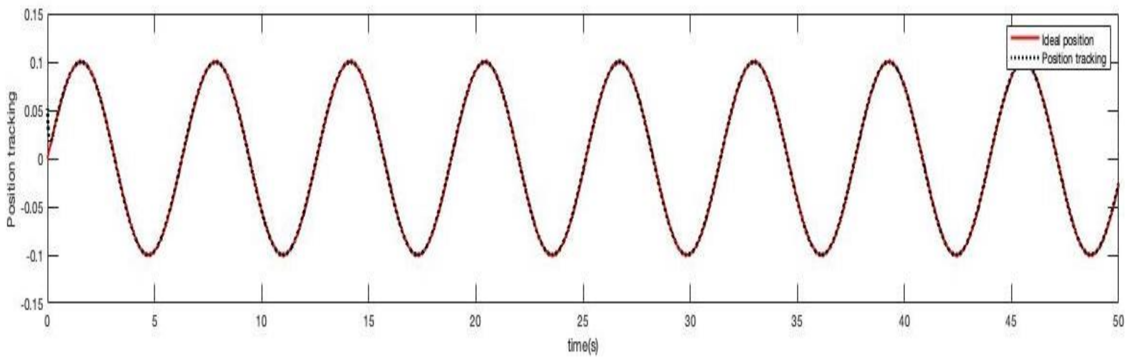


Figure 5.21 Position tracking

Figure 5.21 shows the position of the cart which is moving horizontally to make the pendulum stable using RBF neural network sliding mode control.

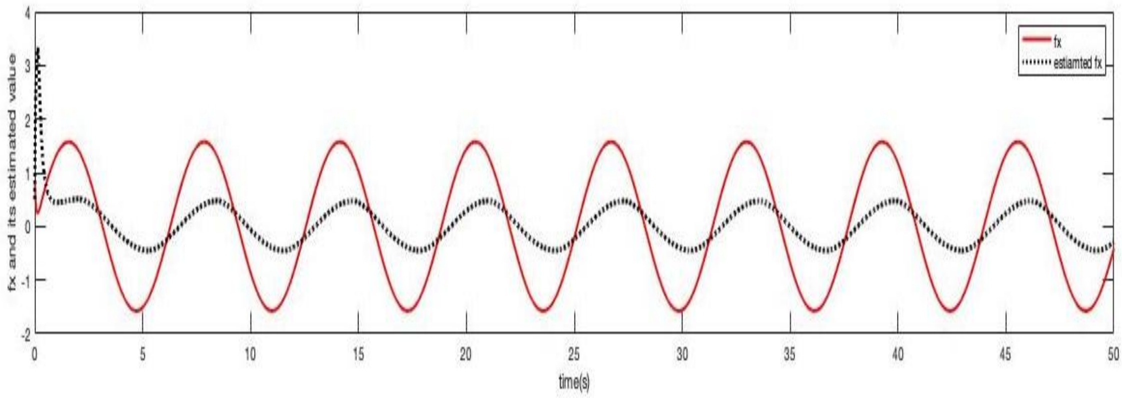


Figure 5.22 Function approximation using RBF

Figure 5.22 shows function approximation since the desired trajectory cannot be remain same and hence by putting several values for the function we get the tracking performance error.

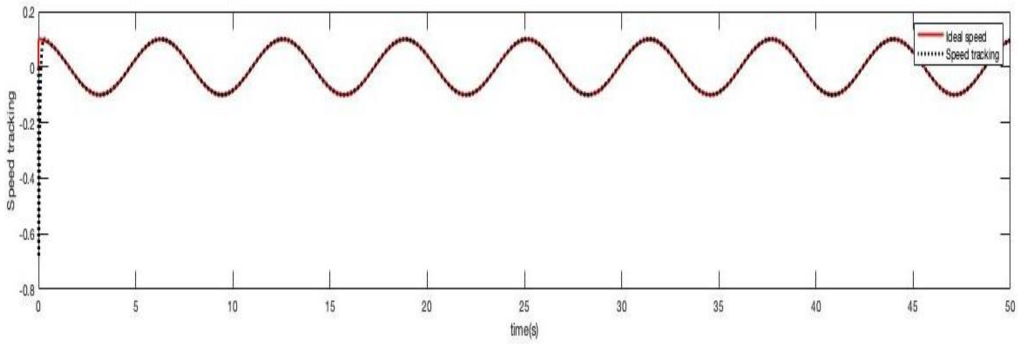


Figure 5.23 Speed tracking of cart

Figure 5.23 shows the speed tracking of the cart on which the pendulum is mounted. The error is very less between the ideal and the calculated trajectory.

The parameters for the inverted pendulum model with RBF sliding mode control is given in table 5.7

Table 5.7 Parameters for RBFNN-SMC based one link inverted pendulum

Parameters	DELAY TIME(sec)	RISE TIME (sec)	PEAK TIME (sec)	SETTLING TIME(sec)	MAXIMUM OVERSHOOT
Position	1.37	1.73	2.09	5.46	1.1
Velocity	0.49	0.65	0.88	4.35	5.20

CHAPTER 6
CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

The thesis started with the study of non linear control system and analysis of nonlinearities such as saturation, deadzone, backlash and on-off. These nonlinearities have drastically affects the stability as well as the performance of the nonlinear control systems so the study of these systems have been done by discussing various nonlinear analysis methods like phase plane analysis, Lyapunov theory and describing functions. The controller is then designed for them which are applicable to different classes of nonlinear system. Trial and error method, feedback linearization method, variable structure sliding mode control, adaptive control, gain scheduling and intelligent control are some of the methods which are extensively studied for designing the controller. After this, two underactuated mechanical systems i.e. two link robot manipulator and one link inverted pendulum has been taken into consideration and the dynamic model has been derived. The mathematical model of two DOF robot manipulator is designed in the form of differential equations and then feedback linearization technique is applied to obtain the plant model and two control techniques that is adaptive sliding mode control and RBF neural network based sliding mode control have been successfully applied to improve the stability of this system. A comparative study of the parameters for the links of robot manipulator has been carried out.

The second system that is an inverted pendulum system is designed, the system equations has been derived which provides the physical model on which sliding mode control technique is implemented. The RBF based neural network compensation technique is also been implemented and it can be inferenced that tracking performance of the inverted pendulum model is improved. A comparative study of the parameters for one link inverted pendulum model is done at the end

6.2 FUTURE SCOPE

In this project sliding mode controller with and without RBF neural network is implemented on two link robot manipulator, this work can be extended further by applying nature inspired algorithms in place of soft computing techniques as these algorithms provide more better compensation and produce better results, some of such algorithms are particle swarm optimization technique, cuckoo search algorithm or genetic algorithm. The research work can also be extended by considering more complex model such as four link or six link robot manipulator for analysis and compensation using these approaches.

Similar for the case of inverted pendulum, one can apply other types of neural networks such as Hopfield, perceptron etc. for stability or can also apply the above stated algorithms to reduce the errors and deviations further which occurred in the approach used in this project.

REFERENCES

[1] Jin Seok Noh ; Geun Hyeong Lee ; Ho Jin Choi ; Seul Jung .Robust cont rol of a mobile inverted pendulum robot using a RBF neural network co ntroller.Robotics and Biomimetics, 2009 , pp: 1932 – 1937.

[2] Utkin V.I., “Variable structure systems with sliding modes”, IEEE Transactions on Automatic Control, vol. 22, no. 2, pp. 212-22, 1977.

[3]V. I. Utkin, Sliding Modes and Their Application in Variable Structure Systems. Moscow: MIR, 1978.

[4] Slotine J. and Sastry S., “Tracking control of nonlinear system using sliding surfaces with application to robot manipulators”, International Journal of Control, vol. 38, pp.465–492, 1983.

[5] J.J.E.Slotine, W. Li: On the Adaptive Control of Robot Manipu -lators,The Int. J. Robotics Reseach,Vol.6,No.3, 49-59, 1987

[6] Slotine J.E. and Li W.P., “On the adaptive control of robot manipula-tors”, Interational Journal of Robotics Research, vol. 6, no. 3, pp.49–59, 1987

[7] J.J.E.Slotine, W. Li: Adaptive Manipulators Control: A Case Study, IEEE Trans. Auto. Contr.,Vol.33,No.11,995 -1003, 1988

[8] R. A. DeCarlo, S. H. Zak, and G. P. Matthews, “Variable Structure Control of Nonlinear Multivariate Systems: A Tutorial,” Proceedings of the IEEE, Vol. 76, No. 3, pp. 212-232, March 1988.

[9] Astrom K.J. and Wittenmark B., “Adaptive Control”, 2nd edn. Addison-Wesley, NY 1994,1989.

[10] V. I. Utkin and S. Drakunov, “On Discrete-time Sliding Mode Control,” Proceedings of IFAC Symposium on Nonlinear Control Systems (NOLCOS), (Capri, Italy), pp. 484-489, 1989.

[11] Hartman EJ, Keeler JD, Kowalski JM. Layered neural networks with Gaussian hidden units as universal approximations. Neural computation, 1990, 2(2): 210 _ 215

[12] Kanellakopoulos I, Kokotovic PV, Morse AS (1991) Systematic design of adaptive controllersfor feedback linearizable systems. IEEE Trans Autom Control 36(11):1241–1253

[13] Park J, Sandberg IW. Universal approximation using radial-basis-function networks. Neural computation, 1991,3: 246 _ 257

[14] Narendra KS (1991) Adaptive control using neural networks. In: Neural networks for control.MIT Press, Cambridge, MA, pp 115–142

[15] Ozaki T, Suzuki T, Furuhashi T, Okuma S, Ushikawa Y (1991) Trajectory control of roboticmanipulators. IEEE Trans Ind Electron 38(3):195–202

[16] Chen FC, Khalil HK (1992) Adaptive control of nonlinear systems using neural networks. Int JControl 55(6):1299–1317

[17] Kokotovic PV (1992) The joy of feedback: nonlinear and adaptive. IEEE Control Syst Mag 12(3):7–17

[18] B. L. Chin, "Analysis, Design, Construction, and Control of an Inverted Pendulum," Bachelor's Thesis, Tulane University, April 1992

[19] T. D. Ledgerwood and E. A. Misawa, "Controllability and Nonlinear Control of a Rotational Inverted Pendulum," Advances in Robust and Nonlinear Control Sysytems, DSC-Vol. 43, ASME, pp. 81-88, 1992.

[20] Hung, J. Y., Gao, W. B., Hung, J. C., " Variable structure control: a survey", IEEE Transactions on Industrial Electronics, 40, 2-22, 1993

[21] L.L.Whitcomb, A.A.Rizzi: Comparative Experiments with a New Adaptive Controller for Robot Arms ,IEEE Trans. Robotics and Automation , Vol.9, No.1, 59 -69, 1993

[22] K.M.Koo, J.H.Kim: Robust Control of Robot Manipulators with Parameters Uncertainty, IEEE Trans.Auto.Contr., Vol.39, No.6, 1230-1233,1994

[23] E. A. Misawa, M. S. Arrington, and T. D. Ledgerwood, "Rotational Inverted Pendulum: A New Control Experiment ," Proceedings of the 1995 American Control Conference, (Seattle, WA), pp. 29-33, June 1995.

[24] P. G. Grossimon, "Inverted Pendulum Control: A Nonlinear Simulation Using SIMULINK," Bachelor's Thesis, Tulane University, April 1995

[25] Lewis FL, Liu K, Yesildirek A. Neural Net Robot Controller with Guaranteed Tracking Performance. IEEE Transactions on Neural Networks, 1995, 6(3): 703 _ 715

[26] Krstic M, Kanellakopoulos I, Kokotovic P (1995) Nonlinear and adaptive control design. Wiley, New York

[27] J.H.Yang ,F.L.Lian and L.C.Fu: Nonlinear Adaptive Control for Flexible—Link Manipulators , IEEE Robotics and Automation˙ Vol.13,No.1,140-148,1997

[28] Ge SS, Lee TH, Harris CJ. Adaptive Neural Network Control of Robotic Manipulators. World Scientific, London, 1998

[29] Utkin V.I., Guldner J. and Shi J., "Sliding Mode in Control in Electrome-chanical Systems", Taylor and Francis, London, 1999.

[30] Zhang Y, Peng PY, Jiang ZP (2000) Stable neural controller design for unknown nonlinear systems using backstepping. IEEE Trans Neural Netw 11(6):1347–1360

[31]Bartolini G., Ferrara A., Usai E. and Utkin V.I., "On multi-input chattering-free second order sliding mode control", IEEE Transactions on Automatic Control, vol. 45, pp.1711–1717, 2000.

[32] Kwan CM, Lewis FL (2000) Robust backstepping control of nonlinear systems using neural networks. IEEE Trans Syst Man Cybern (Part A) 30(6):753–766

[33]Kwan CM, Lewis FL (2000) Robust backstepping control of induction motors using neural networks. IEEE Trans Neural Netw 11(5):1178–1187

[34] Zhang T, Ge SS, Hang CC (2000) Adaptive neural network control for strict-feedback nonlinear systems using backstepping design. Automatica 36(12):1835–1846

[35] Levant A., "Universal SISO sliding-mode controllers with finitetime convergence", IEEE Transactions on Automatic Control, vol. 49, pp.1447–1451, 2001.

[36] Parra-Vega V., Arimoto S., Yun-Hui L., Hirzinger G. and Akella P., "Dynamic sliding PID control for tracking of robot manipulators: theory and experiments", IEEE Transactions on Robotics and Automation, vol. 19, pp.967–976, 2003.

[37] Choi JY, Farrell JA (2001) Adaptive observer backstepping control using neural networks. IEEE Trans Neural Netw 12(5):1103–1112

[38]Ge SS, Wang C (2002) Direct adaptive NN control of a class of nonlinear systems. IEEE TransNeural Netw 13(1):214–221

[39]Ge SS, Wang C (2002) Adaptive NN control of uncertain nonlinear pure-feedback systems.Automatica 38(4):671–682

[40] Ognjen K, Nitin S, Lewis FL, Chiman MK (2003) Design and implementation of industrial neural network controller using backstepping. IEEE Trans Ind Electron 50(1):193–201

[41]Ge SS, Wang C (2004) Adaptive neural control of uncertain MIMO nonlinear systems. IEEE Trans Neural Netw 15(3):674–692

[42]Huang AC, Chen YC (2004) Adaptive sliding control for single-link flexible joint robot with mismatched uncertainties. IEEE Trans Control Syst Technol 12(5):770–775

[43] Boiko I. and Fridman L., "Analysis of chattering in continuous sliding-mode controllers", IEEE Transaction on Automatic Control, vol. 50, pp.1442–1446, 2005.

[44] Boiko I., Fridman L., Pisano A. and Usai E., "Analysis of chattering in systems with second order sliding modes", IEEE Transaction on Automatic Control, vol. 50, pp.1442–1446, 2005.

[45] Jin Seok Noh ; Geun Hyeong Lee ; Seul Jung .Motion control of a mobil e pendulum system usingneural network. Advanced Motion Control, 2 008 , pp: 450 - 454

[46] Jiang-Jiang Wang ; Chun-Fa Zhang ; You-Yin Jing .Self-adaptive RB F neural network PID control in exhaust temperature of micro gas turbin e. Machine Learning and Cybernetics, 2008 ,pp: 2131 - 2136 .

[47] Bingyu Sun ; Jingyi Zhao ; Zhiyong Wang ; Chengyu Yao ;Rui Guo .Co ntrol strategy and experimental research on girder hoister steering syste m based on RBF neuralnetwork PID. Electronic Measurement & Instru ments, 2009 ,pp: 3-210 - 3-214.

[48] Zhixiang T., Hongtao WU. and Chun F., "hierarchical adaptive back-stepping sliding mode control for underactuated space robot", 2nd InternationalAsia Conferenceon Informaticsin Control, Automationand Robotics, 2010.

[49] Zeyu Li ; Jiangqiang Hu ; Xingxing Huo .PID control based on RBF neu ral network for ship steering. Information and Communication Technolo gies (WICT), 2012 , Page(s): 1076 – 1080.

[50]Olejnik, P. and Awrejcewicz, J., "Low-speed voltage-input tracking control of a dc-motor numerically modelled by a dynamical system with stick-slip friction", Differential Equations and Dynamical Systems, 20(5):Online First, 2012.

[51] Utkin V.I. and Poznyak A.S., "Adaptive sliding mode control with appli-cation to super-twist algorithm: Equivalent control method", Automatica, vol. 49, pp.39–47, 2013.

[52] Lizarralde F., Leite A.C., Hsu L., and Costa R., "Adaptive visual servoing scheme free of image velocity measurement for uncertain robot manipulators", Automatica, vol. 49, no. 5, pp.1304–1309, 2013

[53] Sassi A., Ghorbel C. and Abdelkrim A., "Multiple model adaptive control of complex systems", International Conference on Control, Engineering and Information Technology, pp.229–235, Sousse, Tunisia, March 2014

[54] Sassi A. and Abdelkrim A., "Comparative study of trajectory tracking control using sliding mode based on backstepping and feedback lin-earization", World Symposium on Mechatronics Engineering and Applied Physics, Sousse, Tunisia, June 2015.

Robot manipulator

ORIGINALITY REPORT

18%	9%	12%	14%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.engr.usask.ca Internet Source	1%
2	www.uta.edu Internet Source	1%
3	www.slideshare.net Internet Source	1%
4	www.ijetae.com Internet Source	1%
5	Submitted to Universiti Teknologi Petronas Student Paper	1%
6	Belunce, Andres, Vinicius Pandolfo, Hamid Roozbahani, and Heikki Handroos. "Novel Control Method for Overhead Crane's Load Stability", Procedia Engineering, 2015. Publication	1%
7	Jinkun Liu, Xinhua Wang. "Advanced Sliding Mode Control for Mechanical Systems", Springer Nature, 2011 Publication	1%

8	Submitted to Indian Institute of Technology, Madras Student Paper	<1%
9	Submitted to Asian Institute of Technology Student Paper	<1%
10	documents.mx Internet Source	<1%
11	eprints.utm.my Internet Source	<1%
12	Submitted to BITS, Pilani-Dubai Student Paper	<1%
13	Vehicle Dynamics, 2014. Publication	<1%
14	mafiadoc.com Internet Source	<1%
15	Radial Basis Function (RBF) Neural Network Control for Mechanical Systems, 2013. Publication	<1%
16	www.sdiarticle2.org Internet Source	<1%
17	Fei Liu. "Adaptive RBFNN Based Fuzzy Sliding Mode Control for Two Link Robot Manipulator", 2009 International Conference on Artificial Intelligence and Computational Intelligence,	<1%

	11/2009 Publication	
18	www.ukessays.com Internet Source	<1%
19	Submitted to DeVry University Onlline Student Paper	<1%
20	Submitted to Jawaharlal Nehru Technological University Student Paper	<1%
21	Submitted to Institute of Technology, Nirma University Student Paper	<1%
22	Lewis, . "Robot Dynamics", Automation and Control Engineering, 2003. Publication	<1%
23	Submitted to Universiti Teknologi Malaysia Student Paper	<1%
24	www.aoe.vt.edu Internet Source	<1%
25	Submitted to Higher Education Commission Pakistan Student Paper	<1%
26	Submitted to University of Western Sydney Student Paper	<1%
27	Submitted to University of Babylon	

Student Paper

<1%

28	Submitted to Universiti Teknikal Malaysia Melaka Student Paper	<1%
29	Submitted to iGroup Student Paper	<1%
30	www.scribd.com Internet Source	<1%
31	Reza N. Jazar. "Forward Kinematics", Theory of Applied Robotics, 2010 Publication	<1%
32	Pietrucha, Jozef, Krzysztof Sibilski, Anna Sibilska-Mroziewicz, Maria Zlocka, and Andrzej Zyluk. "Comparative Analysis of Wing Rock Control", 47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition, 2009. Publication	<1%
33	"Book of abstracts", 2009 American Control Conference, 06/2009 Publication	<1%
34	Submitted to Dundalk Institute of Technology Student Paper	<1%
35	Submitted to SASTRA University Student Paper	<1%

36 Siddhartha Bhattacharyya, Ujjwal Maulik. "Soft Computing for Image and Multimedia Data Processing", Springer Nature, 2013
Publication
<1%

37 M IVANESCU. "Control", Mechanical Engineer s Handbook, 2001
Publication
<1%

38 Wang Jianhua, Liu Lei, Zhang Fanghua, Gong Chunying, Ma Yiling. "Modeling and Analysis of Hysteretic Current Mode Control Inverter", 2009 Twenty-Fourth Annual IEEE Applied Power Electronics Conference and Exposition, 2009
Publication
<1%

39 Liu, Jinkun, and Xinhua Wang. "Neural Network Sliding Mode Control", Advanced Sliding Mode Control for Mechanical Systems, 2011.
Publication
<1%

40 Jinkun Liu. "Intelligent Control Design and MATLAB Simulation", Springer Nature, 2018
Publication
<1%

41 Submitted to Montana State University, Bozeman
Student Paper
<1%

42 Submitted to American University of the Middle

East
Student Paper <1%

43 Jemma J. Makrygiorgou, Antonio T. Alexandridis. "Applying fully stable cascaded sliding mode and PI anti-windup controllers on electric vehicles", 2017 25th Mediterranean Conference on Control and Automation (MED), 2017 <1%
Publication

44 Metin Demirtas, Yusuf Altun, Ayhan Istanbullu. "An educational virtual laboratory for sliding mode and PID control of inverted pendulum", 2008 11th International Conference on Optimization of Electrical and Electronic Equipment, 2008 <1%
Publication

45 Gerasimos Rigatos, Pierluigi Siano, Guilherme Raffo. "An H-infinity nonlinear control approach for multi-DOF robotic manipulators", IFAC-PapersOnLine, 2016 <1%
Publication

46 Gaya Prasad, Awadhesh Kumar. "A Comparison Between Sliding Mode Control and Feedback Linearization", 2018 5th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON), 2018 <1%

Publication

47	tr.scribd.com Internet Source	<1%
48	Valluru, Sudarshan K., Madhusudan Singh, and Nawratna Kumar. "Implementation of NARMA-L2 Neuro controller for speed regulation of series connected DC motor", 2012 IEEE 5th India International Conference on Power Electronics (IICPE), 2012. Publication	<1%
49	Submitted to St. Xavier's College Student Paper	<1%
50	F. Moldoveanu, R. Campeanu, D. Floroian. "Control strategy of a uncertain dynamical system: A sliding mode approach", 2007 International Aegean Conference on Electrical Machines and Power Electronics, 2007 Publication	<1%
51	Submitted to Feng Chia University Student Paper	<1%
52	Angueira, . "Equipment and Subsystem Technology Aspects: A Radio Link Designer Approach", Microwave Line of Sight Link Engineering Angueira/Line of Sight, 2012. Publication	<1%
53	R.V. Patel. "Modeling and control of a new	

	actuation mechanism for Interfacial Force Microscopy", 2008 American Control Conference, 06/2008 Publication	<1%
54	Submitted to Brunel University Student Paper	<1%
55	Rainer Palm, Hans Hellendoorn, Dimiter Driankov. "Model Based Fuzzy Control", Springer Nature, 1997 Publication	<1%
56	Submitted to The Hong Kong Polytechnic University Student Paper	<1%
57	Submitted to CSU Northridge Student Paper	<1%
58	Submitted to University of Newcastle upon Tyne Student Paper	<1%
59	Submitted to University of Strathclyde Student Paper	<1%
60	Park, C.-W., and Y.-W. Cho. "Robust fuzzy feedback linearisation controllers for Takagi-Sugeno fuzzy models with parametric uncertainties", IET Control Theory and Applications, 2007. Publication	<1%

61	www.ijifr.com Internet Source	<1%
62	Tat-Bao-Thien Nguyen, Teh-Lu Liao, Jun-Juh Yan. "Adaptive Sliding Mode Control of Chaos in Permanent Magnet Synchronous Motor via Fuzzy Neural Networks", Mathematical Problems in Engineering, 2014 Publication	<1%
63	Submitted to University of Technology Student Paper	<1%

Exclude quotes	Off	Exclude matches	< 14 words
Exclude bibliography	On		