ADDIS ABABA UNIVERSITY

ADDIS ABABA INSTITUTE OF TECHNOLOGY

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

Adaptive Radial Basis Function Neural Network Based Hierarchical Sliding Mode Controller for 2-Dimensional Double Pendulum Overhead Crane

A thesis submitted to School of Graduate Studies, Addis Ababa Institute of Technology, Addis Ababa University in partial fulfillment of the requirement for the Degree of Master of Science in Electrical Engineering (Control Engineering)

BY

Wosene Yirga Balcha

Advisor

Dr.Dereje Shiferaw

January,2024

Addis Ababa, Ethiopia

ADDIS ABABA UNIVERSITY

ADDIS ABABA INSTITUTE OF TECHNOLOGY

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

Adaptive Radial Basis Function Neural Network Based Hierarchical Sliding Mode Controller Design For 2-Dimensional Double Pendulum Overhead Crane

APPROVED BY BOARD OF EXAMINERS

Name	**Signature**	**Date**
--	---------------	-----------
(Dean, School of Graduate Committee)		
--	---------------	-----------
(Advisor)		
--	---------------	-----------
(Internal Examiner)		
--	---------------	-----------
(External Examiner))		

Declaration

The undersigned hereby states that the thesis is entirely original with no submission for credit toward any degree from any institution or university and that all references to materials utilized in the thesis have been duly noted.

Name	**Signature**
--	----------------------

Place: Addis Ababa Institute of Technology, AAiT
Addis Ababa University, AAU
Addis Ababa, Ethiopia.

Submitted in:January , 2024

This thesis has been submitted for examination with my approval as a university advisor.

Advisor	**Signature**
--	----------------------

Acknowledgments

My first thanks is always to almighty God for being with me,guide and strengthen me throughout my life.And then,I would like to thank you my advisor Dr.Dereje Shiferaw who helps me by suggesting and guiding during my thesis work.I am very thankful for his cooperation and assistance in reviewing and commenting to complete this work.

Finally,my special thanks is to my Friend Mr.Dawit Gizachew for his time dedicated to support me until completion of my thesis and my mother for praying for me and encouraging me.

Abstract

Several control methods for an overhead crane modeled as a double pendulum with constant cable length have been published in various studies. Most of the proposed control methods were open-loop and linear control methods or nonlinear control methods that fully depended on the system model.However, the dynamic of an overhead crane is a complex nonlinear function of uncertain or unknown parameters, which reduces the performance of such control methods. In this thesis, an adaptive radial basis function neural network-based hierarchical sliding mode controller (ARBFNN-HSMC) is designed to control a 2-dimensional overhead modeled as a double pendulum system with variable cable length using the Lagrange equation of motion. To reduce the chattering effect of the sliding mode controller as well as increase its robustness, ARBFNN is designed to estimate unknown or uncertain nonlinear functions in the system. The overall control law, which contains only some parts of the crane model, is designed, and the adaptation law is derived from the Lyapunov stability condition to update the weight of the network based on observed errors. The proposed control strategy and derived model are verified using MATLAB/Simulink software.For the same controller parameters,500% changes in model parameters are taken, and trolley displacement settling time and rising time for HSMC are 12.3 seconds and 6.95 seconds, respectively. On the other hand, the maximum hook's and payload's swing angles are around 1.34 deg and 1.9 deg for HSMC, and it is around 1.04 deg and 1.64 deg for ARBFNN-HSMC. The residual hook's and payload's swing angles are 0.0137 deg and -0.0319 deg, respectively, in the case of HSMC and -0.0011 deg and -0.0022 deg for ARBFNN-HSMC. This numerical result shows that ARBFNN-HSMC has better performance than HSMC for large parameter variations. In addition, the controller output of ARBFNN-HSMC is smoother than that of HSMC, as evidenced by the result.

Key words: *RBFNN,HSMC, 2D overhead crane,Double pendulum,Adaptive law*

Contents

List of Tables

List of Figures

List of Abbreviations

2D	Two-dimensional
DP	Double pendulum
DPOC	Double pendulum overhead cran
SMC	Sliding mode controller
HSMC	Hierarchical sliding mode controller
ARBFNN	Adaptive Radial Basis Function Neural Network
LMM	Lambed Mass Modeling
DPM	Distributed Parameter Modeling
ZV	Zero Vibration
ZVD	Zero Vibration Derivative
ZVDD	Zero Vibration Derivative-Derivative
PD	Proportional derivative
PID	Proportional integral derivative
MRCS	Model reference command shaping
PSO	Particle swarm optimization
HJB	Hamilton-Jacobi-Bellman

Chapter 1

Introduction

In this chapter, a general overview of the topic, including background information and motivation, the statement of the problem, the objective of the thesis, the methodology, the contribution of this thesis, the scope, and the thesis outlines, is given.

1.1 Background and Motivation

A crane is a modern means of transportation mechanical system ,to load, move, and unload heavy and hazardous materials from one location to the desired location. Based on their dynamic characteristics and the coordinate systems that can describe the location of the rope suspension point most naturally, cranes are classified as gantry cranes and rotary cranes. Gantry cranes can be either overhead or container cranes. A rotary crane can also be a boom or tower crane[1].Each type of crane system is designed for different purposes. For example, an overhead cranes are mostly utilized in industrial settings including shipyards, factories, and warehouses. Container cranes are used at seaports the container ships.Tower and boom cranes are commonly seen at shipyards, harbors, and tall construction projects.The figures 1.1-1.4 depict the primary varieties of cranes.

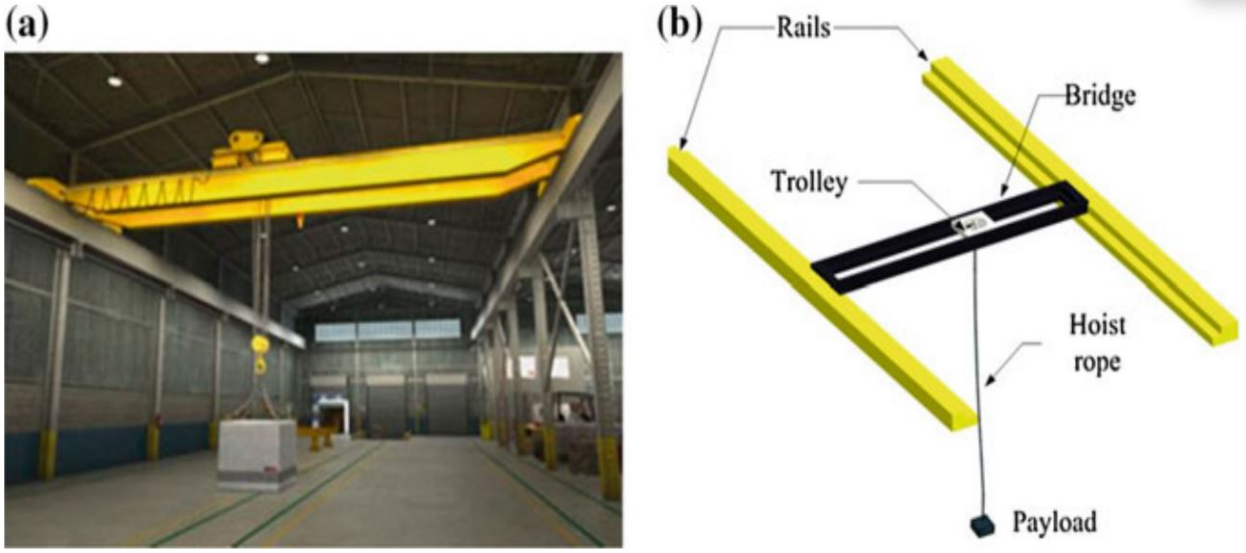


Figure 1.1: Overhead crane

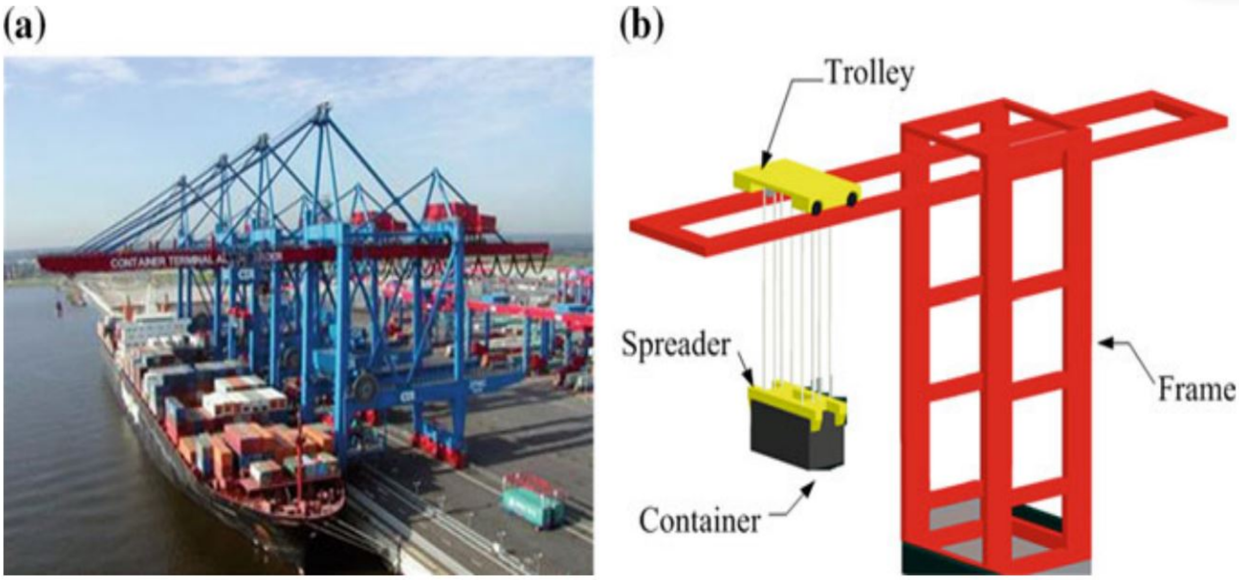


Figure 1.2: Container crane

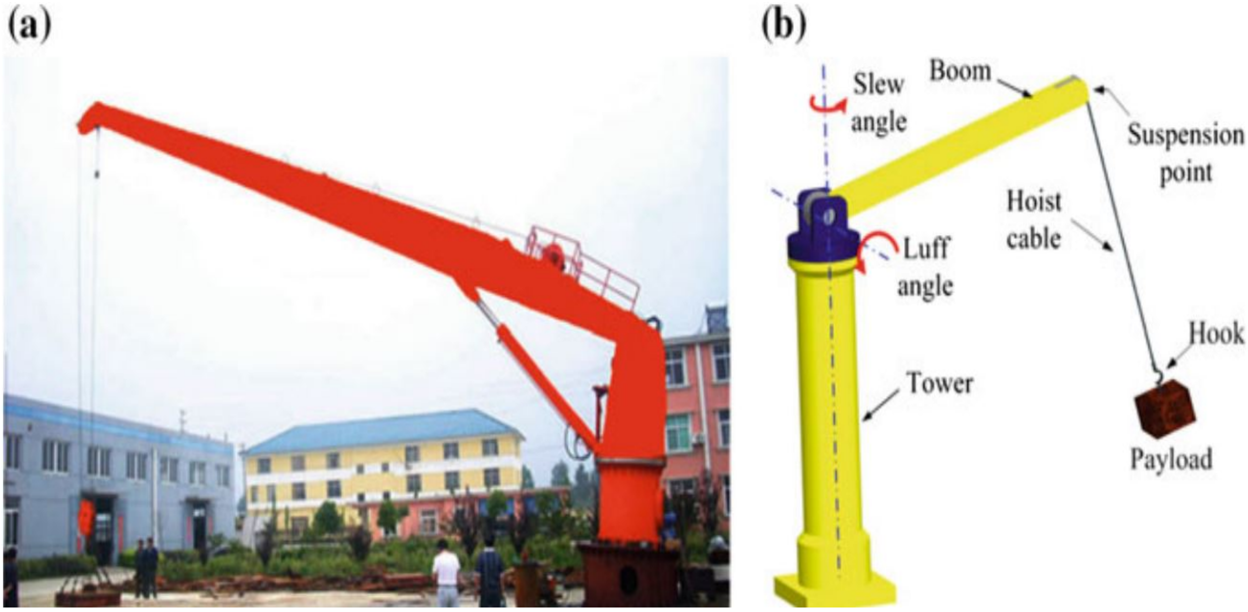


Figure 1.3: Boom crane

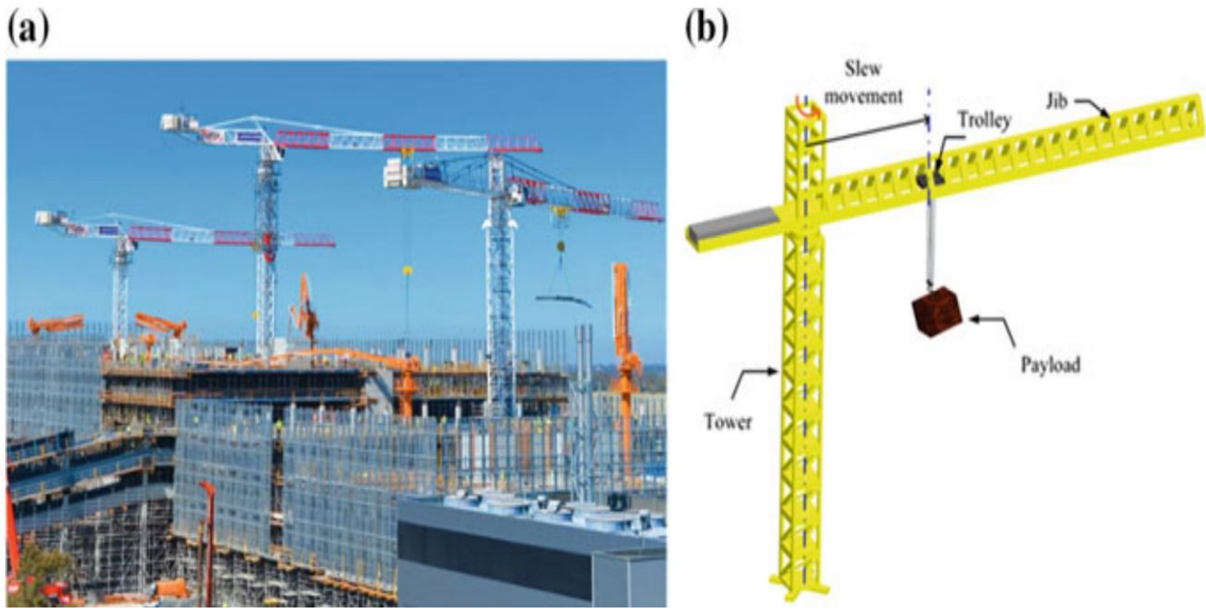


Figure 1.4: Tower crane

Modern industries need all types of cranes to operate at high speeds in order to boost output. Not only that, but the location where the crane works is quite concerned about safety issues. One issue that arises when a crane runs at a fast speed is that the payload experiences big amplitude vibrations. There will be fatalities as well as harm to the cargo and the surroundings from this significant oscillation. For safe and efficient operation of the crane system, it should be controlled appropriately. However, in many industries, most of the crane operations are controlled manually by the crane operator.

Even for experienced operators, operating crane equipment manually becomes exceedingly challenging. After working for a while, operators may make mistakes that reduce productivity and potentially result in safety incidents.citeli2023input.For instance, a steel manufacturing company's accident log from 2014 to 2017 reveals that overhead crane operations account for the majority of incidents, with a percentage of 42% .The most dangerous operations for overhead cranes are those involving human error[3].In order to prevent residual swings, poor precision, and safe crane operation, skilled operators are required. When there is a big hook to suspend the cargo, the manual approach becomes even more challenging because it functions as a double pendulum system. Crane operators with training can prevent payload oscillation if their machines operate like a single pendulum. Nevertheless, some payload configurations and rigging lead to double-pendulum dynamics[4].An automated control crane is required to boost productivity, prevent accidents involving cranes, and lessen the requirement for operators with the necessary training and experience.

The utilization of automatic control in overhead crane systems confers notable benefits over human control in terms of augmenting industrial production and guaranteeing safety measures. The primary function of crane control is to precisely position the cargo and crane at the intended location by suppressing the double-pendulum dynamic through the implementation of feedback or open loop control. Payload swing reduction and crane (trolley) positioning are the primary control mechanisms[5].An appropriate control system should be designed to transport the payload to a desired position as accurately and quickly as possible while suppressing payload oscillations. In order to operate a crane system, this thesis suggests combining an adaptive radial basis function neural network with a hierarchical sliding mode controller. ARBFNN is used to estimate a portion of the overhead crane dynamic model online, which helps to get around the challenge of the complex dynamic model and lessen the sliding mode controller's reliance on it. Furthermore, the neural network should adaptively estimate the plant dynamic online in order to get around uncertainty problems in the crane dynamic equation, such as the payload's mass. The adaptation law is derived from the Lyapunov stability condition.HSMC is selected since it is suitable for systems with fewer actuator numbers than its degree of freedom[6].

In this thesis, a physical model of a 2D overhead crane is obtained from [7].Then, the dynamic model of the system is developed by including both the double pendulum (DP) system (without ignoring hook mass) and hoisting of the payload, which was not considered in most of the work done before. In many published papers, crane modeling is assumed to be a single pendulum with hoisting of the payload or a double pendulum without hoisting of the payload. Moreover, the developed nominal dynamic model of the crane still does not represent its physical model, and to make HSMC more robust and practical, an uncertain dynamic model of the crane is approximated online with a neural network.
RBFNN is initially trained offline, and the best network parameters from offline training are set as initial parameters for the online estimation of the crane dynamic model. This reduces the estimation error between the actual crane model and its estimation. MATLAB/Simulink software is used to verify the model and the suggested controller, and theoretical proof is provided for the stability of the closed-loop system as a whole. The robustness of the controller is tested in a variety of scenarios, and the suggested

approach (ARBFNN-HSMC) is contrasted with completely model-based conventional HSMC. According to simulation data, the suggested approach outperforms HSMC in handling plant uncertainties while still producing results that are comparable to those of nominal model-based HSMC.

1.2 Statement of the Problem

What makes controller design challenging for crane systems is its complex dynamic properties and the error between the actual crane physical model and its mathematical model. In a practical application, overhead cranes are usually subjected to system parameter uncertainties, such as uncertain payload masses, cable lengths, frictions, and external disturbances, such as air resistance[8].These factors reduce the robustness and real application of even nonlinear controllers, such as sliding mode controllers. Moreover, practical crane systems contain huge hooks to suspend large payloads, so they should be modeled as double pendulums with variable cable lengths, which was not considered in many reported papers. Both payload lifting and trolley driving control forces should be designed and applicable to real crane operations. Therefore, this thesis proposes HSMC, which is based on the Adaptive Radial Basis Function Neural Network (ARBFNN). (ARBFNN) estimates the unknown or uncertain nonlinear function of the system so that the robustness of HSMC can be increased.The design of a neural network-based SMC is more suitable than others due to its robustness and stability in terms of external parameter variation. Neural networks have some prior properties, such as online learning, adaptive learning, and the ability to deal with uncertainties. Moreover, the chattering effect in traditional SMC will be eliminated in neural network-based SMC.

1.3 Objective of the study

1.3.1 Main objective

The main objective of this thesis is to design a control law (input forces) applied to the trolley and hoisting cable to lift and transport the payload to the desired position while suppressing both hook and payload oscillations.

1.3.2 Specific objectives

- To design SMC controller based on a nominal crane model.
- To approximate unknown and uncratin nonlinear function of 2D DPOC using adaptive RBFNN with Lyapunove stability condition-based derived adaptation law.
- To design an ARBFFN-based hierarchical switching mode controller.
- To compare HSMC and ARBFNN-HSMC simulation results using Matlab/Simulink.

1.4 Contribution of the thesis

The main contributions of this thesis are summarized as follows:

- The double pendulum system and the payload hoisting and lowering motion are considered in deriving crane dynamic modeling.
- The RBF neural network is designed to model the unknown and uncertain nonlinear functions of DPOC.
- The double pendulum system and the payload hoisting and lowering motion are considered in deriving crane dynamic modeling. The RBF neural network is designed to model the unknown and uncertain nonlinear functions of

DPOC.Without linearization of the system model, HSMC-ARBBFNN is designed so that the accurate model dependency of SMC is reduced.

1.5 Methodology

Figure 1.5 clearly shows the methods followed while doing this research. The overall modeled system analysis is done in MATLAB and Simulink environments, whether the control objectives are achieved or not.

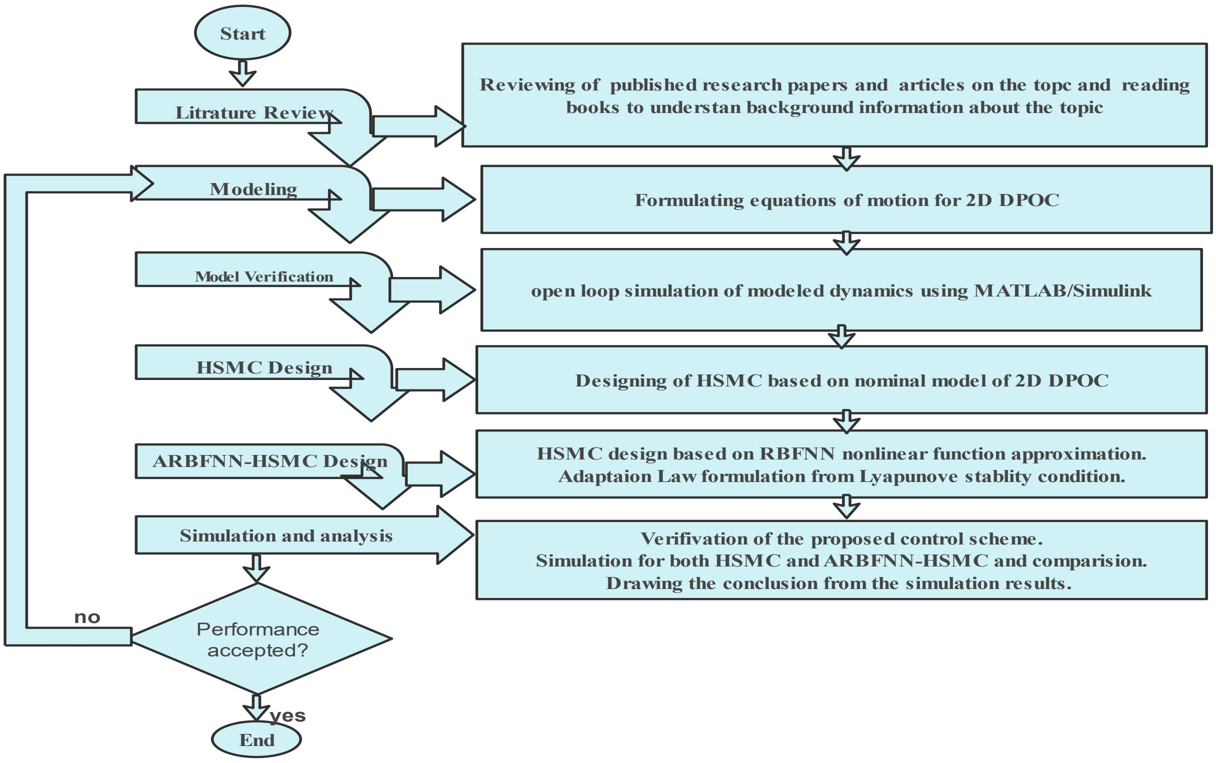


Figure 1.5: Steps during the thesis work

1.6 Scope of the thesis

The formulation of the dynamic model of 2D DPOC using the Lagrange method and its validation are done using MATLAB/Simulation. The proposed control is then applied to hoist the payload at the desired height and drive the crane to the desired position using a simulation model designed in MATLAB/Simulink. The nominal parameters of the crane are taken from the laboratory crane used in[7].For simulation purposes, in this thesis, all mathematical models of the crane are used, but if it is a practical experiment, other parts of the system's dynamics are estimated online using a sensor signal.

1.7 Outline of the thesis

The rest of this thesis is organized as follows:

Chapter 1:Introduction

In this chapter, background information on overhead crane , the objectives of the thesis, and the main contribution of this work are addressed.

Chapter 2:Literature Review

Contains a review of crane control methods, types of crane dynamic modeling, and related work on single-pendulum and double-pendulum 2D overhead cranes.

Chapter 3:System modeling

The dynamics of double-pendulum overhead cranes with payload hoisting and lowering motions are presented.

Chapter 4:Controller design

Gives the detailed steps of controller design and stability analysis.

Chapter 5: Simulation Results and Discussion

Simulation using MATLAB/Simulink is carried out to demonstrate the efficiency of the proposed approach, and a comparison between HSMC and ARBFNN-based HSMC is discussed.

Chapter 6: Conclusion and Future Work

The last chapter presents the conclusions from the studied work and the work to be done in the future.

Chapter 2

Literature Review

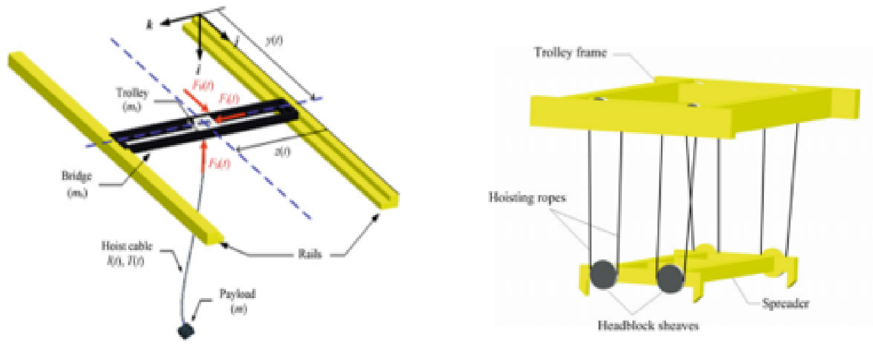
Prominent scholars have published on crane dynamic modeling and various control approaches.A thorough overview of modeling and control techniques for cranes was provided in [9]. This chapter covers approaches to dynamic modeling of crane systems, control techniques, and related works on single- and double-pendulum crane systems.

2.1 Overhead crane modeling

A crucial problem in any control system is figuring out the mathematical equations of the system that has to be controlled. Different control strategies can be developed based on the global dynamic that characterizes the system. There are two general methods for modeling crane systems, which are mentioned in [10].They are given below.

2.1.1 Lambed Mass Modeling (LMM)

The deflection in each of the system's constituent components is not taken into account in LMM. For instance, when hoisting rope is described as a massless cable and is thought to involve rigid body movement, there is supposed to be no deformation.According to [1], there are three payload hoisting mechanisms in LMM.These include double-pendulum systems, multi-rope hoisting mechanisms, and single-rope hoisting mechanisms. A crane dynamic model is developed for a single-rope hoisting mechanism by treating the hook and payload as a single lumped mass.The hook and payload assembly of the multi-rope hoisting mechanism are suspended from the trolley by a number of ropes. Conversely, twin pendulum hoisting systems happen when the payload is the second pendulum mass and the hook's mass is the first, due to its huge mass. It is more precise and useful to model a crane assuming a double-pendulum hoisting mechanism.



a b

Figure 2.1: Overhead crane with single rope(a) and multiple rope hoisting mechanism(b)

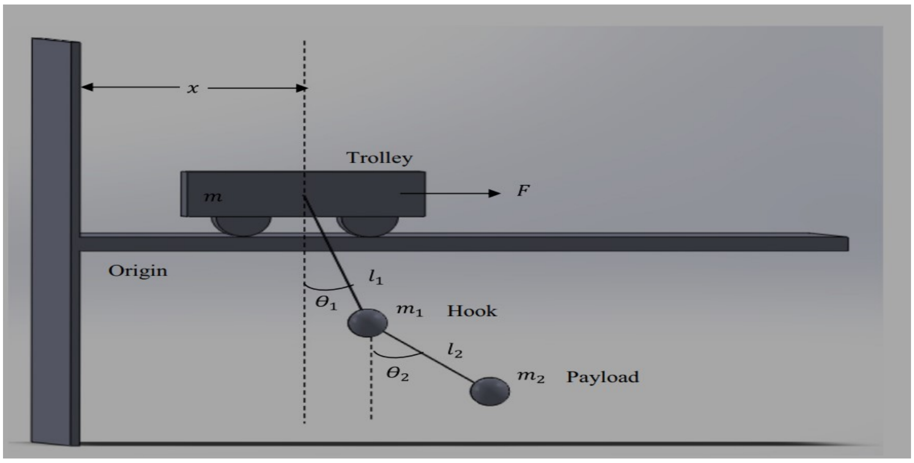


Figure 2.2: Overhead crane with double pendulum hoisting mechanism

2.1.2 Distributed Parameter(mass) Modeling (DPM)

This type of modeling technique does not take into account the different sections of the system having rigid bodily movements.The components' flexibility and deformation are taken into account. For instance, when a crane is operating, the hoisting rope may distort, but the supporting components only vibrate noticeably when a very heavy payload is being moved. The hoisting line, or cable, is represented in this modeling technique as a distributed mass cable and Crane systems' distributed parameter models are significant because they capture the impact of outside pressures and disturbances on the flexible hoisting rope.

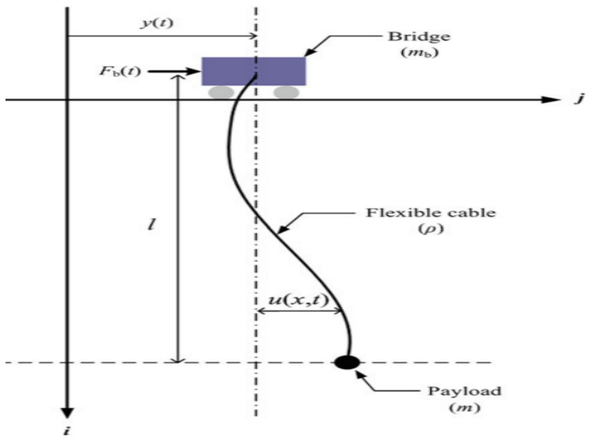


Figure 2.3: 2D overhead crane with flexible hosting rope

For both modeling approaches, various methods were proposed and investigated to derive equations of motion for crane systems. These include the Lagrangian, the Kane, and the discrete-time transfer matrix methods.

2.2 Review on crane control methods

As previously stated, the control problem for any crane type is to find a controlled force that can move the system to the desired reference with minimum payload oscillation during transportation and zero residual vacillation after the transportation of the cargo. Generally, there are two types of crane control methods. These are open-loop and closed-loop controls.

2.2.1 Open loop control methods

Input shaping and trajectory planning are the two commonly known open-loop control methods for a crane[2].

Input shaping:- The foundation of this open-loop crane control technique is the convolation of a human-generated command signal with a series of impulses known as shapers. The crane is driven by the filtered-generated new signal, which also suppresses the payload oscillation shape.

Trajectory-planning:- By determining predetermined trajectories to move the crane through the workspace, this method (optimal trajectory planning) aims to reduce payload oscillation.

2.2.2 Closed loop control methods

Closed-loop crane control generates commands that move the system toward the desired state based on information about the current states of the crane, such as trolley and hoisting cable positions, payload, and hook swing angles. It is based on identifying a feedback control law that stabilizes the system in the face of various uncertainties and external disturbances. Regarding crane control, there are numerous kinds of closed-loop control techniques. A few of them include adaptive control, mixed open loop, fuzzy logic control, neural network control, sliding mode control, energy-based techniques, feedback linearization, and neural networks.

2.3 Related works on 2D overhead crane

In this section, some published papers, reports, and articles are reviewed for both 2D overhead cranes with a single pendulum (2D SPOC) and with a double pendulum (2D DPOC).

2.3.1 Related works on 2D overhead crane with single pendulum

For a 2D overhead crane, numerous academics have created various control schemes and models based on a single pendulum approach.These include the overhead crane system's dynamics analysis and fuzzy anti-swing control design, which are based on the modeling technique of the Riccati discrete-time transfer matrix approach in [11].Additional control techniques include neural network control, energy-based control, sliding mode control, adaptive control, path planning, and input shaping in[12],[13, 14],[15, 16],[6, 17, 18, 19],[20],[21]respectively used the Lagrange method to drive the equation of the system.

2.3.2 Related works on 2D overhead crane with double pendulum

Some works were also given to model and control a 2D overhead crane with a double pendulum, similar to the single pendulum crane system.Based on Kane's modeling technique, a double-pendulum bridge crane with distributed-mass beam dynamics and swing control was given in [22].Motion-induced vibrations were minimized by utilizing a command smoother signal to smooth the initial command signal. Kane's modeling method was applied in [23] to address control of distributed-mass payload bridge cranes in windy conditions conditions.Other papers designed their suggested schemes using the Lagrange approach. For example, input shaping-based control of 2D DPOC was proposed by William Singhose in[25] andTan Ying Jian and Z. Mohamed in[24]. In the first study, payload oscillation was suppressed by designing zero vibration (ZV), zero vibration derivative (ZVD), and zero vibration derivative-derived (ZVDD) input shapers based on the linearized model of 2D DPOC's natural frequency and damping ratio. Both in the simulation and experiment, the addition of the input shapers greatly decreased the system's maximum hook sway. To reduce the double-pendulum effects, an input shaping method based on the linear, time-invariant dynamics of the crane was developed in the second study. Due to its sensitivity to outside disturbances and incapacity to manage unexpected vibration from significant nonlinearity and parameter variance, input shaping is weak[7, 26].

In [27], time-optimal trajectory planning was discussed.The control target, actuator saturation, and the system's maximum swing amplitude were taken into consideration when formulating the optimization problem. The suggested optimization issue was converted into a discrete form using a linearized and discretized model of a double pendulum crane. It was then shown to be a quasi-convexity optimization problem, which was resolved by a bisection approach. After determining the time-optimal trolley trajectory, a tracking controller was created to direct the trolley along the intended path. Through modeling and experimentation, the suggested approach was evaluated, and satisfactory results were produced.

Several articles have proposed combinations of input shaping and feedback control algorithms to guarantee precise crane positioning and minimize payload swing. For example, a dual feedback controller with proportional feedback for the hook deflection and PD feedback for the trolley position was presented in [28]. The proportional feedback loop was added to ensure a null residual vibration even during a transient vibration and a transient vibration response that is an adjustable or continuous disturbance. The PD feedback loop was paired with the shaper to drive the trolley to the target with little induced vibration. Command shaping method based on reference model and feedback controller was done in ref.[29] to control an overhead crane with double pendulum dynamics. PSO (particle swarm optimization) and PID were used in the work to simultaneously modify the MRCS-PID controller parameters. It's not required. to measure or estimate many modes of frequency and damping ratio because the proposed MRCS is constructed using a reference model, in contrast, to earlier input shaping and command shaping approaches[30].

An efficient proportional-integral-derivative (PID) control of a double-pendulum overhead crane with extreme nonlinearity was published in a study on ref.no[31]. The PSO method adjusted the PID controller's ideal parameters. Two PID control strategies were developed using a linearized model of DPOC as the foundation. In the first example, the payload oscillation, the hook, and the trolley position were used as the feedback signals, and three different PID controllers were used for each control variable. In the second example, two PID controllers were employed, and the payload oscillation was measured without the use of a sensor by relying solely on the trolley position and hook oscillation as feedback signals. The PID tuning approach was the paper's primary contribution. To clarify, an improved PSO algorithm was proposed, which took into account the oscillations of the vertical distance and the potential energy of the crane.The controller's effectiveness was assessed by simulations conducted under different double-pendulum crane operation conditions.It was also compared how well the suggested control law worked with the commonly used PSO-tuned PID controllers, and between two and three PID controls. External disruptions and the motion of the cargo being raised or lowered were not considered in the essay either.

Additionally, a number of feedback control techniques were looked into for precise trolley positioning and double pendulum crane anti-swing control. For instance, energy-based control techniques, sometimes referred to as passivity-based and backstepping based control, were introduced in [32] and [33], respectively.The adaptive tracking control of double overhead cranes confined to the tracking error limit, parameter uncertainty, and external disturbance was discussed by the authors of [8]. They developed a tracking controller that uses the S-shaped smooth function as the best trajectory for the trolley to determine its position, acceleration, and velocity to the desired position. The tracking control approach derives from acceleration and velocity. First, the energy-like non-negative function is derived using the total energy of the double pendulum crane. From the derivative total non-negative energy function, a control strategy that guarantees the trolley error always remains within predefined bounds and rapidly approaches zero is then built.Additionally, the paper's update law is applied to estimate online unknown parameters.

In [34], the authors presented a distributed mass modeling approach with time-varying sliding mode control for double pendulum overhead cranes. To lessen vibration and boost the overhead crane's robustness, they created a time-varying sliding mode control technique. A time-varying sliding surface is designed in the article to improve the sliding mode controller's adjusting capability. Good results were obtained from both simulation and experiments. By Biao Lu, Yongchun Fang, and Ning Sun[7], an enhanced-coupling adaptive controller was suggested for DPOC. Within the research, the motion of raising and lowering the payload and a new adaptation law based on estimates of uncertain payload mass were examined. This control problem, which was actually highly hard and challenging with sophisticated design and analysis, takes into account the payload hoisting/lowering motion, payload mass uncertainty, and increased swing feedback. To confirm the suggested method's effectiveness, hardware experiments were conducted.

In[35],novel adaptive hierarchical sliding mode controller was presented for double pendulum overhead crane.The proposed methods use active sliding surface to search for the system states.Time varying parameters in second level sliding surface is estimated by using some designed updating law and used in the control law. Experimental result is compared with other controls method such as conventional sliding mode control and linear state feedback control.

In [36], a non-linear state feedback controller with RBF is suggested.The algorithm relies on two control strategies: the corrective control term, which makes use of RBFNN to account for external disturbances and system uncertainties, and the feedback control term, which is based on the double pendulum nominal model. Without taking into account external disturbances or system uncertainties, the feedback controller is derived from a chosen Lyapunov function. Next, system uncertainties and external disturbances are estimated using adaptive RRFNN to develop a corrective control term. Lyapunov stability analysis is used to establish the update law, and MATLAB simulation is used to evaluate the suggested scheme's performance with alternative control strategies.

Mengyuan Li, He Chen, and Zhaoqi Li[2] provided input-limited optimal control for a double pendulum overhead crane with cargo hoisting and lowering. This investigation has observed that actuators in a real-world crane system have saturated output values.When the intended controller computed an unnecessarily large intended output, control performance would inevitably fall.To accomplish exact trolley position, rope length regulation, and swing suppression for both the hook and the payload, the authors designed an optimal controller that took control input limits into account. With the aim of using less energy to control a double pendulum over a head crane, this controller was created with consideration for the actuator's saturation problem. A sliding surface and a specified performance index function are used to calculate the input-limited optimal controller. It is difficult to solve HJB analytically due of its nonlinearity and complexity. A neural network is used to identify an input-limited optimal controller and predict the best performance index.

Chapter 3

Modeling and Validation of 2D DPOC

In this chapter, dynamic equations representing a double-pendulum overhead crane with variable and hoisting cable are derived based on the Lagrange equation of motion.

3.1 Physical Model and components of overhead crane

Generally, a practical overhead crane consists of a bridge, trolleys, and traveling mechanisms, as shown in figure3.1 as given in[37].

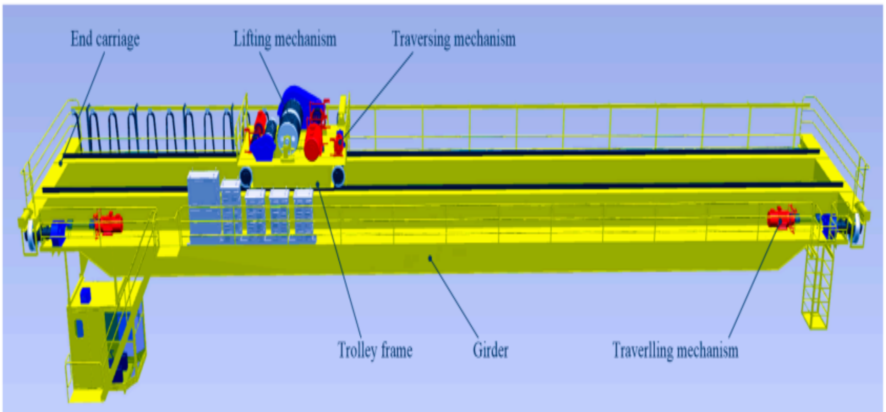


Figure 3.1: Physical model of overhead crane

The bridge of an overhead crane is a supporting element consisting of two identical girders and two identical end carriages that are fused together to form a steel structure with thin walls. Two crane rails support the bridge, and crane wheels with a cylindrical tread support it. A crane trolley, sometimes referred to as a car, has lifting mechanisms, a traversing mechanism, and a trolley frame. It is used to move a bridge. The three primary components of a lifting mechanism are a wire rope, a lifting motor, and a load-handling tool (hook).

3.2 General Modeling assumptions

Given the above features of overhead cranes, some reasonable assumptions are made to simplify the dynamic modeling of overhead cranes, as follows:

1. LMM (Lamped Mass Modeling) assumptions.

- Double pendulum system (the mass of the hook is considered)
- No deflection of hoisting rope (the cable is considered a rigid body).

2.2-Dimensional motion.

- Bridge movement is not considered.
- Only trolley movement in the horizontal x-axis direction and oscillation of both the payload and hook in the XY plane are considered.

3. The mass of the hook and payload is considered the point mass.
4. Varying cable length between trolley and hook and constant cable length between hook and payload with negligible mass.
5. Actuator dynamics are not considered in the dynamic motion of the crane.

The system depicted in the following Figure can be used to abstract the physical model of a standard overhead crane, assuming the aforementioned premises.

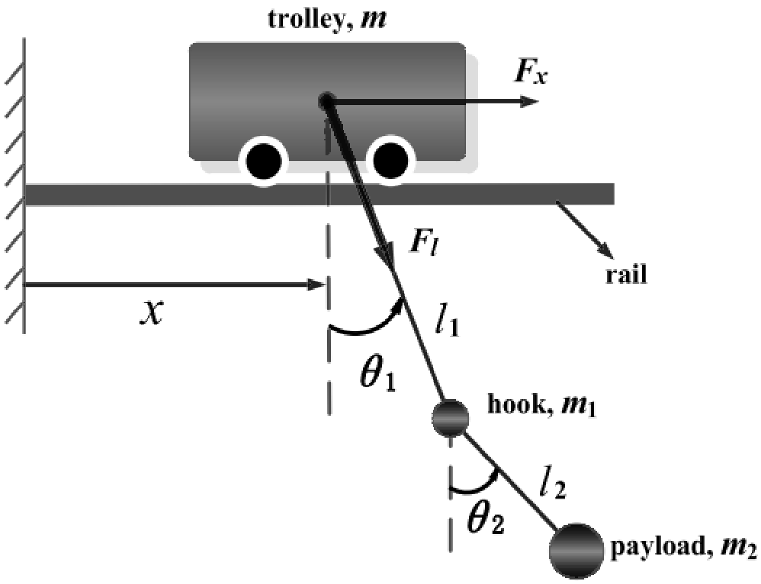


Figure 3.2: Simplified physical model of 2D DPOC

Table 3.1: Parameters and state variables

Parameters	Description
$\boldsymbol{m}$	Total mass of trolley
$\boldsymbol{m_1}$	Mass of the hook
$\boldsymbol{m_2}$	Mass of the payload
$\boldsymbol{x}$	Position of trolley
$\boldsymbol{\theta_1}$	Angle between vertical axis and first pendulum (l_1+hook)
$\boldsymbol{\theta_2}$	Angle between first pendulum and second pendulum (l_2+payload)
$\boldsymbol{l_1}$	Variable Cable length between trolley and hook
$\boldsymbol{l_2}$	Constant cable length between hook and payload
$\boldsymbol{F_x}$	Input force of trolley
$\boldsymbol{F_l}$	Hoisting force

3.3 Equation of motion using Euler-Lagrange method

Modeling a double-pendulum crane system using the Lagrangian method was the most widely used technique. The equations of motion for complex systems of higher order are obtained using Hamilton's principle, which is also referred to as Lagrange's equations. Utilizing the idea of energy conservation, it is founded on the calculus of variations. The difference between the system's potential and kinetic energies is what defines the Lagrange function. Subsequently, the Lagrange function is solved to yield the mathematical equations representing the system. The Lagrange equation with respect to the coordinate q_i is given by[34]:

$$\frac{d}{dt}\left(\frac{L}{\dot{q}_i}\right) - \frac{\partial L}{q_i} = F_i \tag{3.1}$$

Where L represents the Lagrange function, which is defined as follows.

L = KE – PE KE, KP, and F_i represent kinetic energy, potential energy, and the external force of the system, respectively.

Nonlinear dynamic modeling of 2D DPOC

From figure 3.2, the positions of the trolley(x_1, y_1), the hook (x_2, y_2) and the payload (x_3, y_3) in the XY-plane are given by:

$$\begin{cases} \mathrm{x}_1 = x \\ \mathrm{y}_1 = 0 \\ \mathrm{x}_2 = x + \mathrm{l}_1\mathrm{sin}\theta_1 \\ \mathrm{y}_2 = -\mathrm{l}_1\mathrm{cos}\theta_1 \\ \mathrm{x}_3 = x + \mathrm{l}_1\mathrm{sin}\theta_1 + \mathrm{l}_2\mathrm{sin}\theta_2 \\ \mathrm{y}_3 = -\mathrm{l}_1\mathrm{cos}\theta_1 - \mathrm{l}_2\mathrm{cos}\theta_2 \end{cases} \tag{3.2}$$

Let $\boldsymbol{V_T}, \boldsymbol{V_H}$ *and* $\boldsymbol{V_T}$ be velocity vectors of the trolley, hook, hook, and payload, respectively. They are obtained by taking the time derivatives of equation 3.2 as follows:

$$\boldsymbol{V_T} = \dot{\mathrm{x}}_1 i + \dot{\mathrm{y}}_1 j$$

$$\boldsymbol{V_H} = \dot{\mathrm{x}}_2 i + \dot{\mathrm{y}}_2 j$$

$$\boldsymbol{V_P} = \dot{\mathrm{x}}_3 i + \dot{\mathrm{y}}_3 j$$

The total kinetic energy of the system (KE) is the sum of the kinetic energy of the trolley, hook, and payload (KE_T, KE_H *and* KE_P).

$$KE = KE_T + KE_H + KE_P = \frac{1}{2} m V_T^2 + \frac{1}{2} m_1 V_H^2 + \frac{1}{2} m_2 V_P^2$$

where V_T^2, V_H^2 V_H^2, and V_P^2 are the velocities of the trolley,the hook, and the payload, respectively, and calculated as:

$$V_T^2 = |\boldsymbol{V_T}|^2 = \dot{x}_1^2 + \dot{y}_1^2 = \dot{x}^2$$

$$V_H^2 = |\boldsymbol{V_H}|^2 = \dot{x}_2^2 + \dot{y}_2^2 = \left(\dot{x} + \dot{l}_1 sin\theta_1 + l_1\dot{\theta}_1 cos\theta_1\right)^2 + \left(\dot{l}_1 cos\theta_1 - l_1\dot{\theta}_1 sin\theta_1\right)^2$$

$$V_P^2 = |\boldsymbol{V_P}|^2 = \dot{x}_3^2 + \dot{y}_3^2 = \left(\dot{x} + \dot{l}_1 sin\theta_1 + l_1\dot{\theta}_1 cos\theta_1 + l_2\dot{\theta}_2 cos\theta_2\right)^2 + \left(\dot{l}_1 cos\theta_1 - l_1\dot{\theta}_1 sin\theta_1 - l_2\dot{\theta}_2 sin\theta_2\right)^2$$

Then, kinetic energy becomes:

$$KE = \tfrac{1}{2} m\dot{x}^2 + \tfrac{1}{2} m_1 \left(\left(\dot{x} + \dot{l}_1 sin\theta_1 + l_1\dot{\theta}_1 cos\theta_1\right)^2 + \left(\dot{l}_1 cos\theta_1 - l_1\dot{\theta}_1 sin\theta_1\right)^2\right) + \tfrac{1}{2} m_2 \left(\left(\dot{x} + \dot{l}_1 sin\theta_1 + l_1\dot{\theta}_1 cos\theta_1 + l_2\dot{\theta}_2 cos\theta_2\right)^2 + \left(\dot{l}_1 cos\theta_1 - l_1\dot{\theta}_1 sin\theta_1 - l_2\dot{\theta}_2 sin\theta_2\right)^2\right) \tag{3.3}$$

From Figure3.2, it is assumed that the cart or trolley moves only in a horizontal direction, which means its potential energy is unchanged. However, both hook and payload have initial potential energy as they are suspended from the trolley, and they also have final potential energy as they move in the XY plane. So, the total potential energy of the system is the sum of the potential energy of the hook and the payload, and it is derived and given below.

$$PE = PE_h + PE_p \tag{3.4}$$

Where PE_h and PE_p are changes in the potential energy of the hook and payload, respectively.

$$PE_h = PE_{hf} - PE_{hi} = \mathrm{m}_1 g \mathrm{y}_2 - (-\mathrm{m}_1 g \mathrm{l}_1) = -\mathrm{m}_1 g \mathrm{l}_1 \cos\theta_1 - (-\mathrm{m}_1 g \mathrm{l}_1) \tag{3.5}$$

$$PE_p = PE_{pf} - PE_{pi} = \mathrm{m}_2 g \mathrm{y}_2 - (-\mathrm{m}_2 g(\mathrm{l}_1 + \mathrm{l}_2)) = \mathrm{m}_2 g(-\mathrm{l}_1 \cos\theta_1 - \mathrm{l}_2 \cos\theta_2) - (-\mathrm{m}_2 g(\mathrm{l}_1 + \mathrm{l}_2)) \tag{3.6}$$

Now the total potential energy of the system is:

$$PE = PE_h + PE_p = m_1 g l_1 (1 - cos\theta_1) + m_2 g \left[l_1 (1 - cos\theta_1) + l_2 \ (1 - cos\theta_2) \right] \tag{3.7}$$

Where g denotes the gravitational acceleration.

From Figure3.2, there are four independent generalized coordinates of the system: (x(t), $l_1(t)$, $\theta_1(t)$, $\theta_2(t)$) and two generalized forces ($F_x(t)$, $F_l(t)$). Therefore, the Lagrange function L is given by

$L = KE - PE$

$$\begin{aligned} L = &\tfrac{1}{2} m\dot{x}^2 + \tfrac{1}{2} m_1 \left(\left(\dot{x} + \dot{l}_1 sin\theta_1 + l_1 \dot{\theta}_1 cos\theta_1 \right)^2 + \left(\dot{l}_1 cos\theta_1 - l_1 \dot{\theta}_1 sin\theta_1 \right)^2 \right) + \\ &\tfrac{1}{2} m_2 \left(\left(\dot{x} + \dot{l}_1 sin\theta_1 + l_1 \dot{\theta}_1 cos\theta_1 + l_2 \dot{\theta}_2 cos\theta_2 \right)^2 + \left(\dot{l}_1 cos\theta_1 - l_1 \dot{\theta}_1 sin\theta_1 - l_2 \dot{\theta}_2 sin\theta_2 \right)^2 \right) \\ &- \left(m_1 g l_1 (1 - cos\theta_1) + m_2 g \left[l_1 (1 - cos\theta_1) + l_2 \ (1 - cos\theta_2) \right] \right) \end{aligned} \tag{3.8}$$

According to equation3.1, the dynamic equations of 2D DPOC are obtained as follows:

$$\begin{aligned}
&\frac{d}{dt}\left(\frac{L}{\dot{x}}\right)-\frac{\partial L}{x}\ =\left(m+m_1+m_2\right)\ddot{x}+\\
&\left(m_1+m_2\right)sin\theta_1\ddot{l}_1+\left(m_1+m_2\right)l_1\ddot{\theta}_1cos\theta_1+m_2l_2\ddot{\theta}_2cos\theta_2+\\
&2\left(m_1+m_2\right)\dot{l}_1\dot{\theta}_1\ cos\theta_1-\left(m_1+m_2\right)l_1\dot{\theta}_1^2sin\theta_1\ -\ m_2l_2\dot{\theta}_2^2sin\theta_2=F_x
\end{aligned} \tag{3.9}$$

$$\begin{aligned}
&\frac{d}{dt}\left(\frac{L}{\dot{l}_1}\right)-\frac{\partial L}{l_1}\ =(m_1+m_2sin\theta_1\ddot{x}+\left(m_1+m_2\right)\ddot{l}_1+m_2l_2sin\left(\theta_1\ -\theta_2\right)\ddot{\theta}_2-\\
&\left(m_1+m_2\right)l_1\dot{\theta}_1^2-m_2l_2cos\left(\theta_1\ -\theta_2\right)\dot{\theta}_2^2+\left(m_1+m_2\right)g(1-cos\theta_1)=F_l
\end{aligned} \tag{3.10}$$

$$\begin{aligned}
&\frac{d}{dt}\left(\frac{\partial \mathrm{L}}{\partial\dot{\theta}_1}\right)-\frac{\partial \mathrm{L}}{\partial\theta_1}=\left(m_1+m_2\right)l_1cos\theta_1\ddot{x}+\left(m_1+m_2\right)l_1^2\ddot{\theta}_1+\\
&m_2l_1l_2cos\left(\theta_1\ -\theta_2\right)\ddot{\theta}_2+2\left(m_1+m_2\right)l_1\dot{l}_1\dot{\theta}_1\ +\\
&m_2l_1l_2sin\left(\theta_1\ -\theta_2\right)\dot{\theta}_2^2+\left(m_1+m_2\right)gl_1sin\theta_1\ =0
\end{aligned} \tag{3.11}$$

$$\begin{aligned}
&\frac{d}{dt}\left(\frac{\partial \mathrm{L}}{\partial\dot{\theta}_2}\right)-\frac{\partial \mathrm{L}}{\partial\theta_2}=m_2l_2cos\theta_2\ddot{x}+m_2l_2sin\left(\theta_1\ -\theta_2\right)\ddot{l}_1+m_2l_1l_2cos\left(\theta_1\ -\theta_2\right)\ddot{\theta}_1+\\
&m_2l_2^2\ddot{\theta}_2+2m_2l_2cos\left(\theta_1\ -\theta_2\right)\dot{l}_1\dot{\theta}_1-m_2l_1l_2sin\left(\theta_1\ -\theta_2\right)\dot{\theta}_1^2+m_2gl_2sin\theta_2\ =0
\end{aligned} \tag{3.12}$$

The following is a matrix representation of the aforementioned dynamic equations:

$$\mathbf{M}\left(\mathbf{q}\right)\ddot{\mathbf{q}}+\mathbf{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)\dot{\mathbf{q}}+\mathbf{G}\left(\mathbf{q}\right)=\mathbf{u} \tag{3.13}$$

Where $\mathbf{q}=[\mathbf{x},l_1,\theta_1\ ,\theta_2]^{\boldsymbol{T}}$ is a vector of the generalized coordinates of the crane, $\mathbf{u}=[F_x,F_l\ ,0,0]^{\boldsymbol{T}}$ is a vector of the generalized input force, g is the gravitational acceleration, $\mathbf{M}\left(\mathbf{q}\right)$ is a 4×4 inertia matrix, $\mathbf{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)\dot{\mathbf{q}}$ is a vector of Coriolis and centripetal torques, and $\mathbf{G}(\mathbf{q})$ is a vector of the gravitational term.

$$\mathbf{M}=\begin{bmatrix} \boldsymbol{m_{11}} & \boldsymbol{m_{12}} & \boldsymbol{m_{13}} & \boldsymbol{m_{14}} \\ \boldsymbol{m_{12}} & \boldsymbol{m_{22}} & 0 & \boldsymbol{m_{24}} \\ \boldsymbol{m_{13}} & 0 & \boldsymbol{m_{33}} & \boldsymbol{m_{34}} \\ \boldsymbol{m_{14}} & \boldsymbol{m_{24}} & \boldsymbol{m_{34}} & \boldsymbol{m_{44}} \end{bmatrix}, \begin{cases} m_{11} = m + m_1 + m_2 \\ m_{12} = (m_1 + m_2)\, sin\theta_1 \\ m_{13} = (m_1 + m_2)\, l_1 cos\theta_1 \\ m_{14} = m_2 l_2 cos\theta_2 \\ m_{22} = m_1 + m_2 \\ m_{24} = m_2 l_2 sin\,(\theta_1 - \theta_2) \\ m_{33} = (m_1 + m_2)\, l_1^2 \\ m_{34} = m_2 l_1 l_2 cos\,(\theta_1 - \theta_2) \\ m_{44} = m_2 l_2^2 \end{cases}$$

$$L=\begin{bmatrix} \mathbf{0} & \boldsymbol{l_{12}} & \boldsymbol{l_{13}} & \boldsymbol{l_{14}} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{l_{23}} & \boldsymbol{l_{24}} \\ \mathbf{0} & \boldsymbol{l_{32}} & \boldsymbol{l_{33}} & \boldsymbol{l_{34}} \\ \mathbf{0} & \boldsymbol{l_{42}} & \boldsymbol{l_{43}} & \mathbf{0} \end{bmatrix}, \begin{cases} l_{12} = 2\boldsymbol{m_{22}} cos\theta_1 \dot{\theta}_1 \\ l_{13} = \mathbf{2}\boldsymbol{m_{22}} \left(cos\theta_1 \dot{l}_1 - l_1 sin\theta_1 \dot{\theta}_1 \right) \\ l_{14} = -m_2 l_2 sin\theta_2 \dot{\theta}_2 \\ l_{23} = \boldsymbol{-m_{22}} l_1 \dot{\theta}_1 \\ l_{24} = -m_2 l_2 cos\,(\theta_1 - \theta_2)\, \dot{\theta}_2 \\ l_{32} = \mathbf{2}\boldsymbol{m_{22}} l_1 \dot{\theta}_1 \\ l_{33} = \mathbf{2}\boldsymbol{m_{22}} l_1 \dot{l}_1 \\ l_{34} = m_2 l_1 l_2 sin\,(\theta_1 - \theta_2)\, \dot{\theta}_2 \\ l_{42} = 2 m_2 l_2 cos\,(\theta_1 - \theta_2)\, \dot{\theta}_1 \\ l_{43} = 2 m_2 l_2 cos\,(\theta_1 - \theta_2)\, \dot{l}_1 - m_2 l_1 l_2 sin\,(\theta_1 - \theta_2)\, \dot{\theta}_1 \end{cases}$$

$$\mathbf{G}=\begin{bmatrix} 0 \\ g_1 \\ g_2 \\ g_3 \end{bmatrix} = \begin{bmatrix} 0 \\ \boldsymbol{m_{22}} g(1 - cos\theta_1) \\ \boldsymbol{m_{22}} g l_1 sin\theta_1 \\ m_2 g l_2 sin\theta_2 \end{bmatrix} \; and \; \boldsymbol{u} = \begin{bmatrix} \boldsymbol{F_x} \\ \boldsymbol{F_l} \\ 0 \\ 0 \end{bmatrix}$$

3.4 Dynamic separation and state space representation

In figure 3.2, a 2D DPOC is an underactuated system with four outputs and only two forces. The only actuated outputs of the system that are directly controlled by forces are the trolley position and the length of the hoisting cable; hook and payload swing angles are controlled indirectly by varying these actuated outputs. The dynamic equation of the 2D DPOC in3.13 should be divided into actuated and un-actuated equations to aid with HSMC design. Let's rewrite equation 3.10 as

$$\mathbf{M}\left(\mathbf{q}\right)\ddot{\mathbf{q}}+\mathbf{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)\dot{\mathbf{q}}+\mathbf{G}\left(\mathbf{q}\right)=\mathbf{u}$$

Let $\mathbf{q}=[\ \boldsymbol{q_a}\ \ \boldsymbol{q_u}]^{\boldsymbol{T}}$ where $\boldsymbol{q_a}=[\boldsymbol{x}\ \ l_1\ ]^{\boldsymbol{T}}$ is vector of the actuated states and $\boldsymbol{q_u}=[\theta_1\ \ \theta_2\ ]^{\boldsymbol{T}}$ is vector of the un-actuated states.
Let

$$\mathbf{M}\left(\mathbf{q}\right)=\begin{bmatrix} M_1(\mathbf{q}) & M_2(\mathbf{q}) \\ M_3(\mathbf{q}) & M_4(\mathbf{q}) \end{bmatrix}\ ,\mathbf{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)=\begin{bmatrix} L_1(\mathbf{q},\dot{\mathbf{q}}) & L_2(\mathbf{q},\dot{\mathbf{q}}) \\ L_3(\mathbf{q},\dot{\mathbf{q}}) & L_4(\mathbf{q},\dot{\mathbf{q}}) \end{bmatrix},$$

$\mathbf{G}\left(\mathbf{q}\right)=[\ G_1(\mathbf{q})\ \ G_2(\mathbf{q})]^{\boldsymbol{T}}$ and $\mathbf{u}=[\ \boldsymbol{u}\ \ \mathbf{0}]^{\boldsymbol{T}}, \boldsymbol{u}=[\ F_x\ \ F_l\ ]^{\boldsymbol{T}}$ Where each element of $\mathbf{M}\left(\mathbf{q}\right)$ and $\mathbf{L}\left(\mathbf{q},\dot{\mathbf{q}}\right)$ are 2 by 2 matrices and given as:

$$M_1\left(\mathbf{q}\right)=\begin{bmatrix} m+m_1+m_2 & \left(m_1+m_2\right)sin\theta_1 \\ \left(m_1+m_2\right)sin\theta_1 & m_1+m_2 \end{bmatrix}$$

$$M_2\left(\mathbf{q}\right)=\begin{bmatrix} \left(m_1+m_2\right)l_1cos\theta_1 & m_2l_2cos\theta_2 \\ 0 & m_2l_2sin\left(\theta_1\ \ -\theta_2\right) \end{bmatrix}$$

$$M_3\left(\mathbf{q}\right)=\begin{bmatrix} \left(m_1+m_2\right)l_1cos\theta_1 & 0 \\ m_2l_2cos\theta_2 0 & m_2l_2sin\left(\theta_1\ \ -\theta_2\right) \end{bmatrix}$$

$$M_4(\mathbf{q}) = \begin{bmatrix} (m_1+m_2)\,l_1^2 & m_2l_1l_2cos\,(\theta_1\ -\theta_2) \\ m_2l_1l_2cos\,(\theta_1\ -\theta_2) & m_2l_2^2 \end{bmatrix}$$

$$L_1(\mathbf{q}) = \begin{bmatrix} 0 & \boldsymbol{2m_{22}}cos\theta_1\dot{\theta}_1 \\ 0 & 0 \end{bmatrix}$$

$$L_2(\mathbf{q}) = \begin{bmatrix} \boldsymbol{2m_{22}}\left(cos\theta_1\dot{l}_1 - l_1sin\theta_1\dot{\theta}_1\right) & -m_2l_2sin\theta_2\dot{\theta}_2 \\ \boldsymbol{-m_{22}}l_1\dot{\theta}_1 & -m_2l_2cos\,(\theta_1\ -\theta_2)\,\dot{\theta}_2 \end{bmatrix}$$

$$L_3(\mathbf{q}) = \begin{bmatrix} 0 & \boldsymbol{2m_{22}}l_1\dot{\theta}_1 \\ 0 & 2m_2l_2cos\,(\theta_1\ -\theta_2)\,\dot{\theta}_1 \end{bmatrix}$$

$$L_4(\mathbf{q}) = \begin{bmatrix} \boldsymbol{2m_{22}}l_1\dot{l}_1 & m_2l_1l_2sin\,(\theta_1\ -\theta_2)\,\dot{\theta}_2 \\ 2m_2l_2cos\,(\theta_1\ -\theta_2)\,\dot{l}_1 - m_2l_1l_2sin\,(\theta_1\ -\theta_2)\,\dot{\theta}_1 & 0 \end{bmatrix}$$

$$G_1 = \begin{bmatrix} 0 \\ \boldsymbol{m_{22}}g(1-cos\theta_1 \end{bmatrix};G_2 = \begin{bmatrix} \boldsymbol{m_{22}}gl_1sin\theta_1 \\ m_2gl_2sin\theta_2 \end{bmatrix}$$

Then, the actuated and un-actuated models are defined as follows:

$$\begin{bmatrix} M_1(\mathbf{q}) & M_2(\mathbf{q}) \\ M_3(\mathbf{q}) & M_4(\mathbf{q}) \end{bmatrix}\begin{bmatrix} \ddot{\mathbf{q}}_{\boldsymbol{a}} \\ \ddot{\mathbf{q}}_{\boldsymbol{u}} \end{bmatrix} + \begin{bmatrix} L_1(\mathbf{q},\dot{\mathbf{q}}) & L_2(\mathbf{q},\dot{\mathbf{q}}) \\ L_3(\mathbf{q},\dot{\mathbf{q}}) & L_4(\mathbf{q},\dot{\mathbf{q}}) \end{bmatrix}\begin{bmatrix} \dot{\mathbf{q}}_{\boldsymbol{a}} \\ \dot{\mathbf{q}}_{\boldsymbol{u}} \end{bmatrix} + \begin{bmatrix} G_1(\mathbf{q}) \\ G_2(\mathbf{q}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{u} \\ 0 \end{bmatrix} \tag{3.14}$$

$$M_1(\mathbf{q})\,\ddot{\mathbf{q}}_{\boldsymbol{a}} + M_2(\mathbf{q})\,\ddot{\mathbf{q}}_{\boldsymbol{u}} + L_1(\mathbf{q},\dot{\mathbf{q}})\,\dot{\mathbf{q}}_{\boldsymbol{a}} + L_2(\mathbf{q},\dot{\mathbf{q}})\,\dot{\mathbf{q}}_{\boldsymbol{u}} + G_1(\mathbf{q}) = \boldsymbol{u} \tag{3.15}$$

$$M_3(\mathbf{q})\,\ddot{\mathbf{q}}_{\boldsymbol{a}} + M_4(\mathbf{q})\,\ddot{\mathbf{q}}_{\boldsymbol{u}} + L_3(\mathbf{q},\dot{\mathbf{q}})\,\dot{\mathbf{q}}_{\boldsymbol{a}} + L_4(\mathbf{q},\dot{\mathbf{q}})\,\dot{\mathbf{q}}_{\boldsymbol{u}} + G_2(\mathbf{q}) = \mathbf{0} \tag{3.16}$$

Since $M_4(\mathbf{q})$ is positive definite matrix,from equation 3.16 $\ddot{\mathbf{q}}_{\boldsymbol{u}}$ is given by

$$\ddot{\mathbf{q}}_{\boldsymbol{u}} = -M_4(\mathbf{q})^{\mathbf{-1}}\left(M_3(\mathbf{q})\,\ddot{\mathbf{q}}_{\boldsymbol{a}} + L_3(\mathbf{q},\dot{\mathbf{q}})\,\dot{\mathbf{q}}_{\boldsymbol{a}} + L_4(\mathbf{q},\dot{\mathbf{q}})\,\dot{\mathbf{q}}_{\boldsymbol{u}} + G_2(\mathbf{q})\right) \tag{3.17}$$

Substituting equation3.17 into equation3.15,the actuated state $\ddot{\mathbf{q}}_{\boldsymbol{a}}$can be obtained by solving equation 3.18

$$\overline{G}_1\ddot{\mathbf{q}}_{\boldsymbol{a}} + \overline{L}_1\dot{\mathbf{q}}_{\boldsymbol{a}} + \overline{L}_2\dot{\mathbf{q}}_{\boldsymbol{u}} + \overline{GG}_1 = u \tag{3.18}$$

or

$$\ddot{\mathbf{q}}_{\boldsymbol{a}} = -\overline{G}_1^{\,-1}\left(\overline{L}_1\dot{\mathbf{q}}_{\boldsymbol{a}} + \overline{L}_2\dot{\mathbf{q}}_{\boldsymbol{u}} + \overline{GG}_1\right) + \overline{G}_1^{\,-1}u$$

where,

$$\overline{G}_1 = M_1(\mathbf{q}) - M_2(\mathbf{q})M_4(\mathbf{q})^{-1}M_3(\mathbf{q})$$

$$\overline{L}_1 = L_1(\mathbf{q}) - M_2(\mathbf{q})M_4(\mathbf{q})^{-1}L_3(\mathbf{q})$$

$$\overline{L}_2 = L_2(\mathbf{q}) - M_2(\mathbf{q})M_4(\mathbf{q})^{-1}L_4(\mathbf{q})$$

$$\overline{GG}_1 = G_1(\boldsymbol{q}) - M_2(\boldsymbol{q})M_4(\boldsymbol{q})^{-1}G_2(\boldsymbol{q})$$

Similarly $M_1(\mathbf{q})$ is positive definite matrix, from equation 3.15 $\ddot{\mathbf{q}}_{\boldsymbol{a}}$ is given by

$$\ddot{\mathbf{q}}_{\boldsymbol{a}} = -M_1(\mathbf{q})^{\mathbf{-1}}\left(M_2(\mathbf{q})\ddot{\mathbf{q}}_{\boldsymbol{u}} + L_1(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_{\boldsymbol{a}} + L_2(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}_{\boldsymbol{u}} + G_1(\mathbf{q})\right) \tag{3.19}$$

Substituting equation3.19 into equation3.16,un-actuated state $\ddot{\mathbf{q}}_{\boldsymbol{u}}$ can be obtained by solving equation 3.20.

$$\overline{G}_2\ddot{\mathbf{q}}_{\boldsymbol{u}} + \overline{L}_3\dot{\mathbf{q}}_{\boldsymbol{a}} + \overline{L}_4\dot{\mathbf{q}}_{\boldsymbol{u}} + \overline{GG}_2 = Hu \tag{3.20}$$

or

$$\ddot{\mathbf{q}}_{\boldsymbol{u}} = -\overline{G}_2^{\,-1}\left(\overline{L}_3\dot{\mathbf{q}}_{\boldsymbol{a}} + \overline{L}_4\dot{\mathbf{q}}_{\boldsymbol{u}} + \overline{GG}_2\right) + \overline{G}_2^{\,-1}Hu$$

where,

$$\overline{G}_2 = M_4(\boldsymbol{q}) - M_3(\boldsymbol{q})M_1(\boldsymbol{q})^{-1}M_2(\boldsymbol{q})$$

$$\overline{L}_3 = L_3(\boldsymbol{q}) - M_3(\boldsymbol{q})M_1(\boldsymbol{q})^{-1}L_1(\boldsymbol{q})$$

$$\overline{L}_4 = L_4\left(\boldsymbol{q}\right) - M_3(\boldsymbol{q})M_1(\boldsymbol{q})^{-1}L_2(\boldsymbol{q})$$

$$\overline{GG}_2 = G_2\left(\boldsymbol{q}\right) - M_3(\boldsymbol{q})M_1(\boldsymbol{q})^{-1}G_1(\boldsymbol{q})$$

$$H = -M_3(\boldsymbol{q})M_1(\boldsymbol{q})^{-1}$$

In state space form equation and can be written as

$$\ddot{\mathbf{q}}_{\boldsymbol{a}} = f_1\left(\mathbf{q}, \dot{\mathbf{q}}\right) + \overline{G}_1^{\;-1}u \tag{3.21}$$

$$\ddot{\mathbf{q}}_{\boldsymbol{u}} = f_2\left(\mathbf{q}, \dot{\mathbf{q}}\right) + \overline{G}_2^{\;-1}Hu \tag{3.22}$$

where,$f_1\left(\mathbf{q}, \dot{\mathbf{q}}\right)$ and $f_1\left(\mathbf{q}, \dot{\mathbf{q}}\right)$ are nonlinear functions defined as shown bellow.

$$f_1\left(\mathbf{q}, \dot{\mathbf{q}}\right) = -\overline{G}_1^{\;-1}\left(\overline{L}_1\dot{\mathbf{q}}_{\boldsymbol{a}} + \overline{L}_2\dot{\mathbf{q}}_{\boldsymbol{u}} + \overline{GG}_1\right) \tag{3.23}$$

$$f_2\left(\mathbf{q}, \dot{\mathbf{q}}\right) = -\overline{G}_2^{\;-1}\left(\overline{L}_3\dot{\mathbf{q}}_{\boldsymbol{a}} + \overline{L}_4\dot{\mathbf{q}}_{\boldsymbol{u}} + \overline{GG}_2\right) \tag{3.24}$$

Further, the above equation can be reduced to an ordinary differential equation as follows:

$$\begin{aligned}
\dot{\underline{x}}_1 &= \underline{x}_2 \\
\dot{\underline{x}}_2 &= f_1\left(\mathbf{q}, \dot{\mathbf{q}}\right) + \overline{G}_1^{\;-1}u \\
\dot{\underline{x}}_3 &= \underline{x}_4 \\
\dot{\underline{x}}_4 &= f_2\left(\mathbf{q}, \dot{\mathbf{q}}\right) + \overline{G}_2^{\;-1}Hu
\end{aligned} \tag{3.25}$$

Where $\underline{x}_1, \underline{x}_2, \underline{x}_3, \underline{x}_4$ shows state vectors and given by: $\underline{x}_1 = \boldsymbol{q_a} = [\boldsymbol{x} \;\; l_1\;]^{\boldsymbol{T}}$, $\underline{x}_2 = \dot{\mathbf{q}}_{\boldsymbol{a}}$, $\underline{x}_3 = \boldsymbol{q_u} = [\theta_1 \;\; \theta_2\;]^{\boldsymbol{T}}$ and $\underline{x}_4 = \dot{\mathbf{q}}_{\boldsymbol{u}}$

3.5 Open loop system response

In order to examine the behavior of the created model, the dynamic model of the 2D DPOC is validated in this section using MATLAB/Simulink software. The same payload-lifting step input is employed with two distinct trolley-driving step inputs. Table 3.2 provides the model parameters of the 2D DPOC with variable cable length, which are taken from [7] for both the closed-loop simulation in chapter5 and the open-loop simulation in this section.

Open loop Simulink model:- Figure 3.3 shows a nonlinear dynamic model of the 2D DPOC built-in Simulink environment.

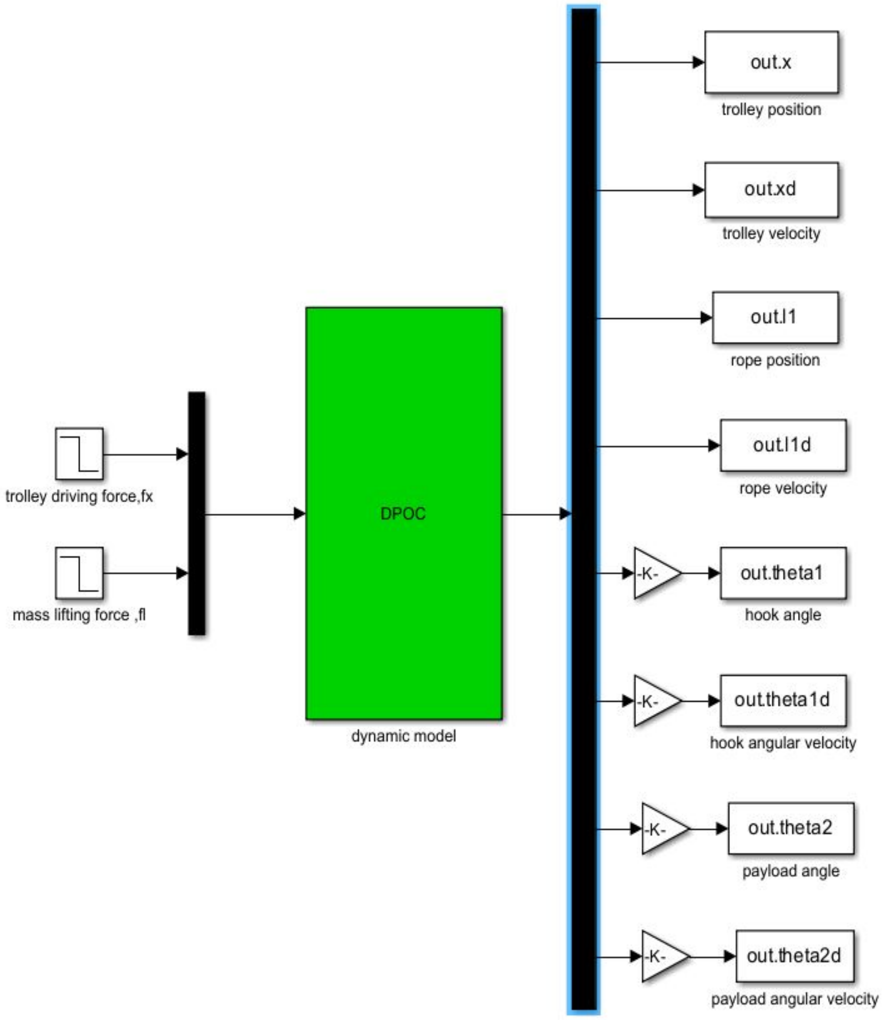


Figure 3.3: Open loop simulation block diagram

Open loop simulation is done for model parameter given in following table[7].

Table 3.2: Model parameter values

Symbol	Parameter	Value
m	Trolley mass	9.2 kg
m_1	Hook mass	2 kg
m_2	Payload mass	1 kg
l_2	Load/mass pendulum length	0.6m
g	Gravity acceleration	9.8m/s

Case 1: step inputs of 10 magnitudes for trolley driving and lifting force as shown in figures 3.4 a and b.

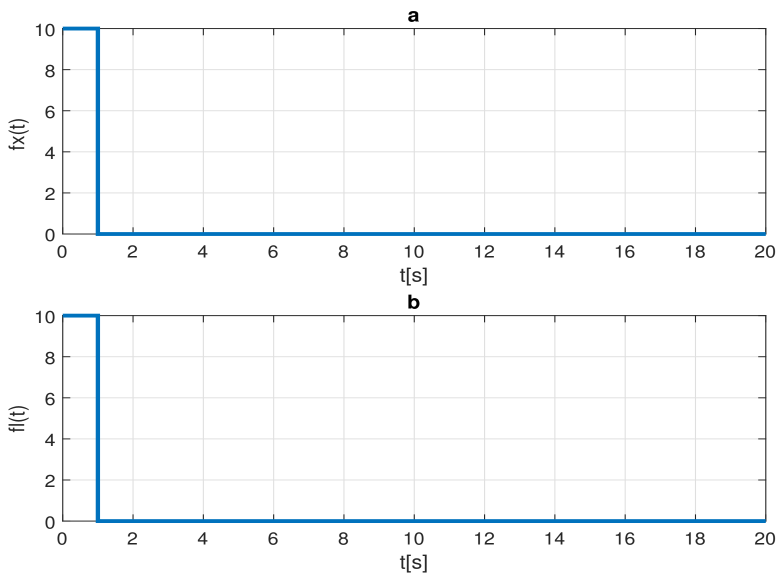


Figure 3.4: Step inputs:a) trolley driving input b) payload hoisting input

Figures 3.5 and 3.6 show the open loop response of 2D DPOC for case 1 step inputs. From 3.5 a and b, it is observed that the trolley displacement is unbounded with a constant speed of about 0.8 since there is no friction force to stop it. In this simulation, it is assumed that the payload is lowered to certain heights so that the length of the cable does not go to infinity. The applied hoisting step input with a zero initial value lowers the payload to some height and stops as shown in 3.5 c and d. Both the hook and the payload oscillate with a maximum angle of about 5° and

10°, respectively, as shown in 3.6 a, b, c, and d.

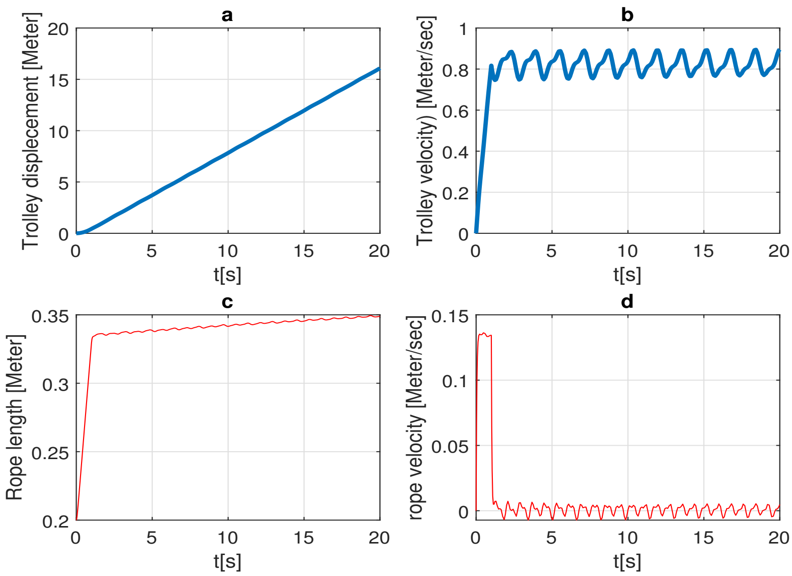


Figure 3.5: a) Trolley position b) trolley velocity c) rope length and d) rope velocity

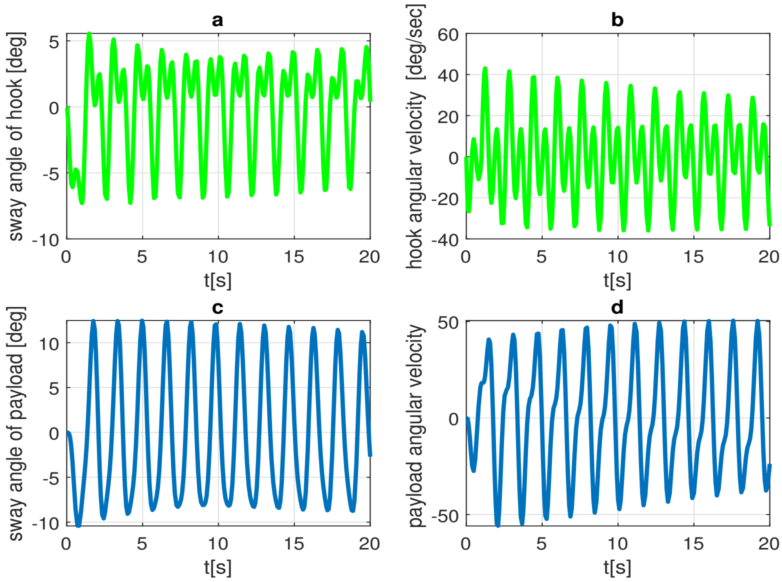


Figure 3.6: a) Hook swing angle b) hook angular velicty velocity c) payload swing angle and d)payload angular velocity

Figures 3.5 and 3.6 show the following observations for the given input forces in case 1.

1. When the friction force between the cart and the horizontal plane (rail) is assumed

to be zero, the cart position goes to infinity, even for a bounded step input. However, the velocity of the trolley is constant after some time, which is between 0.79 and 8.5 for this simulation case.

2. The objective of the hoisting force is to lift or lower the object to the desired height, which is then transported to its final destination. In addition, the objective of this open-loop simulation is to understand the behavior of 2D DPOC for a given uncontrolled force. From figure 3.5c, it is observed that the cable length between the trolley and hook is increased from its initial position of 0.2m to some constant value when the hoisting step input given in figure 3.4 b is applied, and friction force is assumed to exist to prevent the cable height from going to infinity. It should be understood that the cable length between the hook and the payload is assumed to be constant in cases 1 and 2.

3. It can be seen from figure 3.6 that the hook swing angle and the load swing angle oscillate with high frequency and high angular speed.

Case 2: In this case, step inputs shown in figures 3.7 a and b are applied, and the simulation results are illustrated in figures 3.8 and 3.9.

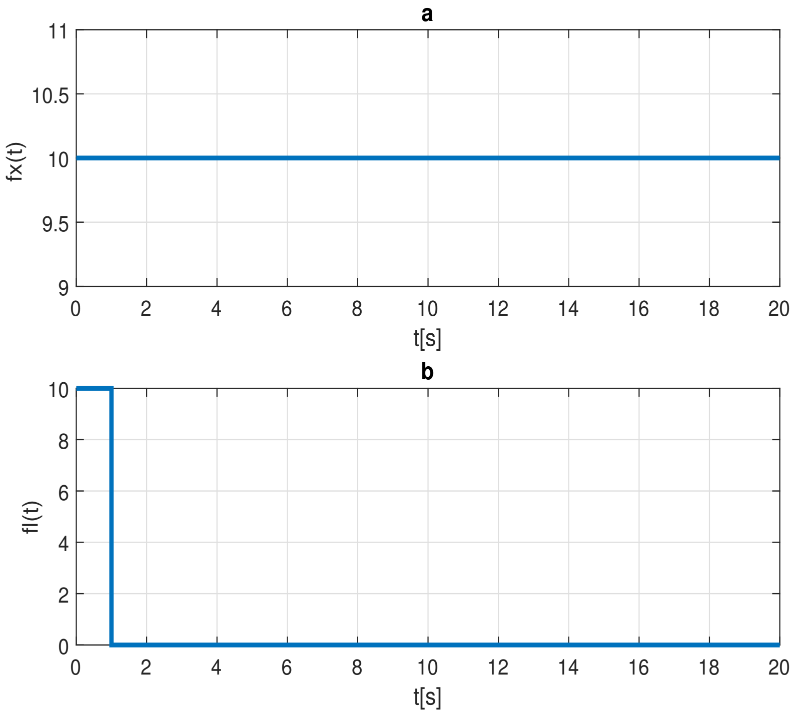


Figure 3.7: Step inputs for case 2 a) trolley driving force b) hoisting force

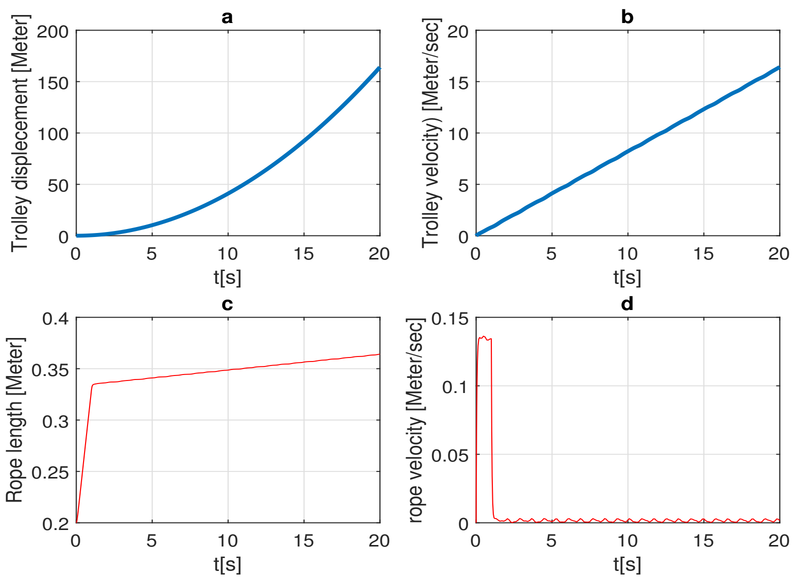


Figure 3.8: a) Trolley position b) trolley velocity c) rope length and d) rope velocity

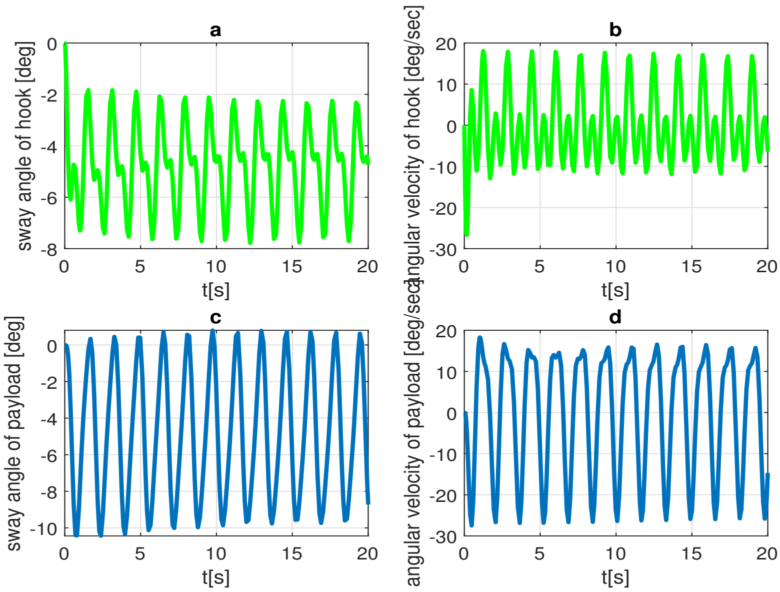


Figure 3.9: a) Hook swing angle b) hook angular velicty velocity c) payload swing angle and d)payload angular velocity

As observed from figures 3.8 and 3.9, open loop simulations are different from case 1. In figures 3.8 a and b, both trolley displacement and velocity are unbounded because of the unbounded trolley driving step input given in figure 3.7 a. In this case, the same assumption is taken for payload hoisting force. Again, in this case, both the hook and the payload oscillate with a high frequency and maximum angle of about -7 deg and -10 deg, as shown in refreff3.9 a and b.

Case 3: Finally, to show the effect of payload suspending cable length (between hook and payload) on the dynamic model, the step inputs given in case 2 are applied with payload suspending cable length changed from 0.6 to 4m. The results are illustrated in figures 3.10 and 3.11. From these figures, the open loop response for trolley position and cable length is the same as in case 2. However, the oscillation frequency of the hook and the payload increases as the payload suspending cable length increases.

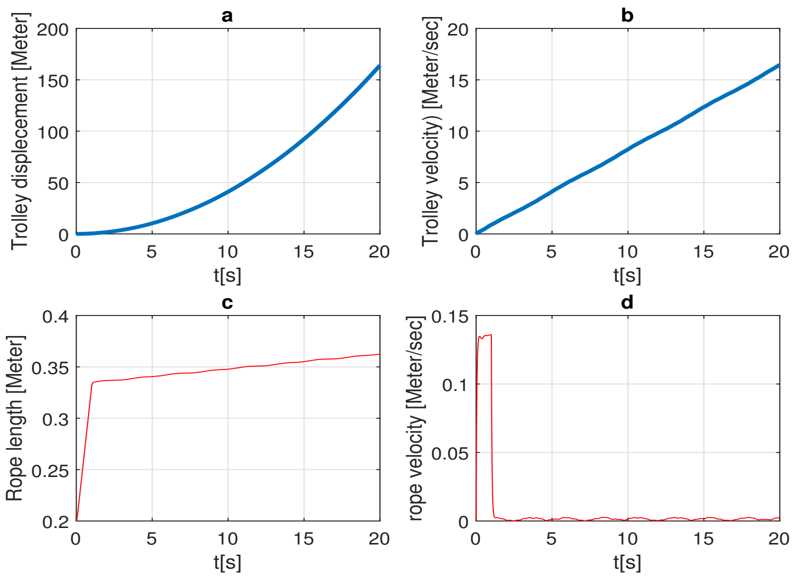


Figure 3.10: a) Trolley position b) trolley velocity c) rope length and d) rope velocity

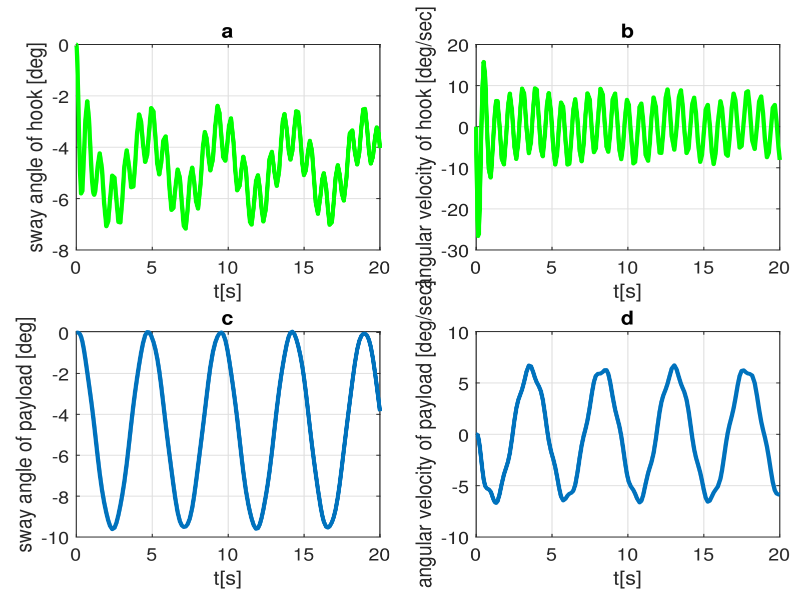


Figure 3.11: a) Hook swing angle b) hook angular velicty velocity c) payload swing angle and d)payload angular velocity

Therefore,the derived model is acceptable since the result obtained is almost similar to the experimentally verified model shown in some published papers, so it can be used for control design.

Chapter 4

RBFNN Based HSMC Design

4.1 Introduction

As was indicated in the earlier chapters, the primary goal of the thesis is to construct an adaptable hierarchical sliding mode controller (RBFNNAHSMC) based on neural networks in order to place the overhead crane at the desired location with the least amount of sway angles caused by the load and the hook. One benefit of developing neural network-based sliding mode control for crane systems is that it can lessen the sliding mode controller's deterministic model dependency. SMC requires a precise mathematical representation of the system, with all model parameters understood. However, as system modeling involves many assumptions and may include behaviors that are not described, the representation of the physical system in a mathematical model and the actual system cannot be identical at all. Furthermore, because of system disruptions during actual operation, crane specifications are often ambiguous. Therefore, utilizing RBFNN, the crane's unknown nonlinear dynamics component can be approximated online. Neural network-based SMC does not require such precise model information.

Over all block diagram and methodology for ARBFNN-HSMC:- Figures4.1 and 4.1 show the overall controller block diagram and steps to design ARBFNN-based HSMC, respectively.

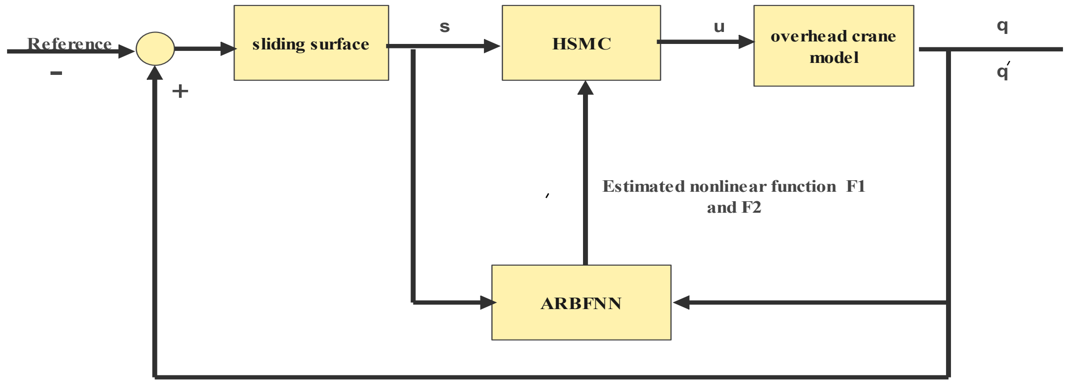


Figure 4.1: Over all block diagram for Proposed method

A 2D DPOC system's suggested control scheme is depicted in the block diagram above. The input of an adaptive RBFNN is the system states. Based on the Lyapunov stability criterion, sliding surfaces are used to develop its updating law. Next, using ARBFNN to identify an online system model, HSMC is built, and reference and feedback signals are used to determine the error. The feedback signals, which are computed from the proposed controller and include trolley position, cable length, hook swing angle, and payload swing angle, have an initial condition. RBFNN modifies its weight in response to the error signal until the target is achieved. The time derivative of each location signal is used to obtain the velocity feedback signals needed for simulation.In a real application, a sensor should be used to measure these feedback signals.

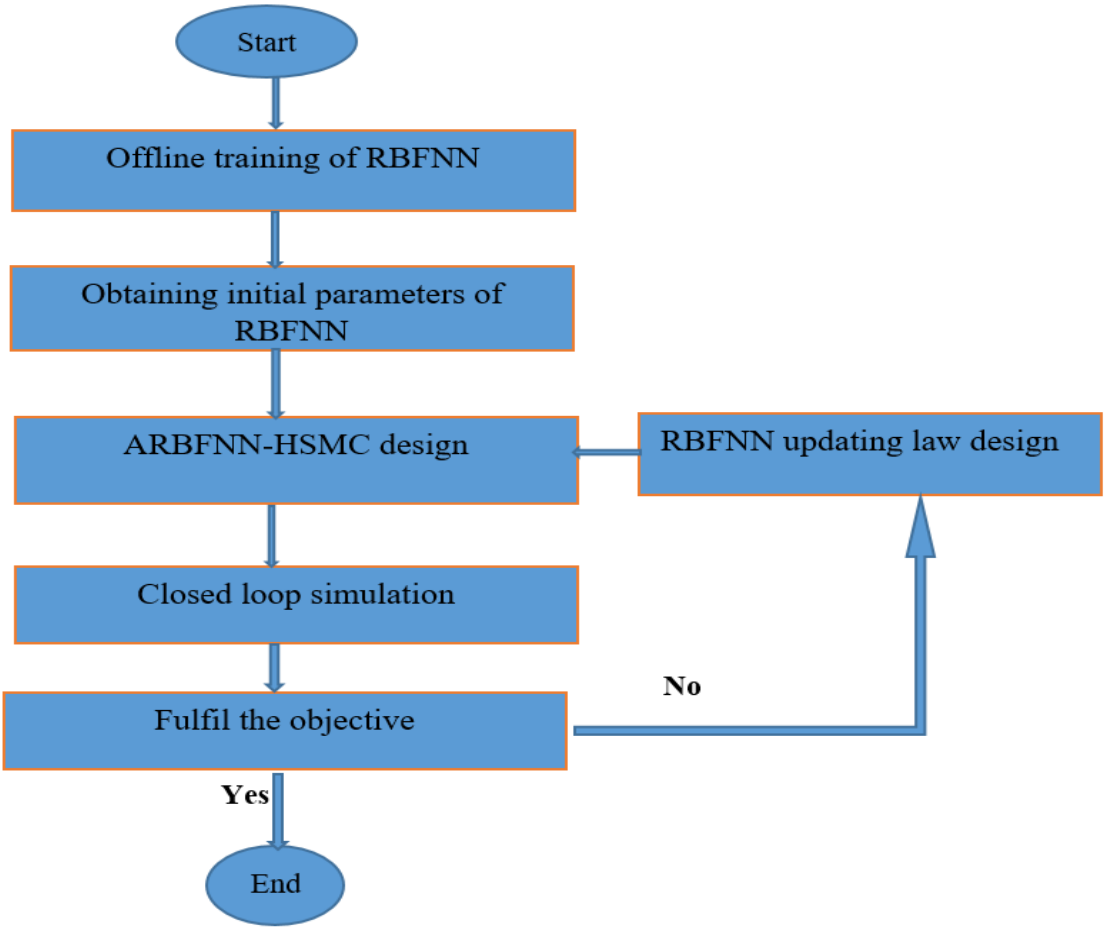


Figure 4.2: Methodology to design ARBFNN-HSMC

4.2 Hierarchical Sliding Mode controller(HSMC) Design

Even in situations when the system is unknown, SMC is the most reliable option and the most robust control solution for an underactuated system. But there's a difficulty with the way overhead cranes are controlled. To ensure the stability of both actuated and unactuated subsystems, for example, sliding surface definition is necessary when the number of control inputs is less than the number of actuators[38]. Hierarchical sliding mode control is a highly effective approach for controlling overhead cranes in this situation. The HSMC technique focuses on the hierarchical structure of the sliding surfaces and creates the control law based on the hierarchical sliding surfaces, as described in [5]. A suitable control rule is created by creating first-order sliding surfaces, which are built from a linear combination of state variables, and then higher-order sliding surfaces, which are created from a linear combination of first-order sliding surfaces.

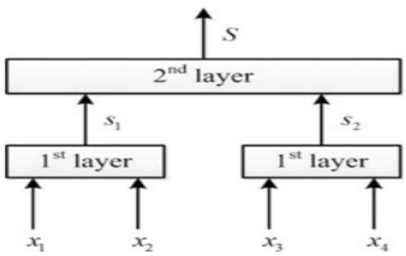


Figure 4.3: Hierarchical sliding surface

4.2.1 HSMC design for 2D DPOC

In this section, HSMC is designed based on state space equations of 2D DPOC developed in equation3.25 in chapter3 According to the HSMC design law for under-actuated systems, the two first-order sliding surfaces are defined based on the error dynamics of the actuated and actuated states of the overhead crane, and then the total second-order sliding surface is formulated by a linear combination of the two second-order sliding surfaces.

Controller design:-

Let $q_{\mathbf{ad}}=\begin{bmatrix} \boldsymbol{x_d} & \mathbf{l_{1d}} \end{bmatrix}^{\boldsymbol{T}}$ and $q_{\mathbf{ud}}=\begin{bmatrix} \mathbf{0} & \mathbf{0} \end{bmatrix}^{\boldsymbol{T}}$ be desired reference signals of actuated and unactuated states respectively.Then, the corresponding errors are defined as follow.

$$\mathbf{e}_{\boldsymbol{a}} = \boldsymbol{q_a} - \boldsymbol{q_{ad}}, \mathbf{e}_{\boldsymbol{u}} = q_u - q_{\boldsymbol{ud}} \tag{4.1}$$

Let us define two 1^{st}level sliding surfaces for both actuated and un-actuated states as follows:

$$\mathbf{s_1} = \dot{\boldsymbol{e}}_{\boldsymbol{a}} + \boldsymbol{\alpha}_1\mathbf{e}_{\boldsymbol{a}}, \mathbf{s_2} = \dot{\boldsymbol{e}}_{\boldsymbol{u}} + \boldsymbol{\alpha}_2\mathbf{e}_{\boldsymbol{u}} \tag{4.2}$$

where $\boldsymbol{\alpha_1}$ and $\boldsymbol{\alpha_2}$ are positive constant parameters and given by

$\boldsymbol{\alpha_1 = diag(\alpha_{11}\ , \alpha_{12})}$ and $\boldsymbol{\alpha_2 = diag(\alpha_{21}\ , \alpha_{22})}$

The second-level sliding surface for the whole system is then obtained by aggregating the two first-level sliding surfaces.

$$\boldsymbol{S} = \lambda_\mathbf{1}\mathbf{s_1} + \lambda_\mathbf{2}\mathbf{s_2} \tag{4.3}$$

Where $\lambda_\mathbf{1} = \boldsymbol{diag}(\lambda_{\mathbf{11}}, \lambda_{\mathbf{12}})$ and $\lambda_\mathbf{2} = \boldsymbol{diag}(\lambda_{\mathbf{21}}, \lambda_{\mathbf{22}})$ are matrices of constant gain. The stability of the second-level sliding surface can be ensured by designing an appropriate HSMC legislation. Siding mode control equivalent control is divided into two components. The system states are moved toward the sliding surface by means of the switching control portion, and the system states are maintained on the sliding surface by means of the corresponding control law. The following is a mathematical explanation for this:.

$$u = u_{\boldsymbol{eq}} + u_{\boldsymbol{sw}} \tag{4.4}$$

To design SMC, let us take the time derivative of the of the second-level sliding surface.

$$\dot{\mathrm{S}} = \lambda_1 \dot{\boldsymbol{s}}_1 + \lambda_2 \dot{\boldsymbol{s}}_2 = \lambda_1 \left(\boldsymbol{\alpha_1 \dot{e}_a} + \boldsymbol{\ddot{q}_a} - \boldsymbol{\ddot{X}_{ad}}\right) + \lambda_2 \left(\boldsymbol{\alpha_2 \dot{e}_u} + \boldsymbol{\ddot{q}_u}\right) \tag{4.5}$$

Substituting for $\ddot{q}_a$ and $\ddot{q}_u$ from equation 3.21, equation4.6 is obtained.

$$\dot{\mathrm{S}} = \left(\lambda_1 \boldsymbol{\alpha_1 \dot{e}_a} + \lambda_1 f_1(\mathbf{q}, \dot{\mathbf{q}}) + \lambda_1 \overline{G}_1^{-1} u - \lambda_1 \boldsymbol{\ddot{X}_{ad}}\right) + \left(\lambda_2 \boldsymbol{\alpha_2 \dot{e}_u} + \lambda_2 f_2(\mathbf{q}, \dot{\mathbf{q}}) + \lambda_2 \overline{G}_2^{-1} H u\right) \tag{4.6}$$

If $\boldsymbol{V} = \frac{1}{2}\boldsymbol{S^2}$ is selected as a candidate Lyapunov function,after some calculation its time derivatives is given in equation 4.7

$$\dot{\mathrm{V}} = \boldsymbol{S^T}\dot{\mathrm{S}} = \boldsymbol{S^T}\left(\lambda_1\left(\boldsymbol{\alpha_1 \dot{e}_a} + f_1 + \overline{G}_1 u - \boldsymbol{\ddot{q}_{ad}}\right) + \lambda_2\left(\boldsymbol{\alpha_2 \dot{e}_u} + f_2 + \overline{G}_2 H u\right)\right)$$

$$\dot{\mathrm{V}} = \boldsymbol{S^T}\dot{\mathrm{S}} = \boldsymbol{S^T}(\lambda_1\left(\boldsymbol{\alpha_1 \dot{e}_a} + f_1 + -\boldsymbol{\ddot{q}_{ad}}\right) + \lambda_2\left(\boldsymbol{\alpha_2 \dot{e}_u} + f_2\right) + \left(\lambda_1 \overline{G}_1 + \lambda_2 \overline{G}_2 H\right)\boldsymbol{u})$$

$$\dot{\mathrm{V}} = \boldsymbol{S^T}\dot{\mathrm{S}} = \boldsymbol{S^T}(\lambda_1\left(\boldsymbol{\alpha_1 \dot{e}_a} + f_1 - \boldsymbol{\ddot{q}_{ad}}\right) + \lambda_2\left(\boldsymbol{\alpha_2 \dot{e}_u} + f_2\right) + \left(\lambda_1 \overline{G}_1 + \lambda_2 \overline{G}_2 H\right)(u_{\boldsymbol{eq}} + u_{\boldsymbol{sw}})) \tag{4.7}$$

To ensure the stability of the second-level sliding surface, the following equations are defined from equation 4.7, and equivalent and switching control laws are derived and given in equations 4.8 and 4.9, respectively.

$$\lambda_1\left(\boldsymbol{\alpha_1 \dot{e}_a} + f_1 - \boldsymbol{\ddot{q}_{ad}}\right) + \lambda_2\left(\boldsymbol{\alpha_2 \dot{e}_u} + f_2\right) + \left(\lambda_1 \overline{G}_1 + \lambda_2 \overline{G}_2 H\right) u_{\boldsymbol{eq}} = \boldsymbol{0}$$

$$\left(\lambda_1 \overline{G}_1 + \lambda_2 \overline{G}_2 H\right) u_{\boldsymbol{sw}} + \boldsymbol{k_1}(\boldsymbol{s}) + \boldsymbol{k_2(sign(s))} = \boldsymbol{0}$$

$$u_{\boldsymbol{eq}} = -\left(\lambda_1 \overline{G}_1 + \lambda_2 \overline{G}_2 H\right)^{-1}(\lambda_1\left(\boldsymbol{\alpha_1 \dot{e}_a} + f_1 - \boldsymbol{\ddot{q}_{ad}}\right) + \lambda_2\left(\boldsymbol{\alpha_2 \dot{e}_u} + f_2\right)) \tag{4.8}$$

$$u_{\boldsymbol{sw}} = -\left(\lambda_1 \overline{G}_1 + \lambda_2 \overline{G}_2 H\right)^{-1}(\boldsymbol{k_1}(\boldsymbol{s}) + \boldsymbol{k_2(sign(s))}) \tag{4.9}$$

Where $\boldsymbol{k_1}$ ***and*** $\boldsymbol{k_2}$ are positive control gain matrices.

Then the total HSMC law is obtained from the summation of the equivalent and switching controls derived above.

$$u = u_{\boldsymbol{eq}} + u_{\boldsymbol{sw}}$$

$$u = -\left(\lambda_{\mathbf{1}}\overline{G}_1 + \lambda_{\mathbf{2}}\overline{G}_2 H\right)^{-1}\left(\lambda_{\mathbf{1}}\boldsymbol{\alpha_1}\dot{\boldsymbol{e}}_{\boldsymbol{a}} + \lambda_{\mathbf{1}}f_1 - \lambda_{\mathbf{1}}\ddot{\boldsymbol{q}}_{\boldsymbol{ad}} + \lambda_{\mathbf{2}}\boldsymbol{\alpha_2}\dot{\boldsymbol{e}}_{\boldsymbol{u}} + \lambda_{\mathbf{2}}f_2 + \boldsymbol{k_1 S} + \boldsymbol{k_2 sign}\left(\boldsymbol{S}\right)\right) \tag{4.10}$$

In equation 4.10, the two nonlinear functions f_1 and f_2, which are defined in 3.23 and 3.24 respectively, contain modeling information such as $m, m_{\mathbf{1}}, m_{\mathbf{2}}, l_{\mathbf{1}}, l_{\mathbf{2}}, \theta_1$ *and* θ_2 . Substituting the above total control law into equation 4.7, it can be shown that the designed control law guarantees the stability of the second-level sliding surface as shown in equation 4.11.

$$\dot{\mathrm{V}} = \boldsymbol{S^T}\dot{\mathrm{S}} = \boldsymbol{S^T}\left(-\boldsymbol{k_1 S} - \boldsymbol{k_2 sign}\left(\boldsymbol{S}\right)\right) = -\boldsymbol{k_1 S^T S} - \boldsymbol{k_2 norm(S)} \leq \mathbf{0} \tag{4.11}$$

4.3 Radial Basis Function Neural Network based Adaptive Hierarchical Sliding Mode Control (RBFNN-AHSMC)

The model-based HSMC method designed above can be effectively applied if the model information (parameters) in the nonlinear functions f_1 and f_2 are certain. However, the dynamic model of 2D DPOC developed in 3.9 to 3.12 is ideal, and it is the nominal model of the system.In a genuine dynamical system, uncertainties and external disturbances cannot be avoided due to modeling errors and environmental factors. According to [5], overhead crane systems are really frequently operated under unpredictable situations including payload mass changes, unmodeled dynamics, skidding, and slippage.As a result, the physical system of the crane cannot be adequately represented by the dynamic model of 2D DPOC created in Chapter 3. These distinguish the nominal model (parameters) from the real plant model (parameters). Finding or estimating a complex and uncertain nonlinear component of the online crane dynamic model is a good idea. As a result, HSMC can only use a portion of the dynamic model, which will make the controller more reliable when used with a real crane.

4.3.1 Radial basis function neural network (RBFNN)

The primary use of neural networks is the identification of models for unknown non-linear processes. The neural network model's output response can be altered to meet predetermined system parameters by varying the weights of the neural network to different values. A particular type of artificial neural network, the RBF network offers the benefits of a speedier algorithm, better approximation of a nonlinear connection, and a simpler topology.The input layer, the hidden layer, and the output layer make up the three layers of an RBF neural network. A radial basis function drives the activity of neurons in the buried layer. The array of computational components known as hidden nodes makes up the hidden layer. Every hidden node has a parameter vector called the center c, which has the same dimensions as the input vector x.The diagram presented in figure4.4 illustrates the usual three-layer RBF neural network structure:

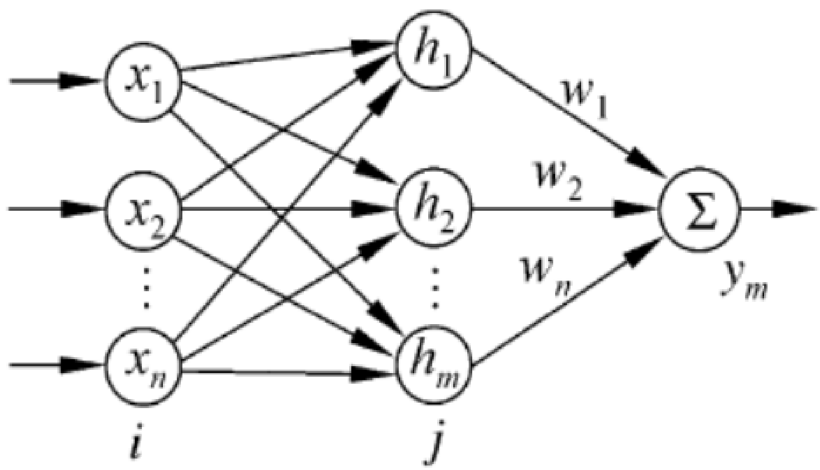


Figure 4.4: RBFNN architecture

Let x be the input vector of RBFNN,then the approximation algorithm used by

RBFNN to estimate uncertain nonlinear function $f(x)$ is given by:

$$f(x) = W^T h(x) + \varepsilon \tag{4.12}$$

$h(x) = e^{\left(-\frac{\left\|x-c_j\right\|^2}{2b^2{}_j}\right)}$ for j=1, 2,. . . . . . .m

where b_j denotes a positive scalar called a width and m notes the number of hidden nodes and $\|x - c_j\|$ is the Euclidean distance between the center and the network input vector x.

$h = [h_1 \quad h_1 \ \ . \ . \ .h_n]^T$ is output of Gaussian function, W is the neural network weights, ε is approximation error of neural network. The output of RBF neural network is then given by

$$\hat{f}(x) = \hat{w}^T h(x)$$

In the 2D DPOC overhead system, the two dynamic models f_1 and f_2, as explained above, are nonlinear, unknown, and uncertain, which is not trivial to practically determine. So two RBFNNs are constructed to estimate these two nonlinear uncertain functions, as shown below. Since the nonlinear functions going to be estimated are expressed in terms of system outputs and their derivatives, the inputs for the RBFNN are $q = [x \ \ l_{\mathbf{1}} \ \ \theta_1 \ \ \theta_2\ ]^T$ and $qd = \dot{q} = [\ \dot{x} \quad \dot{l}_1 \ \ \dot{\theta}_1 \ \ \dot{\theta}_2]^T$ for the position vector and the velocity vector, respectively.

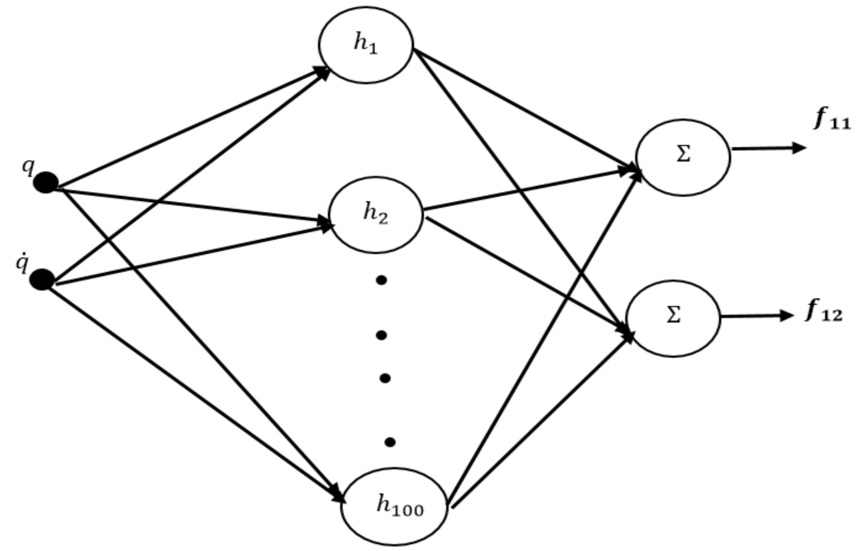


Figure 4.5: RBFNN used for f1 estimation

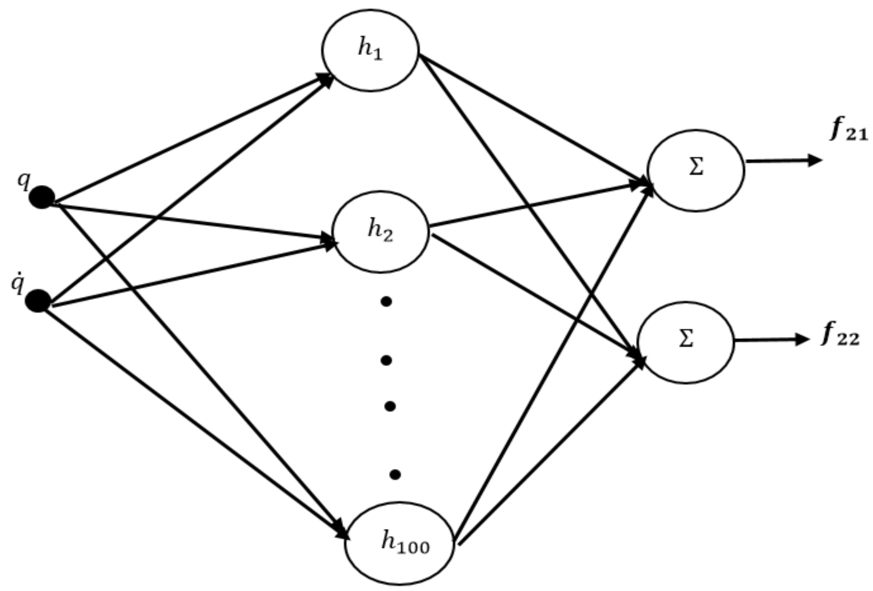


Figure 4.6: RBFNN used for f2 estimation

The two nonlinear functions given by equations 3.23 and 3.24 can be approximated by RBFNNs as given in equations 4.13 and 4.14 respectively.

$$\hat{f}_1 = \left[\hat{W}_1\right]^T h(x) \tag{4.13}$$

$$\hat{f}_2 = \left[\hat{W}_2\right]^T h(x) \tag{4.14}$$

where $\hat{W}_1$ and $\hat{W}_1 \in R^{m\times 2}$ are the adaptive weight matrices, which is obtained by the use of the Lyapunov stability analysis in the next section.

4.3.2 AHSMC Design Based on a Radial Basis Function Neural Network

To design ARBFNN based HSMC,the new Lyapunov candidate fanction is slected as in equation4.15 From equation 4.10 HSMC law is defined as:

$$V = \frac{1}{2}\, S^T\, S + \frac{1}{2} tr(\widetilde{W}_a^T\, \gamma_1^{-1}\, \widetilde{W}_a) + \frac{1}{2} tr(\widetilde{W}_u^T\, \gamma_2^{-1}\, \widetilde{W}_u) \tag{4.15}$$

where $\widetilde{\mathrm{W}}_{\mathrm{a}} = \mathrm{W}_{\mathrm{a}} - \widehat{\mathrm{W}}_{\mathrm{a}}$ and $\widetilde{\mathrm{W}}_{\mathrm{u}} = \mathrm{W}_{\mathrm{u}} - \widehat{\mathrm{W}}_{\mathrm{u}}$ are error weights and tr is trace of a matrix. After some calculation ,the time derivative of the above Lyapunov function obtained as shown bellow.

$$\dot{V} = S^T\, \dot{S} - tr(\widetilde{W}_a^T\, \gamma_1^{-1}\, \dot{\widehat{W}}_a) - tr(\widetilde{W}_u^T\, \gamma_2^{-1}\, \dot{\widehat{W}}_u) \tag{4.16}$$

Substituent equation 4.6 into equation 4.16 with RBFNN estimated estimated nonlinear functions,equation 4.17 is derived, from which ARBFNN based HSMC obtained as given in 4.18

$$\begin{aligned} \dot{V} &= S^T(\lambda_1\alpha_1\dot{e_a} + \lambda_1\hat{f}_1 + \lambda_2\alpha_2\dot{e_u} + \lambda_2\hat{f}_2 - \lambda_1\ddot{q_{ad}} + (\lambda_1\bar{G}_1^{-1} + \lambda_2\bar{G}_2^{-1}H)u \\ &+ \lambda_1(W_a^T h + \varepsilon_a) + \lambda_2(W_u^T h + \varepsilon_u)) - tr(W_a^T\gamma_1^{-1}\dot{\hat{W}}_a) - tr(W_u^T\gamma_2^{-1}\dot{\hat{W}}_u) \end{aligned} \tag{4.17}$$

$$u = -\left(\lambda_1\overline{G}_1 + \lambda_2\overline{G}_2 H\right)^{-1}\left(\lambda_1\boldsymbol{\alpha_1}\dot{\boldsymbol{e}}_{\boldsymbol{a}} + \lambda_1 f_1 - \lambda_1\ddot{\boldsymbol{q}}_{\boldsymbol{ad}} + \lambda_2\boldsymbol{\alpha_2}\dot{\boldsymbol{e}}_{\boldsymbol{u}} + \lambda_2 f_2 + \boldsymbol{k_1 S} + \boldsymbol{k_2 sign}\left(\boldsymbol{S}\right)\right) \tag{4.18}$$

Updating law from Lyapunov stability analysis

Substituting control from equation4.18 into equation4.17 and simplifying,

$\dot{V}$ is obtained as

$$\begin{aligned}\dot{V} =& - S^T k_1 S - S^T k_2\, sign(S) + S^T \lambda_1 \widetilde{W}_a^T h + S^T \lambda_1 \varepsilon_a + \\ & S^T \lambda_2 \widetilde{W}_u^T h + S^T \lambda_2 \varepsilon_u - tr(\widetilde{W}_a^T \gamma_1^{-1} \dot{\widehat{W}}_a) - tr(\widetilde{W}_u^T \gamma_2^{-1} \dot{\widehat{W}}_u)\end{aligned} \tag{4.19}$$

where,

$$S^T \lambda_1 \widetilde{W}_a^T h = tr(\widetilde{W}_a^T h\, S^T \lambda_1) \tag{4.20}$$

Then,after simplification, $\dot{V}$ written as given in equation4.22 and updating laws based on Lyapunov stability is derived and given in equations 4.23 and 4.24 respectively.

$$S^T \lambda_2 \widetilde{W}_u^T h = tr(\widetilde{W}_u^T h\, S^T \lambda_2) \tag{4.21}$$

$$\begin{aligned}\dot{V} =& - S^T k_1 S - S^T k_2\, sign(S) + tr(\widetilde{W}_a^T (h\, S^T \lambda_1 - \gamma_1^{-1} \dot{\widehat{W}}_a)) + \\ & tr(\widetilde{W}_u^T (h\, S^T \lambda_2 - \gamma_2^{-1} \dot{\widehat{W}}_a))\end{aligned} \tag{4.22}$$

$$\dot{\widehat{W}}_a = \gamma_1 (h\, S^T \lambda_1 - \eta_1 \, ||S||\, \widehat{W}_a) \tag{4.23}$$

$$\dot{\widehat{W}}_u = \gamma_2 (h\, S^T \lambda_2 - \eta_2 \, ||S||\, \widehat{W}_u) \tag{4.24}$$

Based on the above updating laws and if conditions 4.25 and 4.26 are satisfied,the system will be stable according to Lyapunov stability condition.

$$||S|| \geq \max(\frac{\eta_1\, ||S||\, ||\, W_a\, ||_F^2 - 4\, \varepsilon_{aN}}{4\, \eta_{1\,\min}}, \frac{\eta_2\, ||S||\, ||\, W_u\, ||_F^2 - 4\, \varepsilon_{uN}}{4\, \eta_{2\,\min}}) \tag{4.25}$$

where η_1 and η_2 are the smallest eigenvalues of D_1 and D_2 respectively for which D_1 and D_2 are two arbitrary positive definite matrices that satisfies equation4.26,

$$k_1 = D_1 + D_2 \tag{4.26}$$

Substituting the above updating laws in equation 4.22,the following equation is obtained.

$$\begin{aligned}\dot{V} = &- S^T k_1 S - S^T k_2\, sign(S) + S^T(\lambda_1 \varepsilon_a + \lambda_2 \varepsilon_u) + \\ &\eta_1 ||S|| tr(\widetilde{W}_a^T(W_a - \widetilde{W}_a)) + \eta_2 ||S|| tr(\widetilde{W}_u^T(W_u - \widetilde{W}_u))\end{aligned} \tag{4.27}$$

Using Cauchy-Schwarz inequality,one can write:

$$\begin{aligned} tr(\widetilde{W}_a^T(W_a - \widetilde{W}_a)) \leq ||\, \widetilde{W}_a \,||_F \, ||\, W_a \,||_F - ||\, \widetilde{W}_a \,||_F^2 \\ tr(\widetilde{W}_u^T(W_u - \widetilde{W}_u)) \leq ||\, \widetilde{W}_u \,||_F \, ||\, W_u \,||_F - ||\, \widetilde{W}_u \,||_F^2 \end{aligned}$$

From which the following inequality is obtained.

$$\begin{aligned}\dot{V} \leq &- S^T k_1 S - S^T k_2\, sign(S) + (S^T \lambda_1 \varepsilon_a + \tfrac{1}{4}\eta_1 ||S|| \, ||\, W_a \,||_F^2) + \\ &(S^T \lambda_2 \varepsilon_u + \tfrac{1}{4}\eta_2 ||S|| \, ||\, W_u \,||_F^2) - \eta_1 ||S|| \, (||\, \widetilde{W}_a \,||_F - \tfrac{1}{2} ||\, W_a \,||_F)^2 - \\ &\eta_2 ||S|| \, (||\, \widetilde{W}_u \,||_F - \tfrac{1}{2} ||\, W_u \,||_F)^2\end{aligned} \tag{4.28}$$

From equation 4.26,since D_1 and D_2 matrices are positive definite,the inequality of quadratic form can be applied as given in equation4.29

$$\phi_{q\max} ||x||^2 \geq x^T Q x \geq \phi_{q\min} ||x||^2 \geq 0; \forall x \neq 0 \tag{4.29}$$

where $\phi_{q\max}$ and $\phi_{q\min}$ are the largest and the smallest eigenvalues of the square matrix Q.Therefor,equation 4.30 can be deduced from the above condition.

$$S^T k_1 S = S^T(D_1 + D_2)S \geq \eta_{1min} ||S||^2 + \eta_{2min} ||S||^2 \tag{4.30}$$

Since equation 4.31 is true,equation4.28 is simplified as given in4.32 and if 4.25 is satisfied,the system is stable.

$$\begin{aligned}\varepsilon_{aN} &= \inf_{\varepsilon_a > 0, \varepsilon_a \in \partial(0)} ||\, \lambda_1 \varepsilon_a \,|| \leq \lambda_1 \varepsilon_a \\ \varepsilon_{uN} &= \inf_{\varepsilon_u > 0, \varepsilon_u \in \partial(0)} ||\, \lambda_2 \varepsilon_u \,|| \leq \lambda_2 \varepsilon_u\end{aligned} \tag{4.31}$$

$$\dot{V} = -S^T k_2 \, sign(S) - \eta_1 \, ||S|| \, (|| \, \widetilde{W}_a \, ||_F - \tfrac{1}{2} \, || \, W_a \, ||_F)^2 - \eta_2 \, ||S|| \, (|| \, \widetilde{W}_u \, ||_F - \tfrac{1}{2} \, || \, W_u \, ||_F)^2 - ||S||(\varepsilon_{aN} - \tfrac{1}{4} \, \eta_1 \, || \, W_a \, ||_F^2 + \eta_{1min} \, ||S||) - ||S||(\varepsilon_{uN} - \tfrac{1}{4} \, \eta_1 \, || \, W_u \, ||_F^2 + \eta_{2min} \, ||S||) \tag{4.32}$$

Chapter 5

Results and discussions

5.1 Introduction

Results from simulations performed with MATLAB/SIMULINK software are presented in this chapter.The aim is to demonstrate the effectiveness of the control action aimed to suppress the two swing angles and converge the trolley and hoist cable responses to the required steady states. A comparison is made between the suggested ARBFNN-HSMC scheme and HSMC, particularly in cases when there is a significant variance in the model parameters. We take and discuss a variety of simulated instances to demonstrate the robustness of the suggested control mechanism. The 2D DPOC system's physical parameters are extracted from [7] and are provided in table3.2.

5.2 Simulation block diagram

Figures 5.1 and 5.2 show simulation block diagrams of HSMC without RBFNN and ARBFNN-based HSMC, respectively. In this thesis, the simulation results of both controllers are displayed and compared. From figure 5.2, the overall simulation block diagram contains four main components: the plant dynamic model, the controller, RBFNN, and error dynamics. For online nonlinear function estimation, two RBFNNs with 100 hidden neurons are selected. To facilitate estimation speed and decrease estimation error, the initial parameters of RBFNN, such as weight, width, and center, are obtained after training it offline. These initial parameters are obtained from the gradient decent methods given in AppendixA.2, and the objective function given in AppendixA.1 is selected to drive these parameters based on the gradient decent method. Initial weights of RBFNN are given in AppendixA.3 for evidence, and other parameters are used in MATLAB.

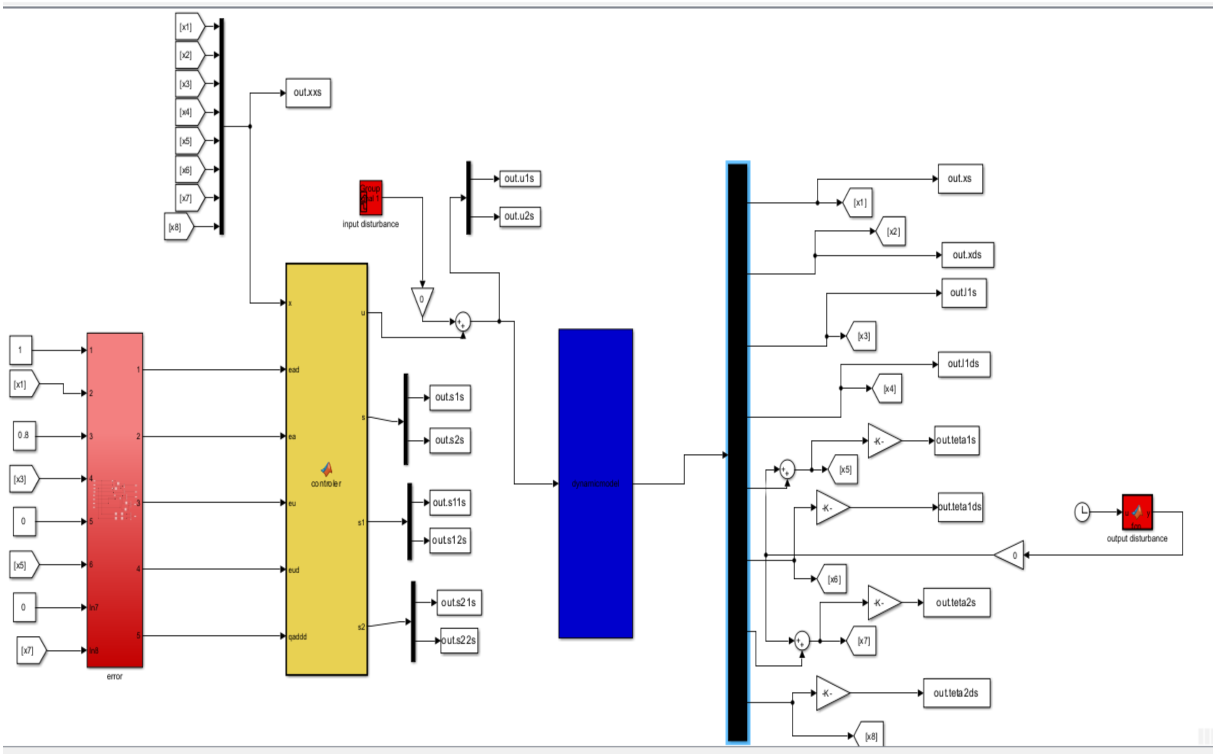


Figure 5.1: HSMC simulation block diagram

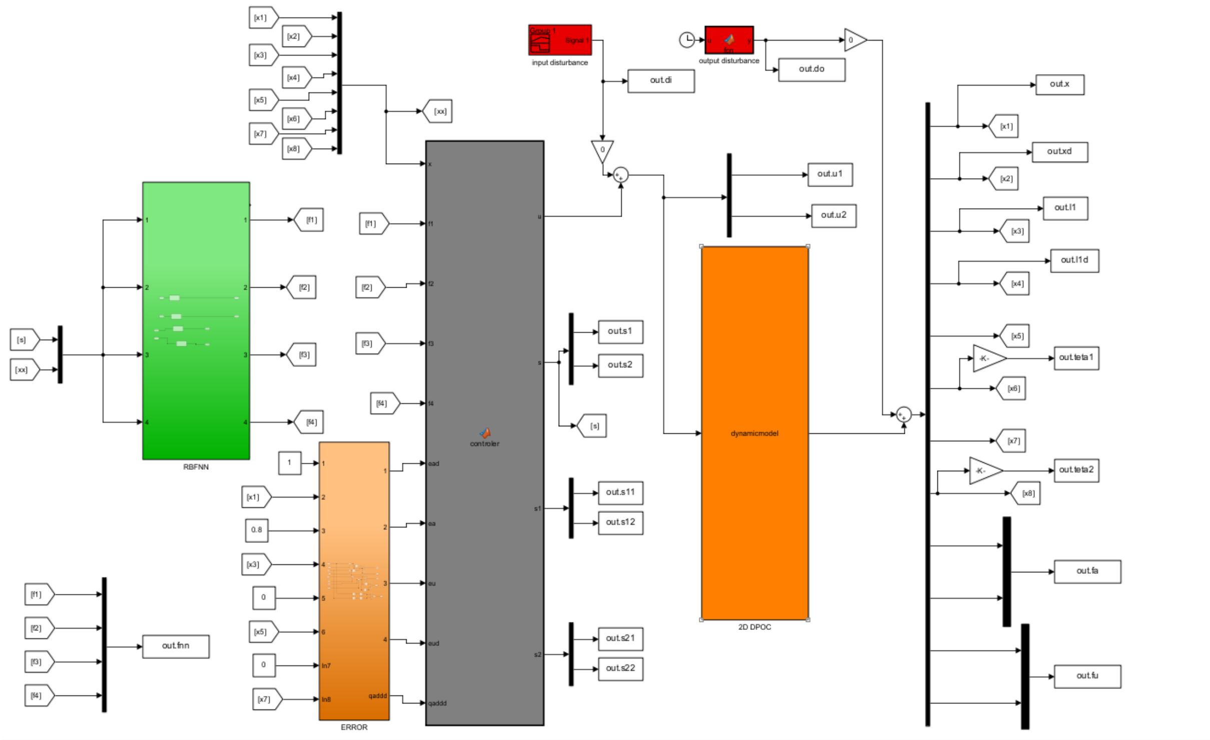


Figure 5.2: ARBFNN-HSMC simulation block diagram

Table 5.1: Initial and desired state vectors

Initial state vector $x_0 = [\ x_0 \quad \theta_{10}\ \theta_{20} \quad l_{10}]$	$[\ 0m \quad 0\ rad \quad 0rad \quad 0.4m]$
Desired state vector $x_d = [\ x_d \quad \theta_{1d} \quad \theta_{2d} \quad l_{1d}]$	$[\ 1m \quad 0\ rad \quad 0\ rad \quad 0.8m]$

The controller parameters are obtained after try and error and given in table5.2.

Table 5.2: Controller parameters

HSMC parameters	**ARBFNN-HSMC parameters**
The two first-layer sliding surfaces parameters $\boldsymbol{\alpha_1}$ = [0.3 0;0 5] $\boldsymbol{\alpha_2}$ = [6 0;0 16]	The two first-layer sliding surfaces parameters The same as HSMC
The second-layer sliding surface parameters $\lambda_{\mathbf{1}}$ =[5 0;0 20]; $\lambda_{\mathbf{2}}$ =[1.5 0;0 0.06]	The second-layer sliding surface parameters The same as HSMC

The coefficients of the switching control law
k2=[0.0001 0;0 0.01];
k1=[2.5 0;0 30];
	The coefficients of the switching control law
	
	The same as HSMC

-	Learning rate η =0.2
-	$\gamma_{\mathbf{1}} = \gamma_{\mathbf{2}} = \boldsymbol{diag}\left(\gamma_{\mathbf{1\ 1}}, \gamma_{\mathbf{12}}, \ldots, \gamma_{\mathbf{1\ 100}}\right)$ $\gamma_{\mathbf{1\ 1}} = \gamma_{\mathbf{12}}, \ldots = \gamma_{\mathbf{1\ 100}} = \mathbf{0.3}$

5.3 Closed loop simulation results

In this section, different conditions are considered, and the performance of both HSMC and ARBFNN-based HSMC is discussed.

Simulation results under normal conditions:- This section simulates both HSMC and ARBFNN-HSMC using the crane model parameters listed in table 3.2, the starting and intended state variables listed in table 5.1, and the controller parameters listed in table 5.2. The required references provide a thorough tracking of the outcomes produced by the HSMC and AHSMC methodologies. Figures below display the crane performances for both controllers: Figure5.3 plots the cargo angle, angular velocity, position of the trolley, and trolley velocity. This figure shows that both controllers precisely move the trolley to the required location, control the length of the rope to the desired length, and effectively suppress the swing angles of the hook and payload.

For ARBFNN-HSMC, it takes approximately 12 seconds for the trolley to reach the 1-meter mark, whereas for HSMC, it takes about 13 seconds. The payload may be lowered from 0.4m to 0.8m in about 0.74s. Figure5.3 b and d illustrate the limited range of control over the trolley's speed and cable length. Table5.3 provides a full comparison of the two controllers with respect to rising time t_r and settling time t_{sett} for trolley position and cable. The table also includes information on the maximum oscillation angles of the hook and payload during crane transportation, as well as their residual angles following transit. Even if portions of the plant dynamics are adaptively computed online, it is evident from the table that the performance of the suggested ARBFNN-HSMC scheme is nearly identical to that of the HSMC legislation.

Table 5.3: Performance analysis for both controllers

Controller	**Trolley displacement(m)**			**Cable length(m)**			θ_{1max} **(deg)**	θ_{2max} **(deg)**	θ_{1res} **(deg)**	θ_{2res} **(deg)**
	xd	t_{sett} **(sec)**	t_r **(sec)**	**l1d**	t_{sett} **(sec)**	t_r **(sec)**				
HSMC	**1**	**13**	**7.1**	**0.8**	**0.74**	**0.32**	**1.5**	**2.578**	**2.7265e-05**	**-0.00012**
ARBFNN-HSMC	**1**	**12**	**6.2**	**0.8**	**0.74**	**0.32**	**1.2**	**2.01**	**3.1504e-05**	**3.1568e-05**

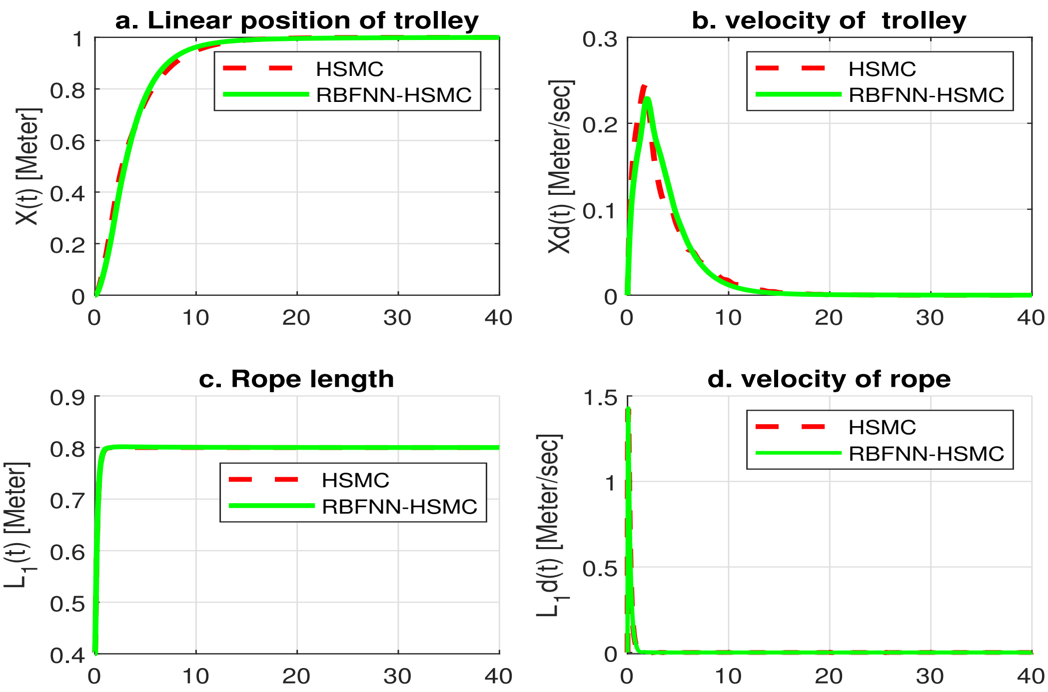


Figure 5.3: Positions and speeds of trolley and cable

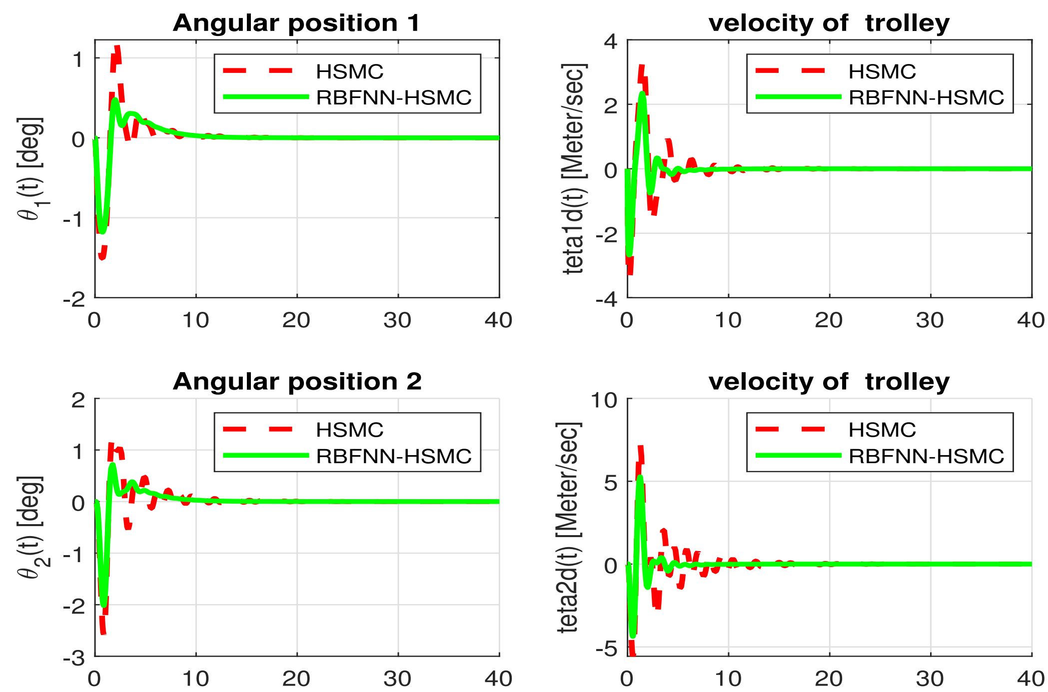


Figure 5.4: Swing angles and angular velocities of hook and payload

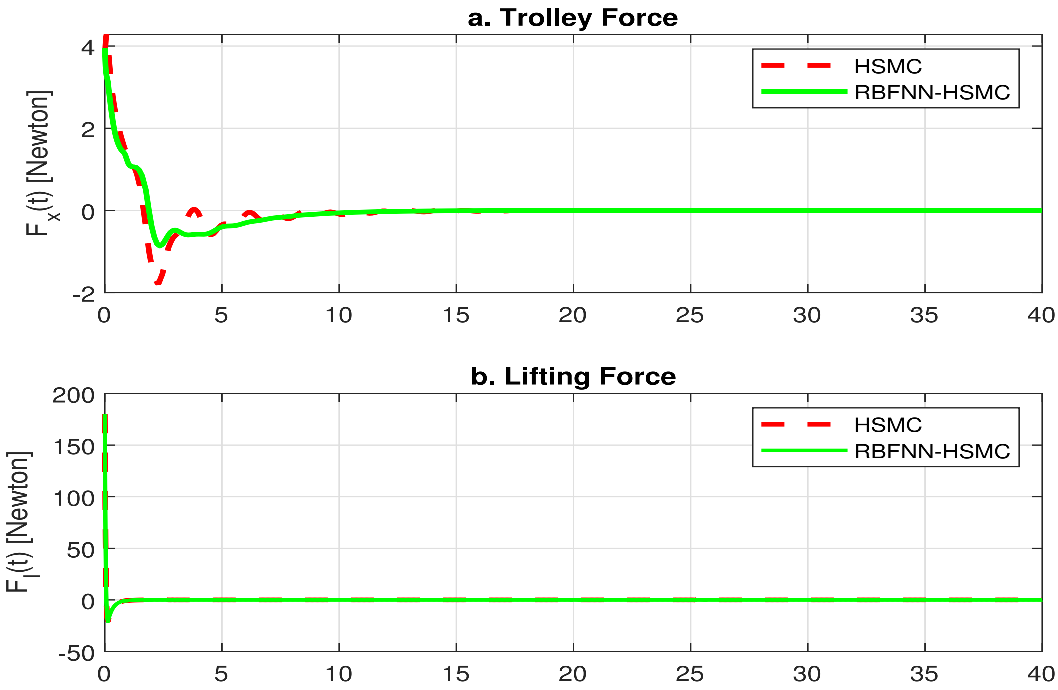


Figure 5.5: Control inputs

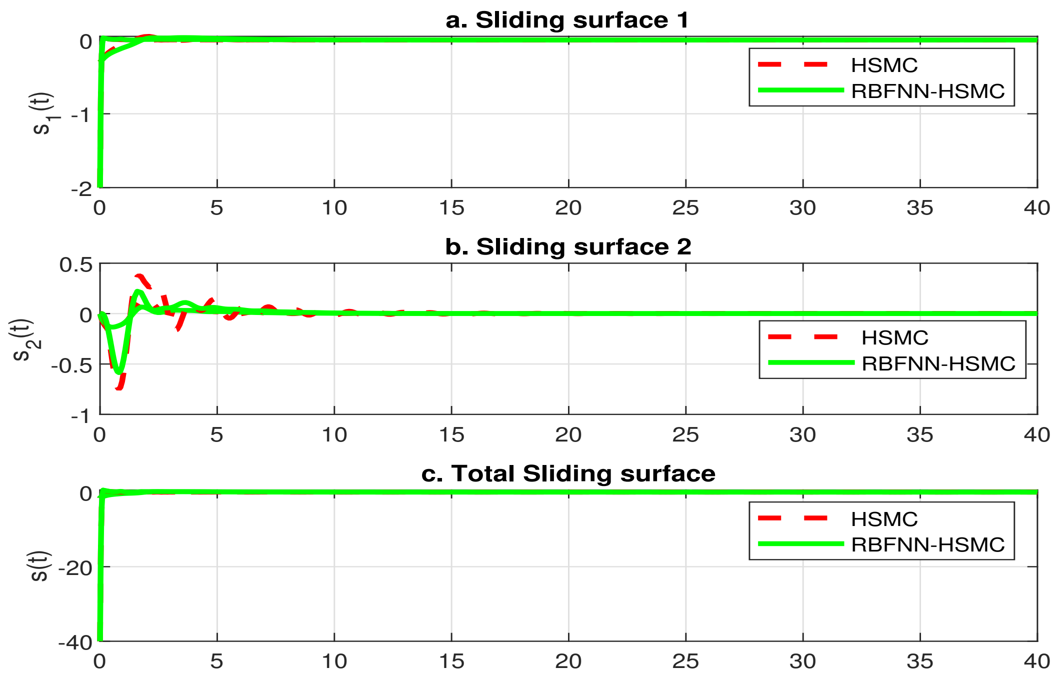


Figure 5.6: Sliding surfaces

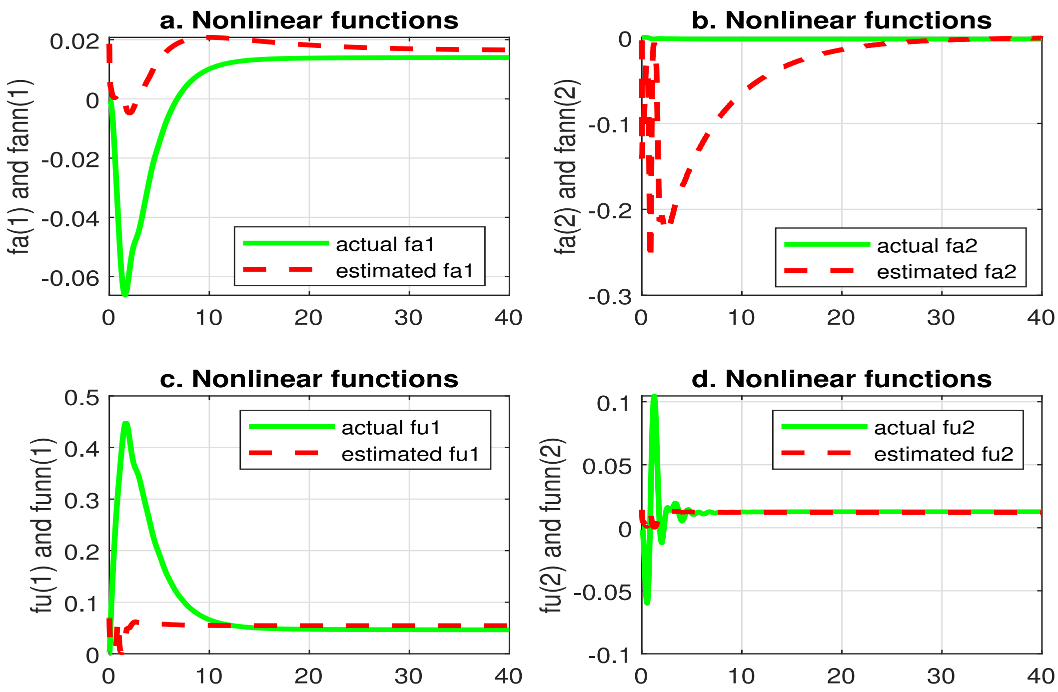


Figure 5.7: Actual and estimated functions

From figure 5.5, the control output of ARBFNN-based HSMC is smoother without chatter than that of HSMC alone. The zoomed view of figure 5.5 is shown in figure 5.8 to show the difference between the two control outputs from the controllers. HSMC control output has some chattering, while ARBFNN-based HSMC eliminates chattering effects on control force. The control input has a chattering phenomenon. The HSMC, as a branch of SMC, inherits the inner drawbacks of SMC as well. This shows one of the advantages of using a neural network in SMC. Sliding surfaces and the RBFNN estimated nonlinear dynamic of the plant are also illustrated in figures 5.6 and 5.7, respectively. As proven in 4.11, the hierarchical sliding surfaces are stable. The sliding mode of the 2nd-order sliding surface is reachable in less than 2 seconds. The two first-order sliding surfaces are also stable. The output of RBFNN and the actual nonlinear function are almost the same. The estimation error can be reduced by increasing the number of neurons in RBFNN, as given in the next section.

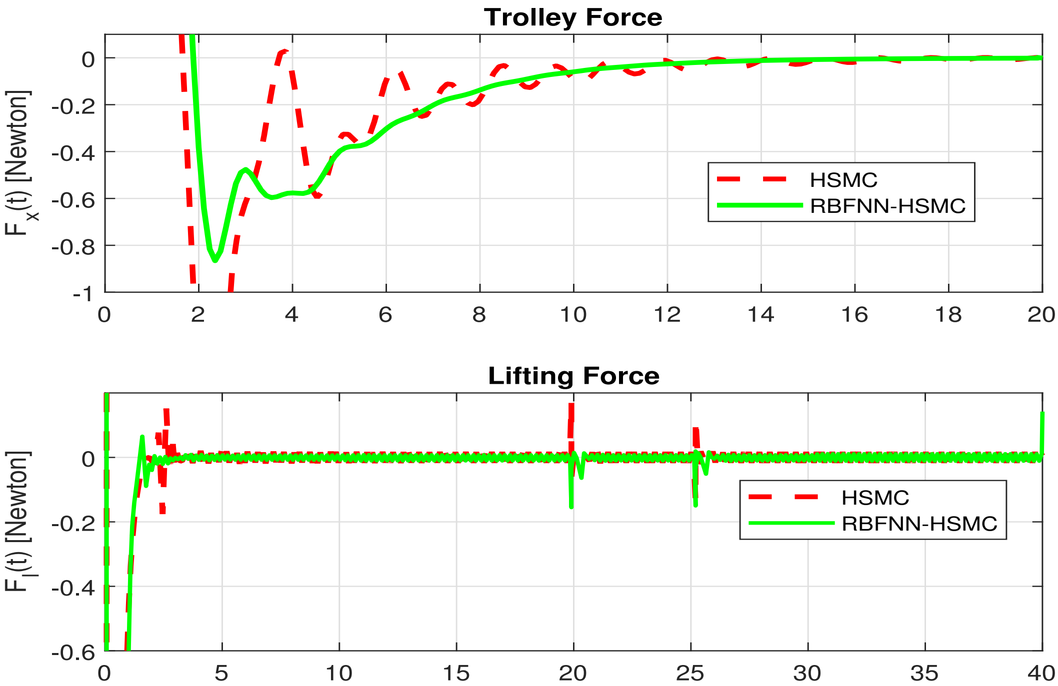


Figure 5.8: zoomed view of Control inputs

On the other hand, the coefficient of switching control law also has an effect on increasing chattering on the control output of the controller. For example, the value of the coefficient of the switching control law from table 5.2 is k2=[0.0001 0;0 0.01]. Increasing this value will decrease the smoothness of the control forces. For k2=[0.1 0; 0 0.1], the system outputs and control forces are presented in figures 5.9, 5.10, and 5.11.

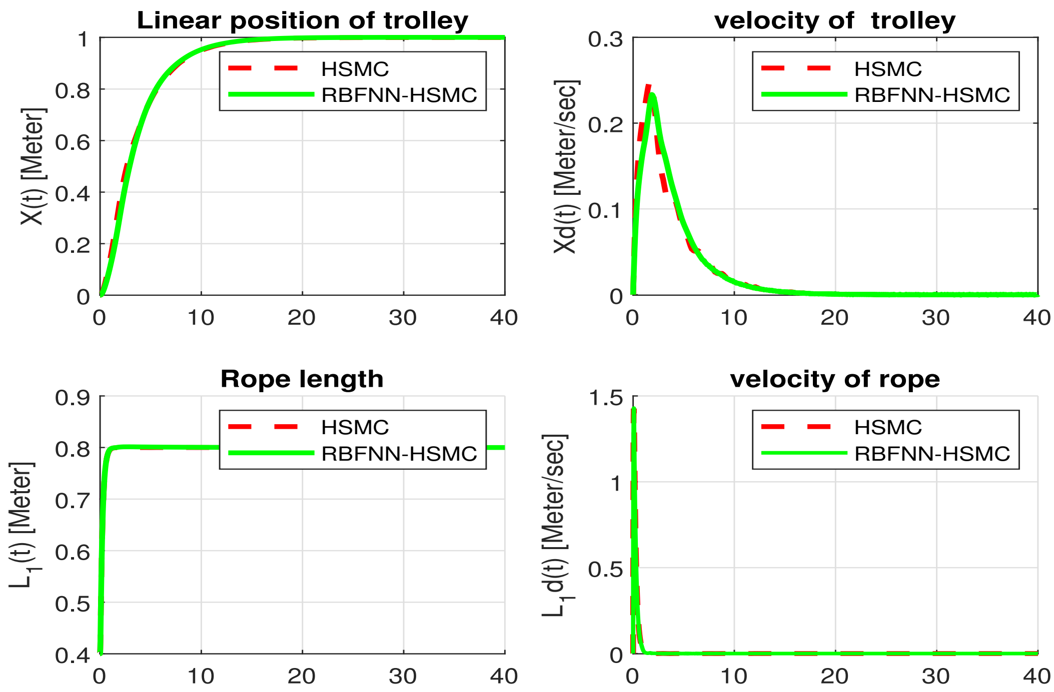


Figure 5.9: Positions and speeds of trolley and cable

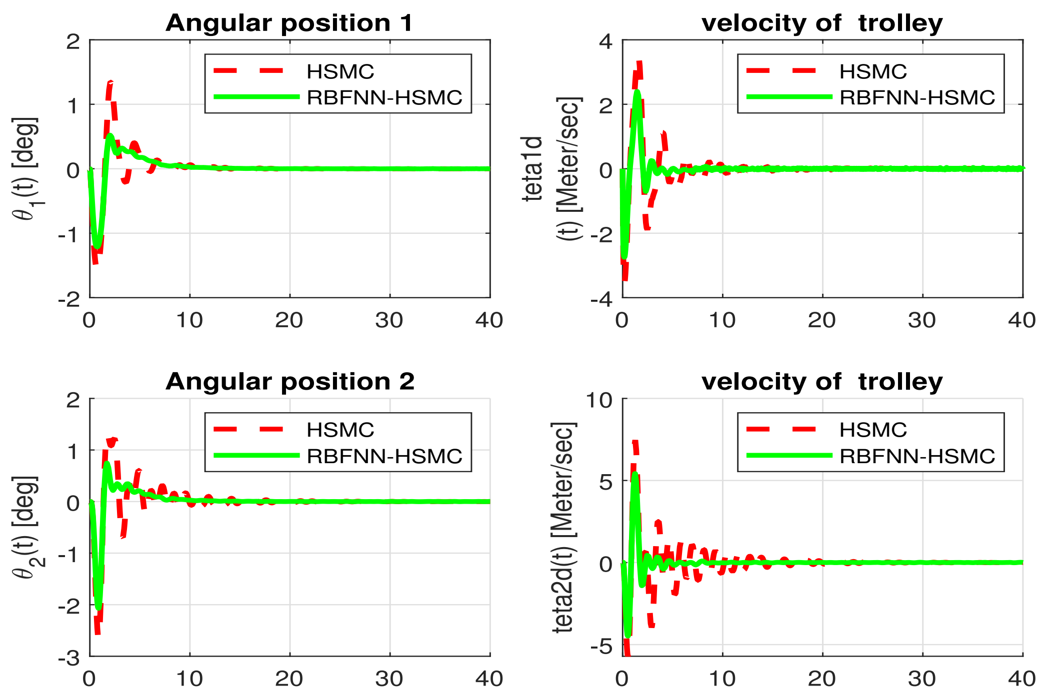


Figure 5.10: Swing angles and angular velocities of hook and payload

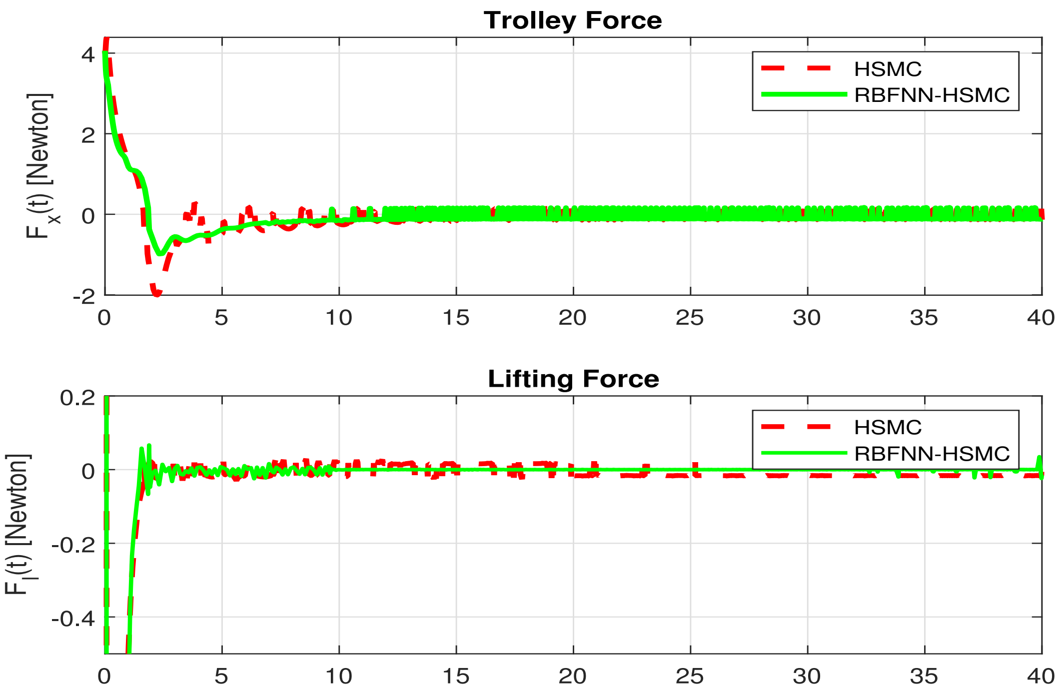


Figure 5.11: Control inputs

Effect of number of neuron in RBFNN

For online nonlinear function estimation, a basic RBFNN structure with 100 hidden neurons is employed in this thesis. The goal of utilizing ARBFNN is to utilize it with HSMC so that the controller tracks the reference signal, rather than modeling the system in its entirety. The designed Lyapunov stability condition-based updating law provided in 4.23 and 4.24 is used to update the weight of RBFNN in the event that the intended reference signal is not monitored. Error identification generally decreases with an increase in the number of hidden neurons. The separability inside the hidden space rises with the number of hidden neurons. However, a trade-off must be made because, in real-time applications, overhead computations rise in tandem with the number of hidden neurons, which requires careful consideration. To show this, RBFNN took 200 hidden neurons and compared them with 100 hidden neurons to estimate the nonlinear function in the system. As it is observed from figures 5.12 and 5.13, when the number of hidden neurons in RBFNN is increased from 100 to

200, the estimation error between the actual nonlinear function and the output of RBFNN is decreased.

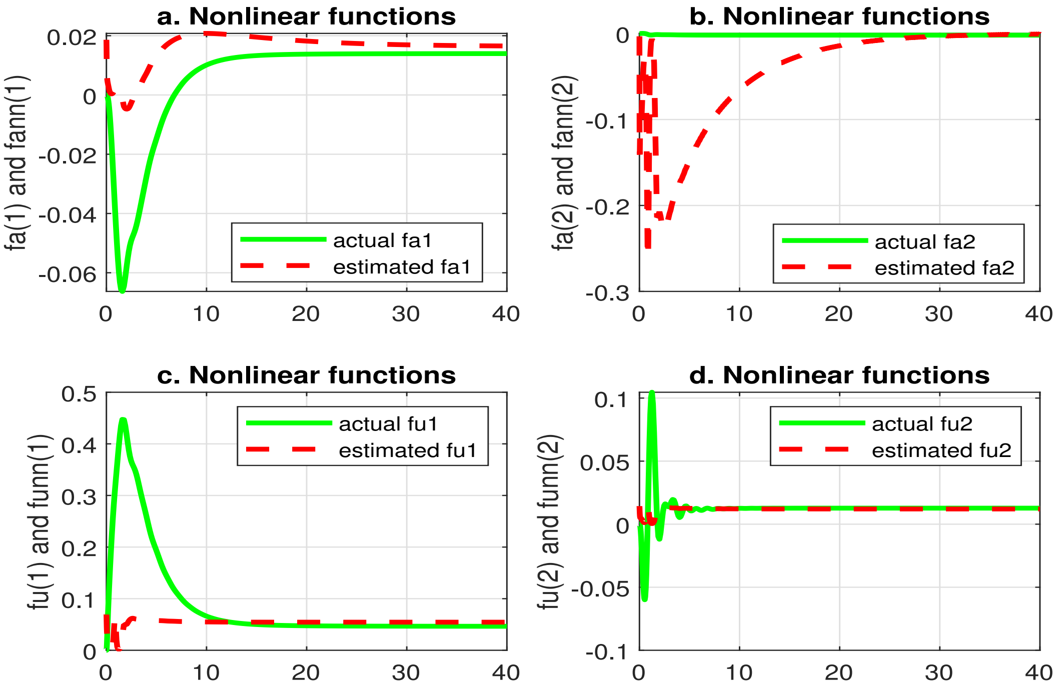


Figure 5.12: Actual and estimated functions with 100 hidden neurons

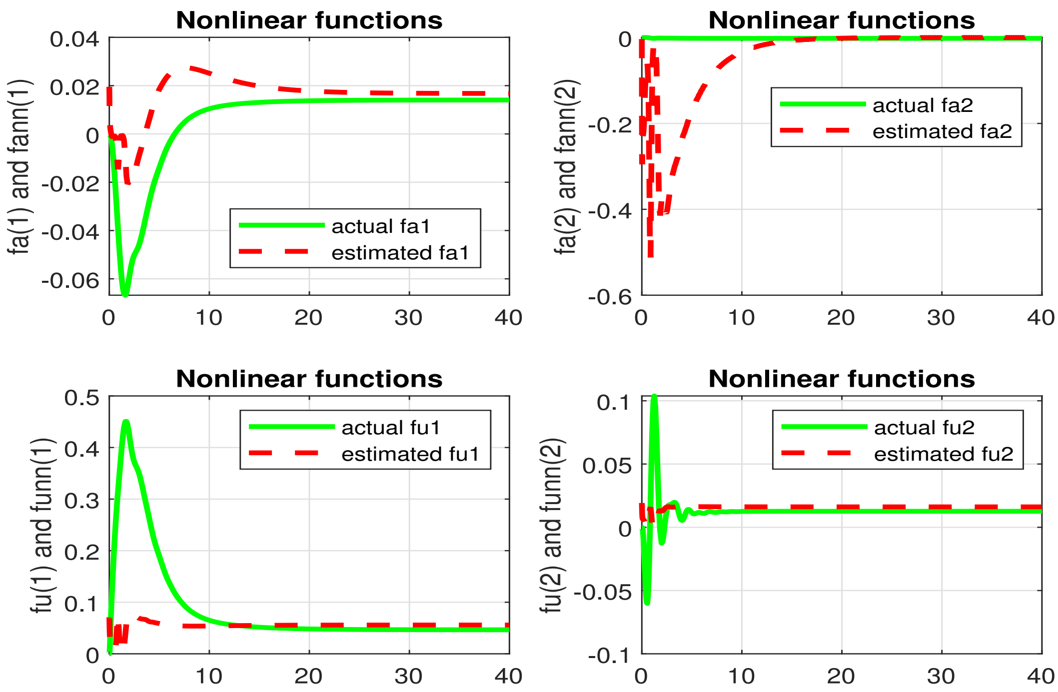


Figure 5.13: Actual and estimated functions with 200 hidden neurons

Simulation results under large parameters variation:- In this part,to show the effectiveness and robustness of ARBFNN based HSMC,the model parameters given in table 3.2 are changed as given in table 5.4.This around 500% change in parameters and simulation results are given in figure5.14 to 5.17.

Table 5.4: Model parameter values for 2nd simulation

Symbol	Parameter	Value
m	Trolley mass	55.2 kg
m_1	Hook mass	12 kg
m_2	Payload mass	6 kg
l_2	Load/mass pendulum length	3.6m
g	Gravity acceleration	9.8m/s

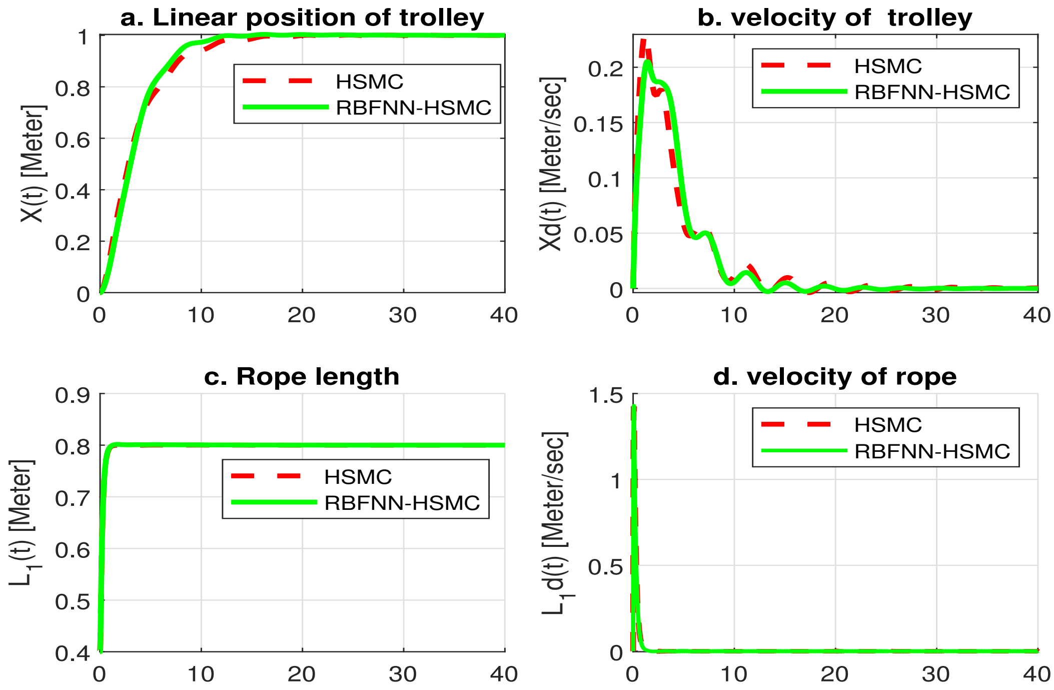


Figure 5.14: Positions and speeds of trolley and cable

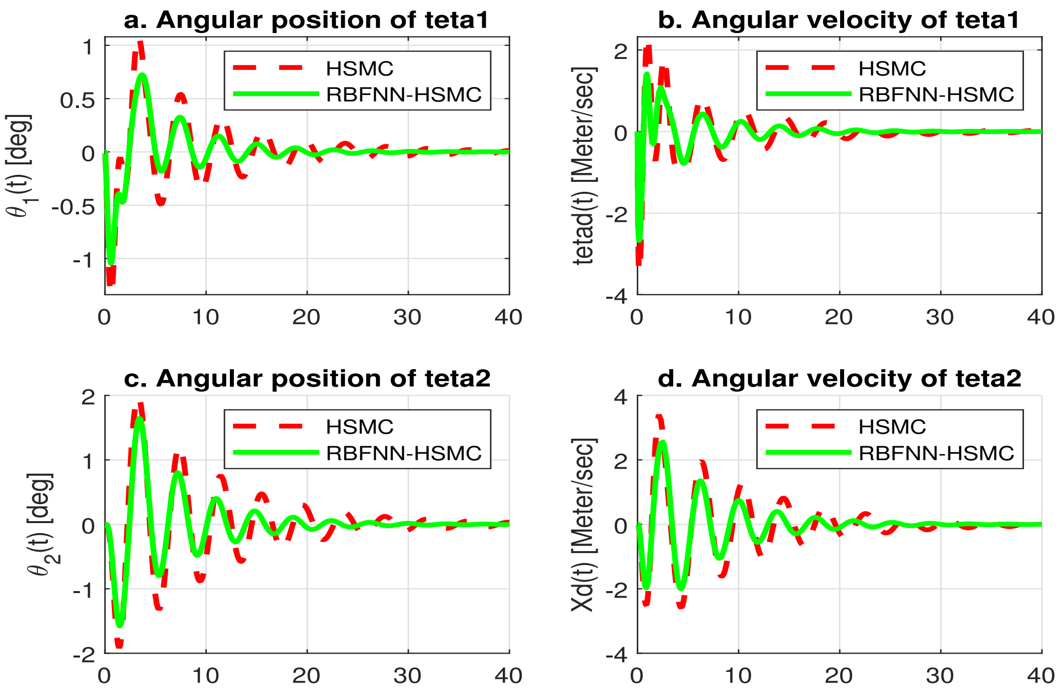


Figure 5.15: Angular positions and velocities of hook and payload

Figures 5.14 and 5.15 show the positions and swing angles of the trolley and cable length and hook and payload, respectively. From these figures, specifically for swing angles of the hook and the trolley, ARBFNN-based HSMC has better performance than HSMC alone because swing angles decay faster for ARBFNN-HSMC. This is due to the adaptation mechanism of RBFNN for changes in parameters. The control forces and sliding surfaces for this case are also shown below. The maximum trolley driving force is around 26 N, and the maximum payload hoisting force is 1000 N. The increase in maximum forces is because of large parameters. The following table shows comparisons between two controllers in this case.

Table 5.5: Performance analysis for both controllers

Controller	**Trolley displacement(m)**			**Cable length(m)**			θ_{1max} **(deg)**	θ_{2max} **(deg)**	θ_{1res} **(deg)**	θ_{2res} **(deg)**
	xd	t_{sett} **(sec)**	t_r **(sec)**	**l1d**	t_{sett} **(sec)**	t_r **(sec)**				
HSMC	**1**	**12.3**	**6.95**	**0.8**	**0.74**	**0.328**	**1.34**	**1.9**	**0.0137**	**-0.0319**
ARBFNN-HSMC	**1**	**10.7**	**6.33**	**0.8**	**0.74**	**0.285**	**1.04**	**1.64**	**-0.0011**	**-0.0022**

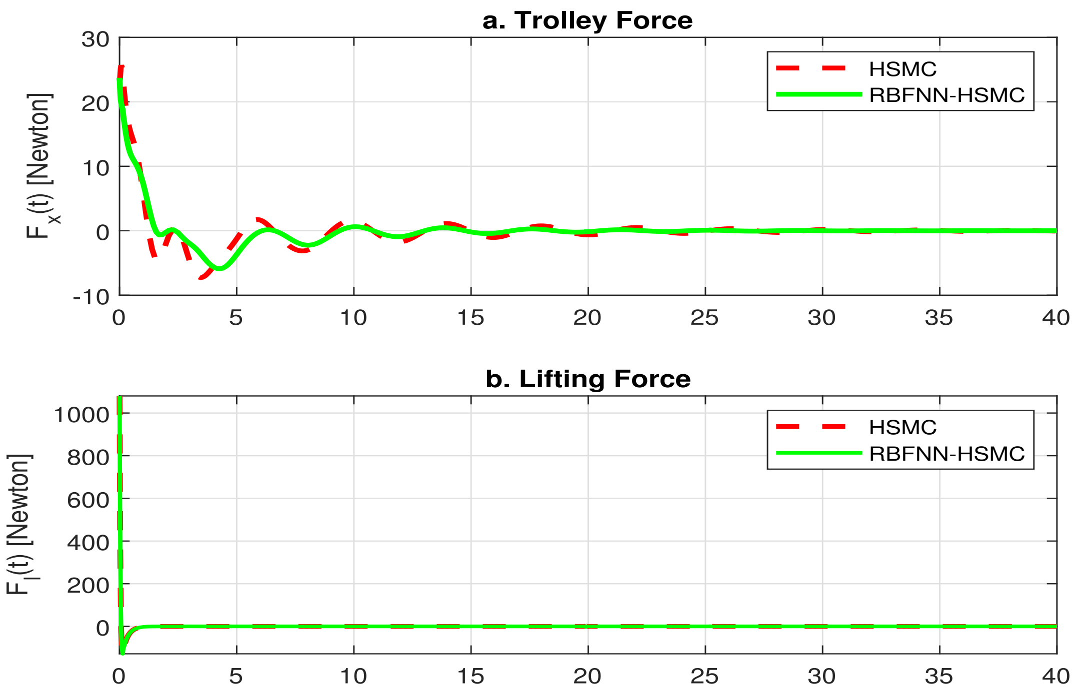


Figure 5.16: Control inputs

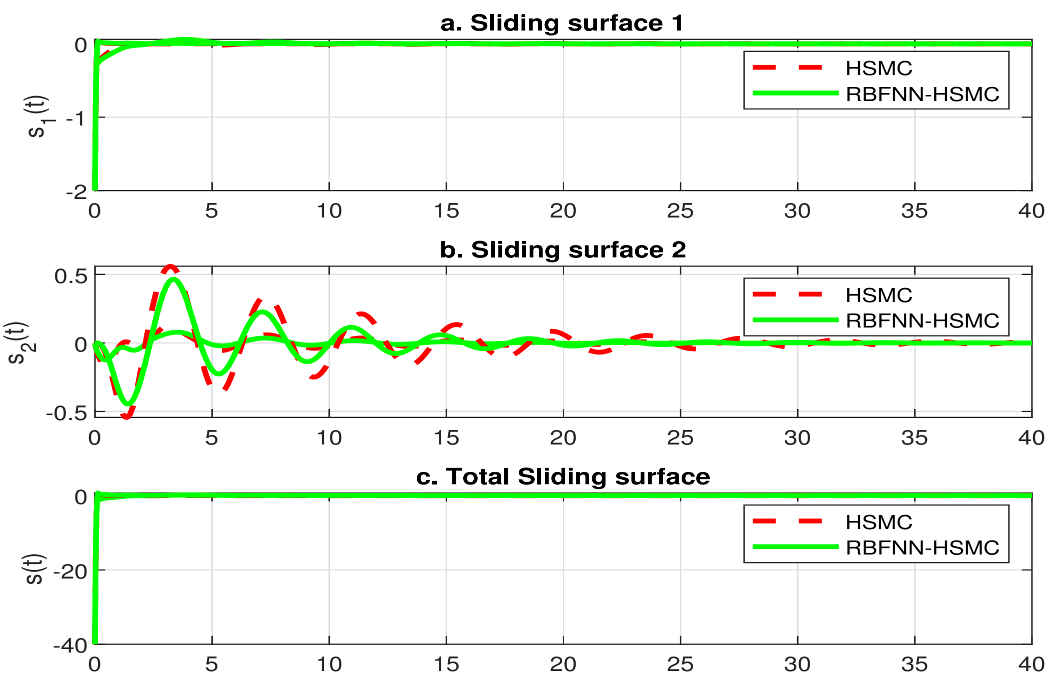


Figure 5.17: Sliding surfaces

Simulation results for different desired references:- In this case, the same controller parameters given in Table 5.2 are used for the simulation. However, the initial state variables of the trolley and cable length are changed from 0m to 2m and 0.4m to 1 m, and the desired state variables are changed from 1m to 0m and from 0.8m to 0.6 m, respectively. The results obtained are shown in the following figures: From this figure, it is observed that, with the same controller parameters, the performance of both controllers is not changed even if the target positions of the trolley and cable length are changed. This shows the robustness of both HSMC and ARBFNN-based HSMC.

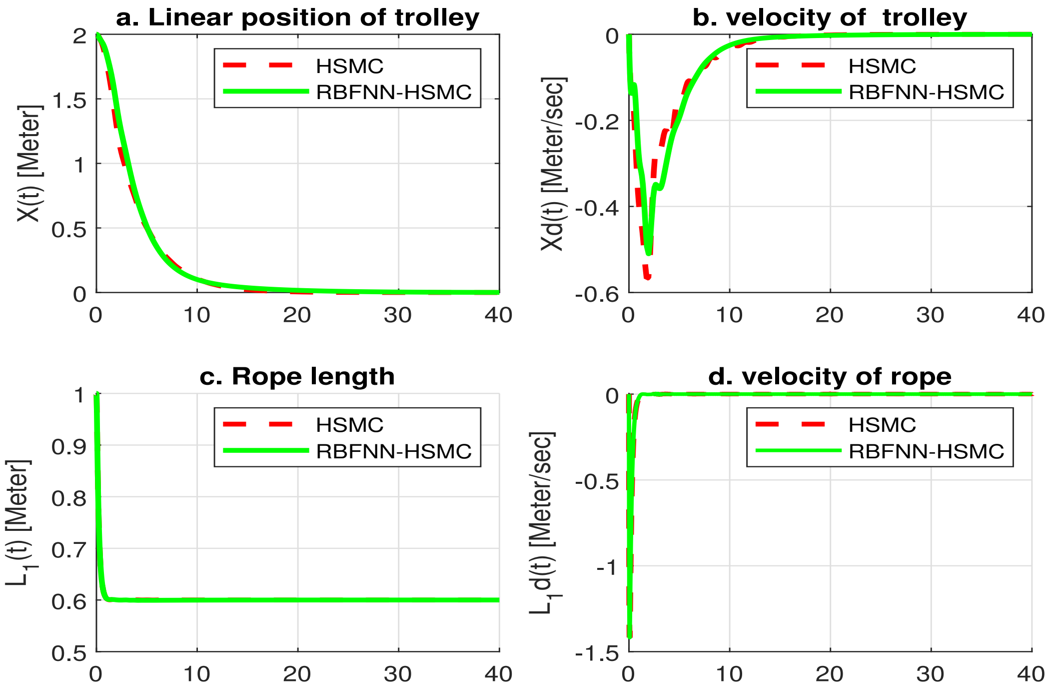


Figure 5.18: Positions and velocities of trolley and cable

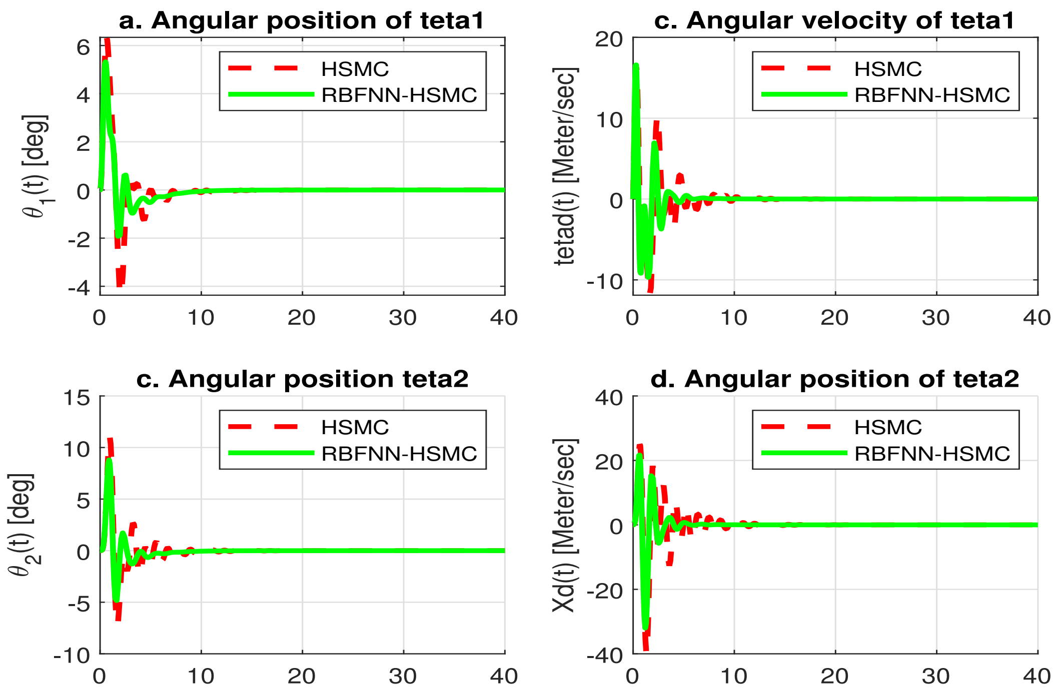


Figure 5.19: Angular positions and velocities of hook and payload

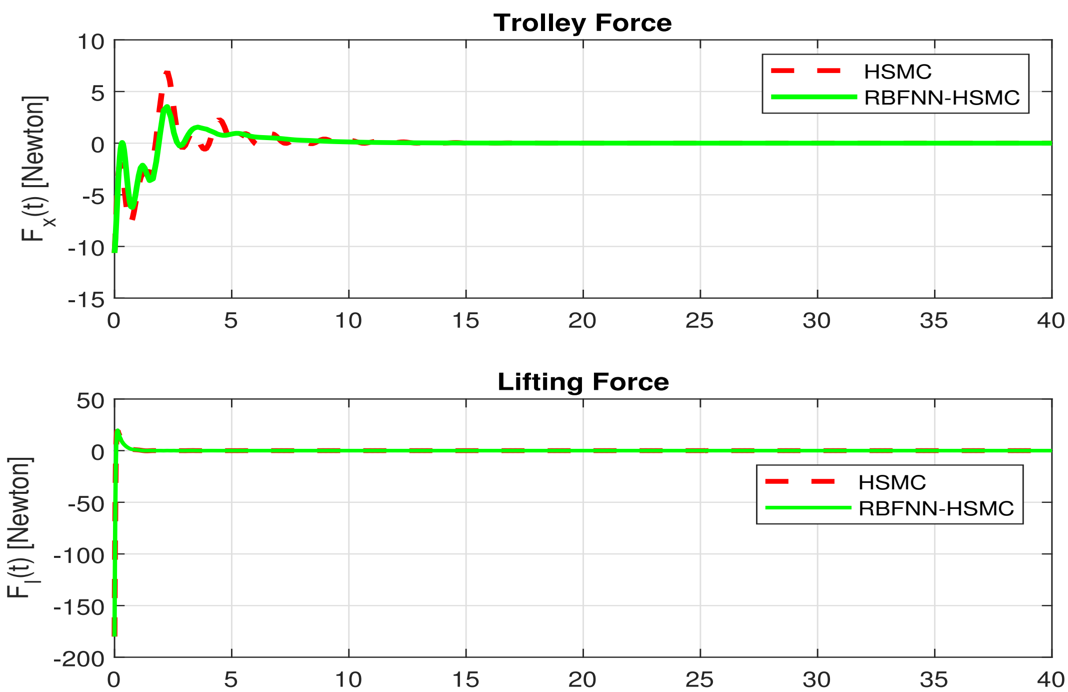


Figure 5.20: Control inputs

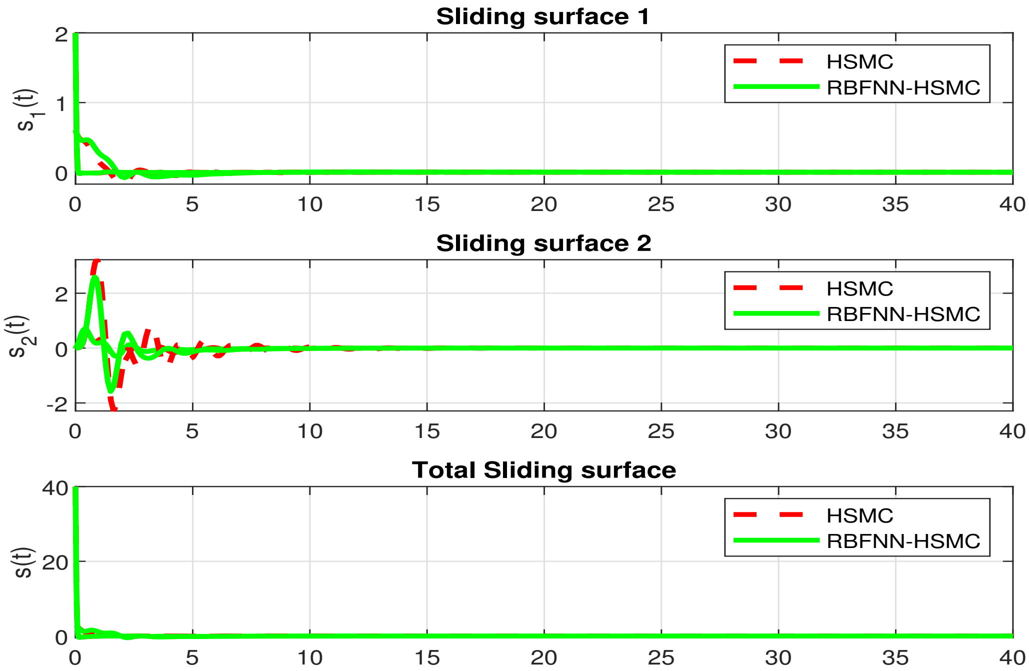


Figure 5.21: Sliding surfaces

From figure 5.18, when the initial condition and desired references are changed, the controller is robust to this change in the trolley position and payload lifting. But the maximum swing angles of the hook and the trolley increase to 6 degrees and 10 degrees, respectively. If an appropriate smooth trajectory is designed to drive the trolley to the target position smoothly and to lift or lower the payload to the desired height instead of a constant trajectory, the swing angles of the hook and payload can be reduced.

Simulation result with initial swing disturbances on the hook and the payload:- In this instance, an initial swing disturbance of 20 degrees is applied to the cargo and hook, as depicted in figures 5.22 through 5.25. It is evident that the payload's swing and the swing of the huge hook were both produced by the first swing disturbance. Nonetheless, the suggested control approach still meets the precise positioning goals for the length of the rope and the trolley position. The swing of the huge hook and the cargo are simultaneously and efficiently reduced.

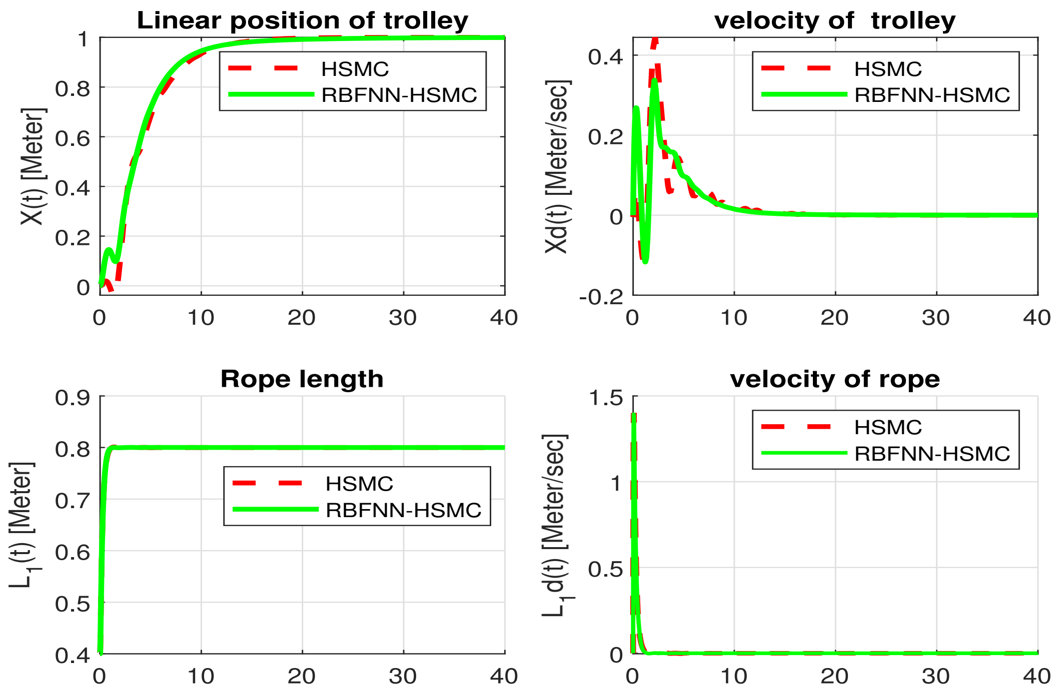


Figure 5.22: Positions and velocities of trolley and cable

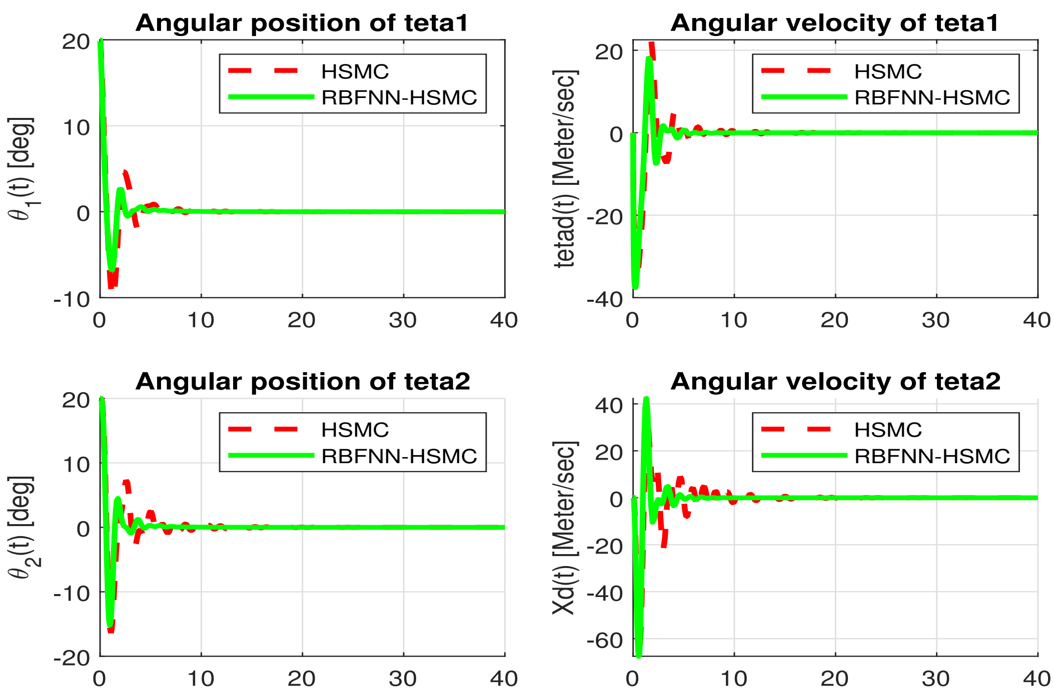


Figure 5.23: Angular positions and velocities of hook and payload

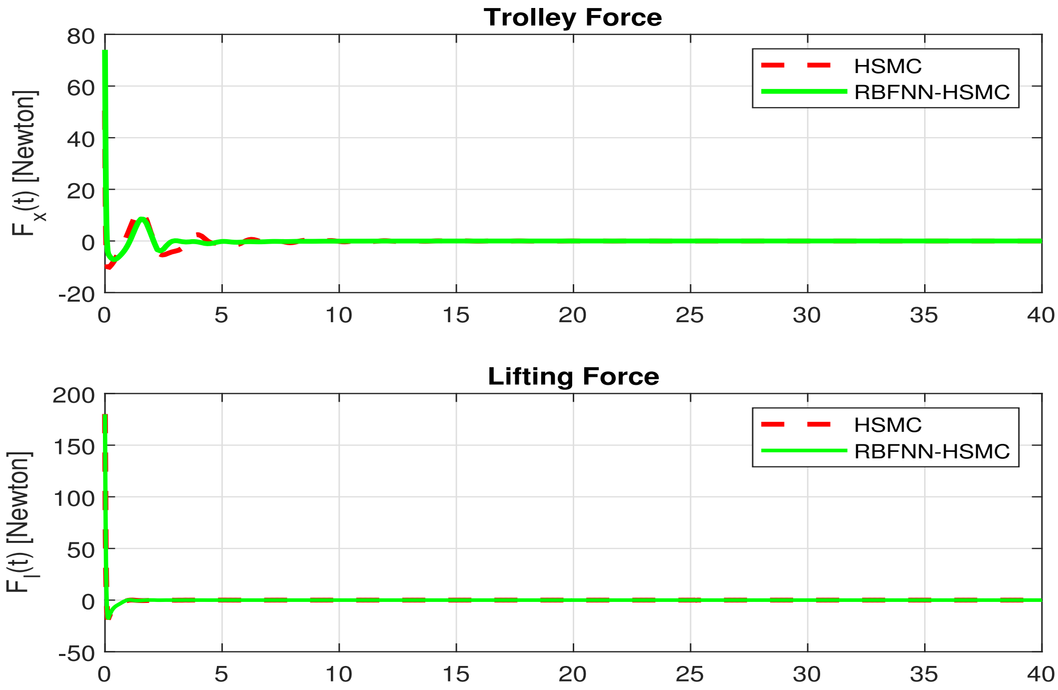


Figure 5.24: Control inputs

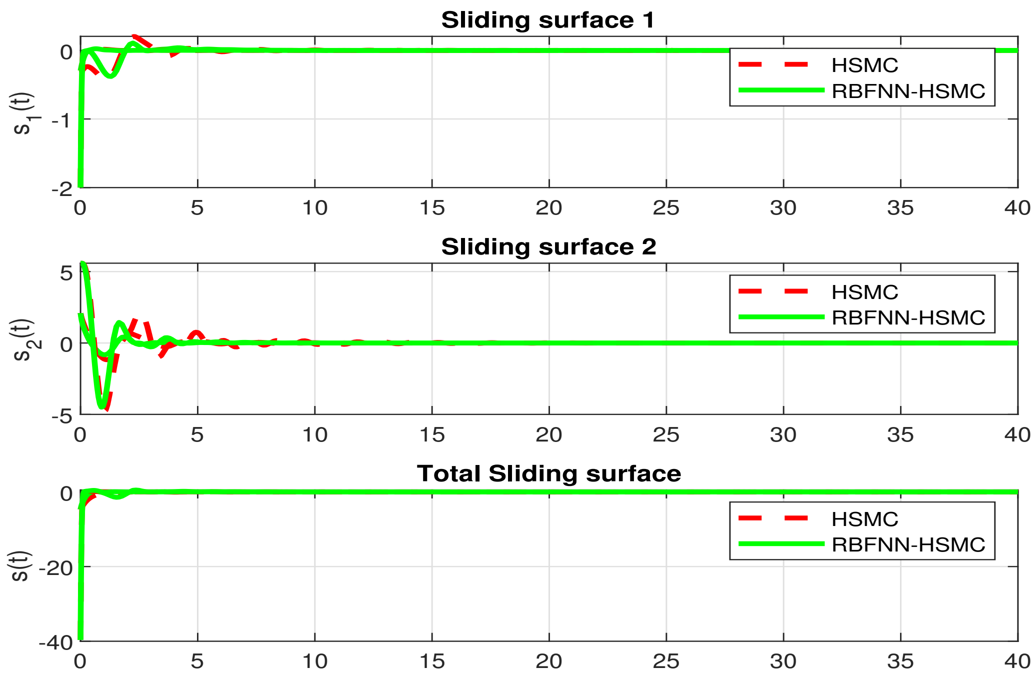


Figure 5.25: Sliding surfaces

Simulation results with input and output external disturbance:- Finally, to show the effectiveness of the proposed control, the input disturbance shown in figure 5.30 b and the output disturbance shown in figure 5.30 a, respectively, are added between 20 sec and 25 sec and between 30 sec and 35 sec, respectively. The output disturbance is added to both the hook and payload. In this simulation case, the same controller parameters and the same initial and desired state vectors given in tables 5.2 and 3.2 are used, and the results are shown in the following figures. Again, it is clear that the proposed control strategy is responsive to both disturbances. The controller rejects the disturbances, and both the trolley and cable reach their desired references. The oscillations of the hook and payload are also suppressed.

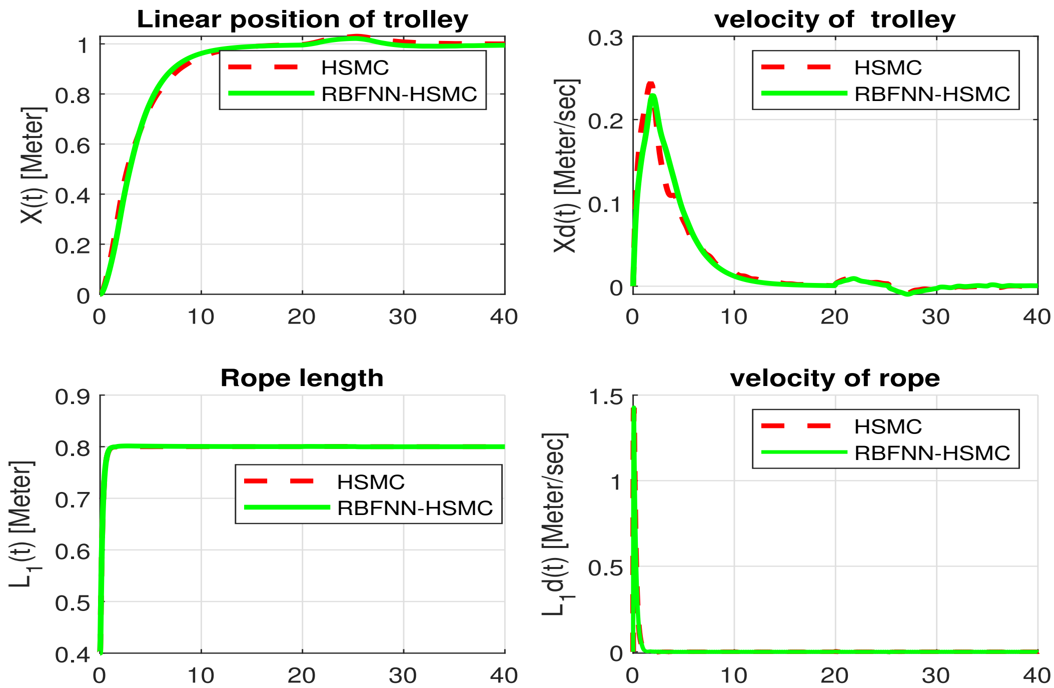


Figure 5.26: Positions and velocities of payload and cable

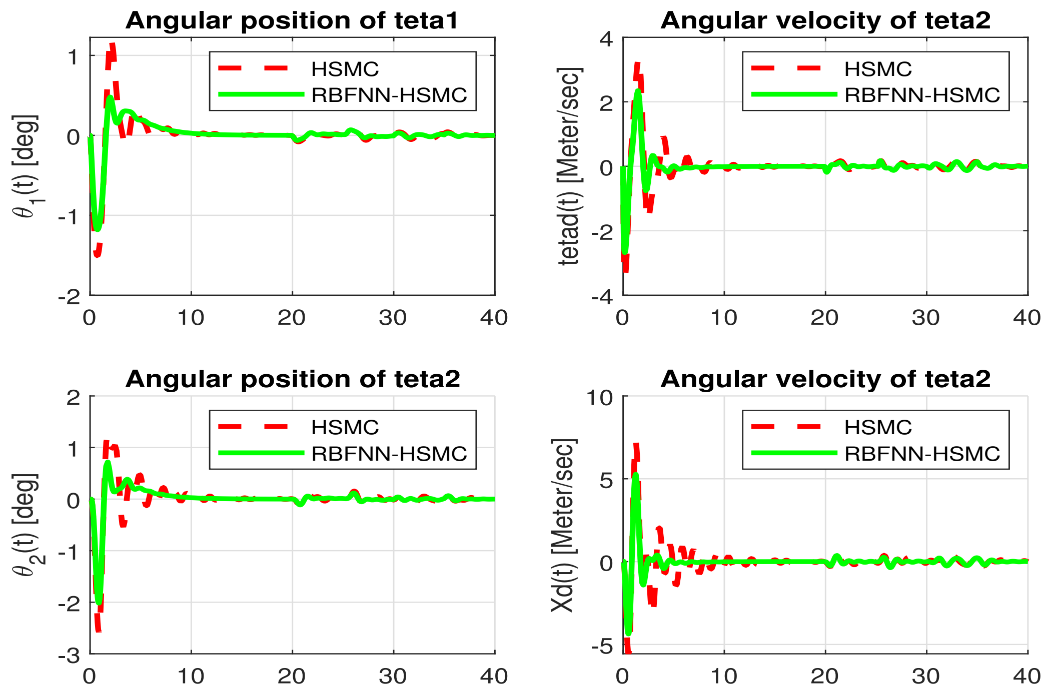


Figure 5.27: Angular positions and velocities of hook and payload

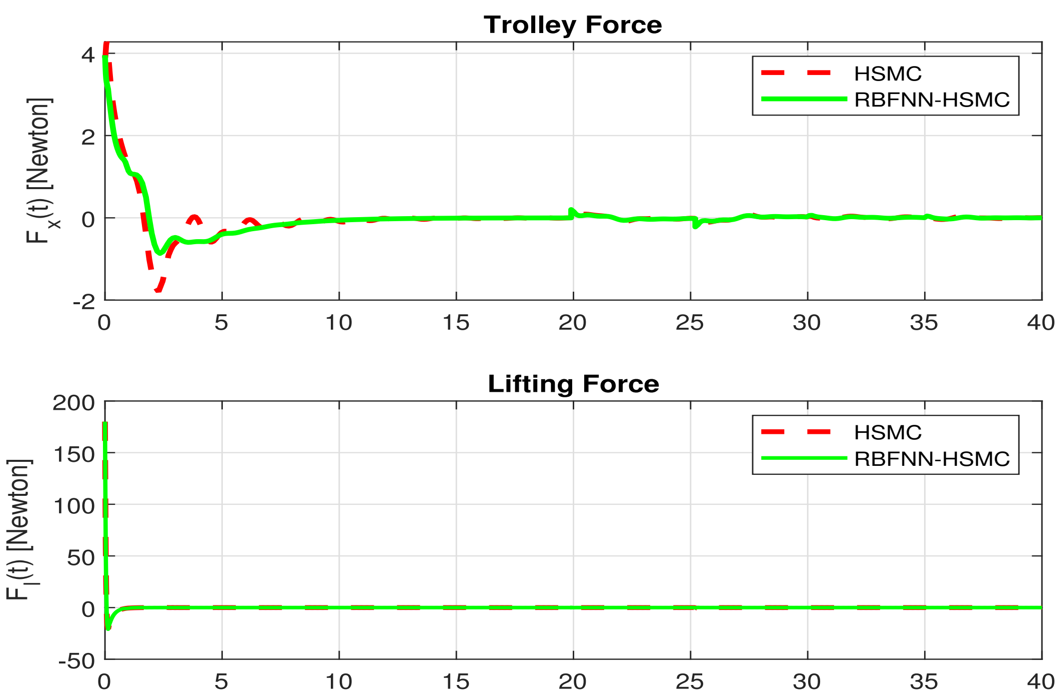


Figure 5.28: Control inputs

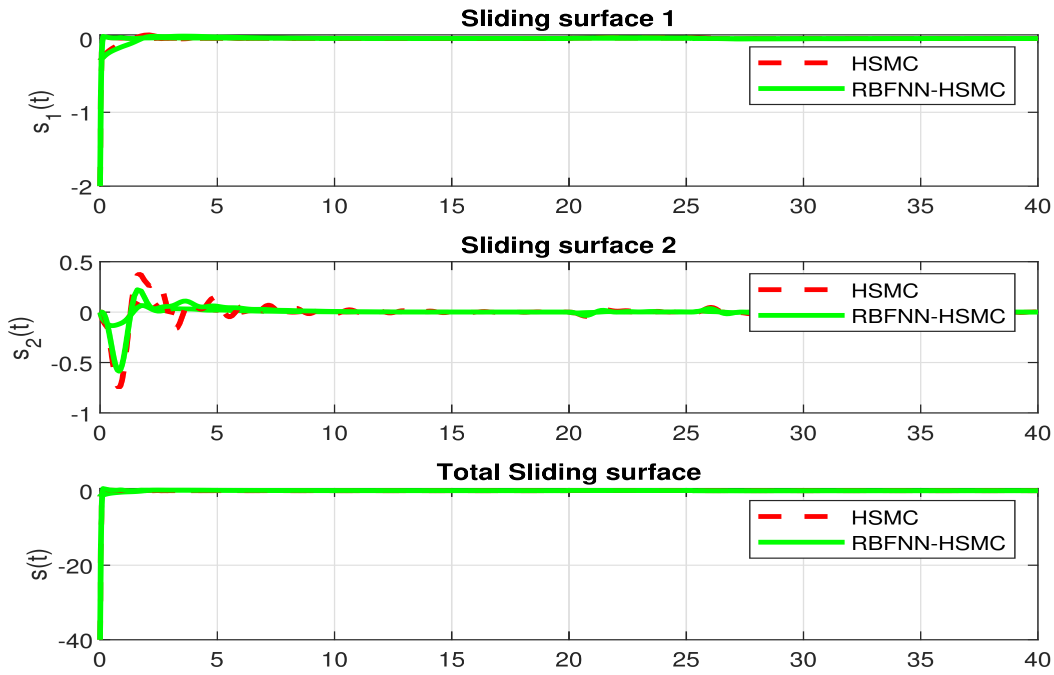


Figure 5.29: Sliding surfaces

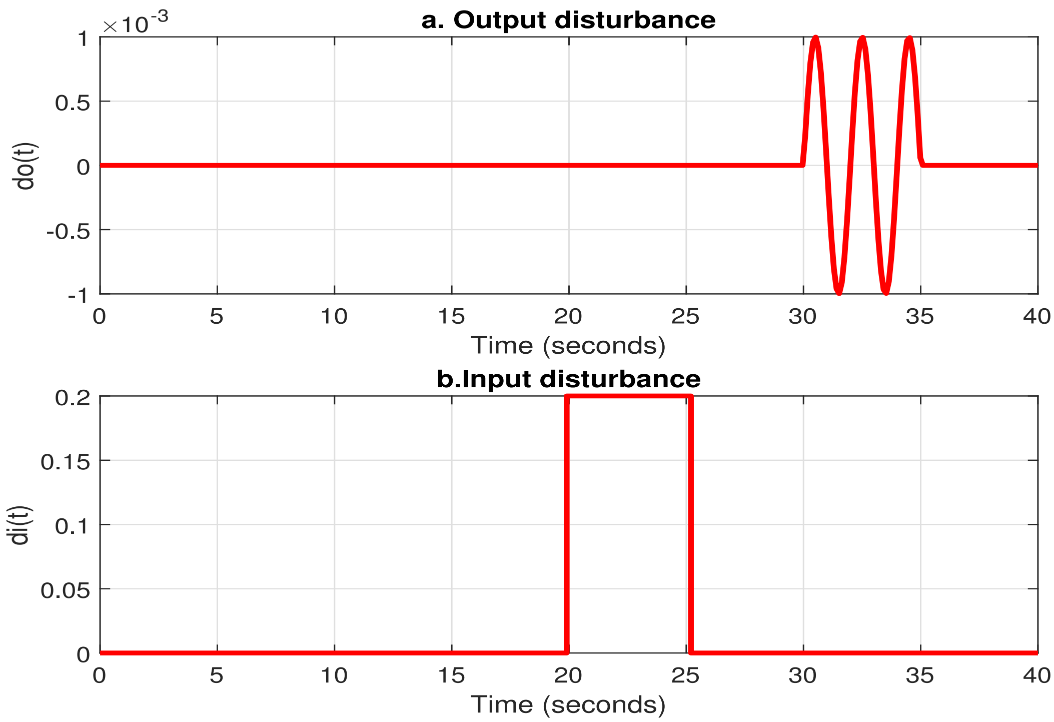


Figure 5.30: External disturbances

According to the aforementioned ARBFNN-based HSMC simulation findings, the system produces smooth, chatter-free outputs. This is due to the benefit of combining a neural network and a sliding-mode controller. SMC is used directly to track the reference signals in models that are known mathematically. Nonetheless, an adaptive neural network (RBF) approach is best suited when combined with SMC to reduce chattering phenomena and boost SMC's robustness in an environment where system dynamics are considerably changing, extremely nonlinear, and, in theory, not fully understood, such as overhead cranes. This thesis uses adaptive RBFNN to estimate the unknown nonlinear function of the 2D DPOC. Then, using the estimated nonlinear function as a guide, the HSMC law based on ARBFNN is built to create the control input.Consequently, it is clear from the aforementioned simulation results that ARBFNN-based HSMC can be used as HSMC with the benefit of a model-free-based controller in which all or a portion of the plant dynamics can be estimated online, which can shorten computation times for complex, uncertain, and nonlinear functions.

Chapter 6

Conclusion and Future Works

6.1 Conclusion

Automatic and closed-loop control of overhead cranes has significant advantages in increasing modern industrial productivity and addressing safety issues as well. In this thesis, a dynamic model of a 2D double pendulum overhead crane with variable length is derived based on the Euler-Lagrange equation of motion and validated using MATLAB/Simulation software. From the open-loop simulation of the crane dynamic model,there is around a 20-degree maximum swing angle in the payload, and both trolley and cable lengths are unstable. Since there is uncertainty in crane systems such as payload mass variation and external disturbance, there is a deference between the nominal mathematical model and the real physical crane system. For this reason, HSMC based on the ARBFNN nonlinear function estimation control scheme is proposed to address the control issues of a 2D double pendulum crane with variable rope lengths. ARBFNN is utilized to online estimate the uncertain dynamic part of 2D DPOC, the swings of the hook, and the payload. ARBFFNN-HSMC has an advantage over HSMC since it doesn't need an accurate mathematical model of the system. The designed controller is simulated using MATLAB/Simulink, and the addressed control method scheme achieves accurate regulation for the trolley position and the rope length, as well as the simultaneous suppression of hook and payload swing angles. The proposed control method finally achieves adequate performance and robustness with respect to large parameter variations,different target positions,initial swing disturbance, and external disturbance.

6.2 Future Work

An extension of this work can be done by designing a completely model-free HSMC by identifying all the dynamic models of the crane. This can make HSMC more robust in uncertain environments. In a crane system, measuring the angular velocity of the payload is difficult, and for cost reduction, the installation of a velocity sensor is not necessary. So, one can include the velocity estimation method in their control design as an extension of this work. As a final recommendation, a 3D overhead crane with a double pendulum and variable cable can also be considered.

Bibliography

[1] U. H. Shah and K.-S. Hong, *Dynamics and Control of Industrial Cranes*. Springer Singapore, 2019.

[2] M. Li, H. Chen, and Z. Li, "Input-limited optimal control for overhead cranes with payload hoisting/lowering and double pendulum effects," *Nonlinear Dynamics*, vol. 111, no. 12, pp. 11 135–11 151, 2023.

[3] A. A. Nurdiawati, L. Handoko, H. N. Amrullah, D. Dermawan, M. Shah, F. Hamzah *et al.*, "Human error probability analysis of overhead crane operation in steel fabrication company using slim-dematel-anp method," in *MATEC Web of Conferences*, vol. 204. EDP Sciences, 2018, p. 05012.

[4] S. Lahres, H. Aschemann, O. Sawodny, and E. P. Hofer, "Crane automation by decoupling control of a double pendulum using two translational actuators," in *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No. 00CH36334)*, vol. 2. IEEE, 2000, pp. 1052–1056.

[5] D. Qian and J. Yi, "Hierarchical sliding mode control for under-actuated cranes," *Springer, Heidelberg*, vol. 10, pp. 978–3, 2016.

[6] H. X. Le, T. V. Nguyen, A. V. Le, T. A. Phan, N. H. Nguyen, and M. X. Phan, "Adaptive hierarchical sliding mode control using neural network for uncertain 2d overhead crane," *International Journal of Dynamics and Control*, vol. 7, pp. 996–1004, 2019.

[7] B. Lu, Y. Fang, and N. Sun, "Enhanced-coupling adaptive control for double-pendulum overhead cranes with payload hoisting and lowering," *Automatica*, vol. 101, pp. 241–251, 2019.

[8] M. Zhang, X. Ma, X. Rong, X. Tian, and Y. Li, "Adaptive tracking control for double-pendulum overhead cranes subject to tracking error limitation, para-

metric uncertainties and external disturbances," *Mechanical Systems and Signal Processing*, vol. 76, pp. 15–32, 2016.

[9] L. Ramli, Z. Mohamed, A. M. Abdullahi, H. I. Jaafar, and I. M. Lazim, "Control strategies for crane systems: A comprehensive review," *Mechanical Systems and Signal Processing*, vol. 95, pp. 1–23, 2017.

[10] E. M. Abdel-Rahman, A. H. Nayfeh, and Z. N. Masoud, "Dynamics and control of cranes: A review," *Journal of Vibration and control*, vol. 9, no. 7, pp. 863–908, 2003.

[11] B. Rong, X. Rui, L. Tao, and G. Wang, "Dynamics analysis and fuzzy anti-swing control design of overhead crane system based on riccati discrete time transfer matrix method," *Multibody System Dynamics*, vol. 43, pp. 279–295, 2018.

[12] A. Khalid, J. Huey, W. Singhose, J. Lawrence, and D. Frakes, "Human operator performance testing using an input-shaped bridge crane," 2006.

[13] H. Chen, P. Yang, and Y. Geng, "A time optimal trajectory planning method for overhead cranes with obstacle avoidance," in *2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2019, pp. 697–701.

[14] Z. Wu and X. Xia, "Optimal motion planning for overhead cranes," *IET Control Theory & Applications*, vol. 8, no. 17, pp. 1833–1842, 2014.

[15] L. Tuan, G. Kim, and S. Lee, "Design of sliding mode controller for the 2d motion of an overhead crane with varying cable length," *Journal of Mechanism and Machine Theory*, 2016.

[16] H.-H. Lee, Y. Liang, and D. Segura, "A sliding-mode antiswing trajectory control for overhead cranes with high-speed load hoisting," 2006.

[17] J. H. Yang and K. S. Yang, "Adaptive coupling control for overhead crane systems," *Mechatronics*, vol. 17, no. 2-3, pp. 143–152, 2007.

[18] T. L. Mien, "Adaptive fuzzy sliding mode control for gantry crane as varying rope length," *International Journal of Engineering and Technology*, vol. 8, no. 4, pp. 1784–1791, 2016.

[19] D. Liu, J. Yi, D. Zhao, and W. Wang, "Adaptive sliding mode fuzzy control for a two-dimensional overhead crane," *Mechatronics*, vol. 15, no. 5, pp. 505–522, 2005.

[20] X. Wu and X. He, "Enhanced damping-based anti-swing control method for underactuated overhead cranes," *IET Control Theory & Applications*, vol. 9, no. 12, pp. 1893–1900, 2015.

[21] L.-H. Lee, P.-H. Huang, Y.-C. Shih, T.-C. Chiang, and C.-Y. Chang, "Parallel neural network combined with sliding mode control in overhead crane control system," *Journal of Vibration and Control*, vol. 20, no. 5, pp. 749–760, 2014.

[22] J. Huang, Z. Liang, and Q. Zang, "Dynamics and swing control of double-pendulum bridge cranes with distributed-mass beams," *Mechanical Systems and Signal Processing*, vol. 54, pp. 357–366, 2015.

[23] R. Tang and J. Huang, "Control of bridge cranes with distributed-mass payloads under windy conditions," *Mechanical Systems and Signal Processing*, vol. 72, pp. 409–419, 2016.

[24] T. Y. Jian and Z. Mohamed, "Modelling and sway control of a double-pendulum overhead crane system," *Applications of Modelling and Simulation*, vol. 1, no. 1, pp. 15–21, 2017.

[25] W. Singhose, D. Kim, and M. Kenison, "Input shaping control of double-pendulum bridge crane oscillations," 2008.

[26] J. Smoczek, "Fuzzy crane control with sensorless payload deflection feedback for vibration reduction," *Mechanical Systems and Signal Processing*, vol. 46, no. 1, pp. 70–81, 2014.

[27] H. Chen, Y. Fang, and N. Sun, "A swing constrained time-optimal trajectory planning strategy for double pendulum crane systems," *Nonlinear Dynamics*, vol. 89, pp. 1513–1524, 2017.

[28] R. Mar, A. Goyal, V. Nguyen, T. Yang, and W. Singhose, "Combined input shaping and feedback control for double-pendulum systems," *Mechanical Systems and Signal Processing*, vol. 85, pp. 267–277, 2017.

[29] H. Jaafar, Z. Mohamed, M. Ahmad, N. Wahab, L. Ramli, and M. Shaheed, "Control of an underactuated double-pendulum overhead crane using improved model reference command shaping: Design, simulation and experiment," *Mechanical Systems and Signal Processing*, vol. 151, p. 107358, 2021.

[30] H. Jaafar, Z. Mohamed, M. Shamsudin, N. M. Subha, L. Ramli, and A. Abdullahi, "Model reference command shaping for vibration control of multimode flexible systems with application to a double-pendulum overhead crane," *Mechanical Systems and Signal Processing*, vol. 115, pp. 677–695, 2019.

[31] H. Jaafar, Z. Mohamed, N. Mohd Subha, A. Husain, F. Ismail, L. Ramli, M. Tokhi, and M. Shamsudin, "Efficient control of a nonlinear double-pendulum overhead crane with sensorless payload motion using an improved pso-tuned pid controller," *Journal of Vibration and Control*, vol. 25, no. 4, pp. 907–921, 2019.

[32] G. Weiping, L. Diantong, Y. Jianqiang, and Z. Dongbin, "Passivity-based-control for double-pendulum-type overhead cranes," in *2004 IEEE Region 10 Conference TENCON 2004.*, vol. 500. IEEE, 2004, pp. 546–549.

[33] Z. Liu, S. Wang, H. Chen, G. Li, and H. Zhang, "Dynamic modeling and anti-swing control of double pendulum payload system of overhead crane," in *2022 IEEE 17th International Conference on Control & Automation (ICCA).* IEEE, 2022, pp. 279–284.

[34] T. Wang, N. Tan, J. Zhou, X. Zhang, and J. Zhang, "A novel sliding mode control method for double pendulum crane," in *Proceedings of the 3rd International Conference on Information Technologies and Electrical Engineering*, 2020, pp. 321–327.

[35] H. Ouyang, J. Wang, G. Zhang, L. Mei, and X. Deng, "Novel adaptive hierarchical sliding mode control for trajectory tracking and load sway rejection in double-pendulum overhead cranes," *Ieee Access*, vol. 7, pp. 10 353–10 361, 2019.

[36] M. Miry *et al.*, "Nonlinear state feedback controller combined with rbf for nonlinear underactuated overhead crane system," *Journal of Engineering Research*, vol. 9, no. 3A, 2021.

[37] C. Niu and H. Ouyang, "Nonlinear dynamic analysis of lifting mechanism of an electric overhead crane during emergency braking," *Applied Sciences*, vol. 10, no. 23, p. 8334, 2020.

[38] V.-A. Le, H.-X. Le, L. Nguyen, and M.-X. Phan, "An efficient adaptive hierarchical sliding mode control strategy using neural networks for 3d overhead cranes," *International Journal of Automation and Computing*, vol. 16, no. 5, pp. 614–627, 2019.

Appendix A

Appendix 1

A.1 Objective function

$$\boldsymbol{E}\left(\boldsymbol{k}\right) = \mathbf{1/2}(\boldsymbol{y}\left(\boldsymbol{k}\right) - \boldsymbol{y_n}(\boldsymbol{k}))^{\mathbf{2}} \tag{A.1}$$

A.2 Gradient decent method(offline training of RBFFNN)

$$\Delta w\left(j\right) = -\gamma_w \frac{\partial E\left(k\right)}{\partial w\left(j\right)} = \gamma_w \left(\boldsymbol{y}\left(\boldsymbol{k}\right) - \boldsymbol{y_n}\left(\boldsymbol{k}\right)\right) h(j)$$

$$w = w(1) + \Delta w + \alpha(w(1) - w(2))$$

$$\Delta b\left(j\right) = -\gamma_b \frac{\partial E\left(k\right)}{\partial b\left(j\right)} = \gamma_c \left(\boldsymbol{y}\left(\boldsymbol{k}\right) - \boldsymbol{y_n}\left(\boldsymbol{k}\right)\right) w\left(j\right) h\left(j\right) b\left(j\right)^3 \|x - c_j\|^2$$

$$b = b(1) + \Delta b + \alpha(b(1) - b(2))$$

$$\Delta c\left(i,j\right) = -\gamma_c \frac{\partial E\left(k\right)}{\partial c\left(i,j\right)} = \gamma_c \left(\boldsymbol{y}\left(\boldsymbol{k}\right) - \boldsymbol{y_n}\left(\boldsymbol{k}\right)\right) w\left(j\right) h\left(j\right) \left(x - c\left(i,j\right)\right) b\left(j\right)^2$$

$$c = c(1) + \Delta c + \alpha(c(1) - c(2))$$

A.3 Initial weights of RBFNN for online function estimation

0.00019	0.001174	0.000708	0.001368	0.00130	0.000283	0.000105	0.00028	0.000163	0.000241
0.000695	0.000867	-0.00032	0.000765	0.00107	0.000241	0.000289	0.000257	0.000165	0.000277
0.001045	0.000454	0.001482	0.000432	0.00113	0.000176	0.00022	0.000292	0.000325	0.000186
0.000575	0.001455	0.001621	0.000161	0.00034	0.000282	0.000289	6.48E-05	0.000215	0.000263
0.001545	0.000654	-1.88E-05	0.000674	0.001	0.000171	0.00022	0.00016	0.000182	0.000197
0.001083	-6.76E-05	0.000214	0.002026	0.00127	8.63E-05	0.000254	0.000266	0.000204	0.000105
-0.00014	-4.03E-06	0.001796	0.00142	0.00065	0.000212	0.000219	0.000173	0.000173	0.00011
0.000422	0.001004	0.001801	0.000927	3.22E-0	0.000296	0.000232	0.000153	0.000141	0.000284
0.001812	0.000903	0.001346	0.000881	0.00065	0.000127	0.000184	8.12E-05	0.000238	0.000139
0.001186	0.001872	0.001536	0.001536	0.00056	0.000251	0.000232	0.000142	0.00024	0.000256
0.000576	0.001378	0.00032	0.001222	0.00150	0.000118	0.000141	0.000141	0.000213	0.000176
0.001168	0.001662	0.000519	0.000878	0.00155	0.000236	0.000224	0.000255	0.000138	0.0002
0.001795	0.002211	0.000279	0.00074	0.00148	0.000163	0.000149	0.000108	0.000128	0.000128
0.000191	0.001157	0.001526	0.00034	0.00154	0.000153	0.000346	0.000157	0.000232	7.80E-05
0.00124	0.001654	0.000685	0.001501	0.0009	0.000164	0.000334	0.000152	0.000247	0.00021
0.000675	0.001172	0.000496	0.000779	0.00156	0.000253	0.000109	0.000205	0.000314	0.000253
0.000653	0.000648	0.00191	-0.00013	0.00056	0.000186	0.000154	0.000173	0.000142	0.000108
0.000844	0.001345	0.001094	0.00182	0.00149	0.000324	0.000293	0.000264	0.00024	0.000143
1.93E-05	0.001269	0.000766	0.000264	0.00084	0.000194	0.000135	0.00022	0.000184	0.000327
0.000963	0.001359	0.000361	0.001183	0.00044	0.000217	0.000162	8.65E-05	0.0002	0.00017

Figure A.1: $W_{01} \in R^{100\times 2}$

0.00019	0.001174	0.000708	0.001368	0.00130	0.000283	0.000105	0.00028	0.000163	0.000241
0.000695	0.000867	-0.00032	0.000765	0.00107	0.000241	0.000289	0.000257	0.000165	0.000277
0.001045	0.000454	0.001482	0.000432	0.00113	0.000176	0.00022	0.000292	0.000325	0.000186
0.000575	0.001455	0.001621	0.000161	0.00034	0.000282	0.000289	6.48E-05	0.000215	0.000263
0.001545	0.000654	-1.88E-05	0.000674	0.001	0.000171	0.00022	0.00016	0.000182	0.000197
0.001083	-6.76E-05	0.000214	0.002026	0.00127	8.63E-05	0.000254	0.000266	0.000204	0.000105
-0.00014	-4.03E-06	0.001796	0.00142	0.00065	0.000212	0.000219	0.000173	0.000173	0.00011
0.000422	0.001004	0.001801	0.000927	3.22E-0	0.000296	0.000232	0.000153	0.000141	0.000284
0.001812	0.000903	0.001346	0.000881	0.00065	0.000127	0.000184	8.12E-05	0.000238	0.000139
0.001186	0.001872	0.001536	0.001536	0.00056	0.000251	0.000232	0.000142	0.00024	0.000256
0.000576	0.001378	0.00032	0.001222	0.00150	0.000118	0.000141	0.000141	0.000213	0.000176
0.001168	0.001662	0.000519	0.000878	0.00155	0.000236	0.000224	0.000255	0.000138	0.0002
0.001795	0.002211	0.000279	0.00074	0.00148	0.000163	0.000149	0.000108	0.000128	0.000128
0.000191	0.001157	0.001526	0.00034	0.00154	0.000153	0.000346	0.000157	0.000232	7.80E-05
0.00124	0.001654	0.000685	0.001501	0.0009	0.000164	0.000334	0.000152	0.000247	0.00021
0.000675	0.001172	0.000496	0.000779	0.00156	0.000253	0.000109	0.000205	0.000314	0.000253
0.000653	0.000648	0.00191	-0.00013	0.00056	0.000186	0.000154	0.000173	0.000142	0.000108
0.000844	0.001345	0.001094	0.00182	0.00149	0.000324	0.000293	0.000264	0.00024	0.000143
1.93E-05	0.001269	0.000766	0.000264	0.00084	0.000194	0.000135	0.00022	0.000184	0.000327
0.000963	0.001359	0.000361	0.001183	0.00044	0.000217	0.000162	8.65E-05	0.0002	0.00017

Figure A.2: $W_{02} \in R^{100\times 2}$

A.4 Updated Sample weights of RBFNN

2.80E-04	4.49E-04	5.35E-04	5.24E-04	1.55E-0
4.68E-04	2.05E-04	4.51E-04	3.37E-04	3.74E-0
4.94E-04	9.37E-05	4.56E-04	1.73E-04	4.96E-0
8.31E-05	2.14E-05	1.69E-04	-1.15E-04	6.42E-0
4.18E-04	1.07E-06	-1.08E-05	3.71E-04	-1.08E-0
-5.34E-05	4.20E-04	-2.40E-05	-1.05E-04	-4.93E-0
1.10E-04	3.12E-04	4.87E-04	6.28E-04	5.36E-0
5.72E-04	3.74E-04	-9.45E-05	5.15E-04	7.15E-0
4.52E-04	5.54E-05	5.87E-04	2.65E-04	1.69E-0
5.03E-04	2.45E-04	9.92E-05	-3.84E-07	4.58E-0
2.66E-04	3.48E-04	5.01E-05	3.16E-04	-1.18E-0
-1.47E-05	1.88E-04	4.00E-05	2.02E-04	2.39E-0
4.21E-04	-4.42E-05	1.30E-04	7.62E-05	6.29E-0
1.40E-04	6.12E-04	2.53E-04	3.04E-04	4.41E-0
1.18E-04	3.89E-04	4.08E-04	5.99E-05	2.16E-0
2.15E-04	1.05E-04	3.15E-05	1.08E-04	1.58E-0
5.04E-04	3.79E-04	-9.85E-05	5.17E-04	1.87E-0
4.21E-04	3.99E-04	-3.91E-05	1.08E-04	2.90E-0
4.68E-04	1.71E-04	-6.65E-05	-1.72E-06	1.71E-0
1.18E-04	-3.76E-05	1.26E-04	-3.52E-05	4.66E-0

-2.19E-04	6.97E-04	-1.07E-04	-5.72E-04	-9.19E-04
0.001121	-0.00142	-1.22E-04	9.93E-04	-7.44E-04
-9.06E-04	-4.09E-04	-0.0012	-2.65E-04	2.33E-04
-2.93E-04	8.19E-04	2.23E-05	4.68E-04	-0.00134
-0.00102	7.51E-05	1.45E-04	-6.24E-04	-4.17E-04
-5.61E-05	0.001052	1.76E-04	9.44E-04	-6.97E-04
-7.13E-04	2.57E-04	0.001148	0.001439	-8.44E-04
-1.53E-04	-8.74E-04	6.08E-04	-0.00152	-0.00188
3.83E-04	-4.95E-04	-0.00109	-9.42E-04	1.42E-04
1.92E-04	-0.00145	1.36E-04	1.01E-04	9.95E-04
-6.17E-04	-1.87E-04	7.52E-04	9.20E-04	-2.58E-04
-7.11E-05	-1.17E-04	0.001166	-3.11E-04	5.20E-05
4.04E-04	0.001285	2.14E-04	-0.00138	-0.00114
-7.54E-05	-7.83E-05	0.00159	-4.85E-04	6.66E-04
3.35E-04	3.85E-04	8.72E-05	7.58E-04	-2.07E-04
8.30E-04	2.38E-04	-8.68E-04	8.46E-04	4.17E-04
-5.16E-04	0.00134	-7.32E-04	-0.00152	2.50E-04
-2.11E-04	-0.00103	-0.00104	-7.16E-04	-9.04E-04
0.001353	-3.87E-04	-6.39E-04	-2.59E-04	-2.76E-04
-2.67E-04	-8.42E-04	-7.37E-04	2.77E-04	-0.00108

Figure A.3: $\hat{W}_1 \in R^{100\times 2}$

1.58E-04	0.001174	7.08E-04	0.001368	0.00130
6.95E-04	8.66E-04	-3.20E-04	7.65E-04	0.00107
0.001053	4.54E-04	0.001482	4.32E-04	0.00113
5.70E-04	0.001455	0.00162	1.61E-04	3.46E-0
0.001577	6.54E-04	-1.88E-05	6.74E-04	0.00149
0.001086	-6.75E-05	2.14E-04	0.002025	0.00127
-1.83E-04	-4.03E-06	0.001796	0.00142	6.55E-0
4.15E-04	0.001004	0.001801	9.27E-04	3.22E-0
0.001854	9.03E-04	0.001346	8.81E-04	6.59E-0
0.00121	0.001872	0.001536	0.001536	5.68E-0
5.76E-04	0.001378	3.20E-04	0.001221	0.00150
0.001168	0.001661	5.19E-04	8.78E-04	0.00155
0.001794	0.002211	2.79E-04	7.40E-04	0.00148
1.91E-04	0.001156	0.001525	3.40E-04	0.00154
0.00124	0.001653	6.85E-04	0.001501	9.80E-0
6.75E-04	0.001172	4.96E-04	7.79E-04	0.00156
6.53E-04	6.48E-04	0.00191	-1.34E-04	5.66E-0
8.44E-04	0.001344	0.001094	0.001819	0.00149
1.93E-05	0.001268	7.65E-04	2.64E-04	8.46E-0
9.63E-04	0.001358	3.61E-04	0.001183	4.48E-0

2.86E-04	1.05E-04	2.80E-04	1.63E-04	2.41E-04
2.40E-04	2.89E-04	2.57E-04	1.65E-04	2.77E-04
1.76E-04	2.20E-04	2.92E-04	3.25E-04	1.86E-04
2.82E-04	2.88E-04	6.48E-05	2.15E-04	2.63E-04
1.71E-04	2.20E-04	1.60E-04	1.82E-04	1.97E-04
8.51E-05	2.54E-04	2.66E-04	2.03E-04	1.05E-04
2.12E-04	2.19E-04	1.73E-04	1.73E-04	1.10E-04
2.98E-04	2.31E-04	1.53E-04	1.41E-04	2.84E-04
1.23E-04	1.84E-04	8.12E-05	2.38E-04	1.39E-04
2.51E-04	2.31E-04	1.42E-04	2.40E-04	2.56E-04
1.18E-04	1.41E-04	1.41E-04	2.13E-04	1.76E-04
2.36E-04	2.24E-04	2.55E-04	1.38E-04	2.00E-04
1.63E-04	1.49E-04	1.08E-04	1.28E-04	1.28E-04
1.53E-04	3.45E-04	1.57E-04	2.32E-04	7.80E-05
1.64E-04	3.34E-04	1.52E-04	2.47E-04	2.10E-04
2.53E-04	1.09E-04	2.05E-04	3.14E-04	2.53E-04
1.86E-04	1.54E-04	1.73E-04	1.42E-04	1.08E-04
3.24E-04	2.93E-04	2.64E-04	2.40E-04	1.43E-04
1.94E-04	1.35E-04	2.20E-04	1.84E-04	3.27E-04
2.17E-04	1.62E-04	8.65E-05	2.00E-04	1.70E-04

Figure A.4: $\hat{W}_2 \in R^{100\times 2}$

Appendix B

Appendix 2

B.1 Inside RBFNN

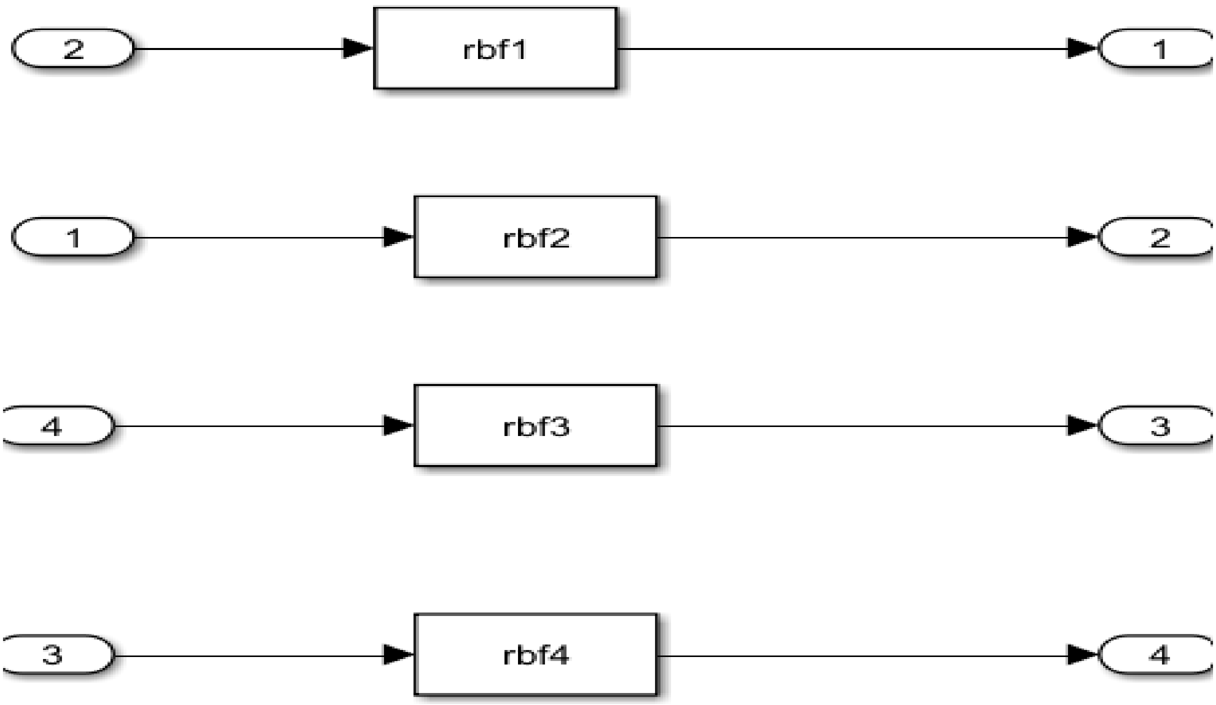


Figure B.1: RBFNN

B.2 Inside error dynamics

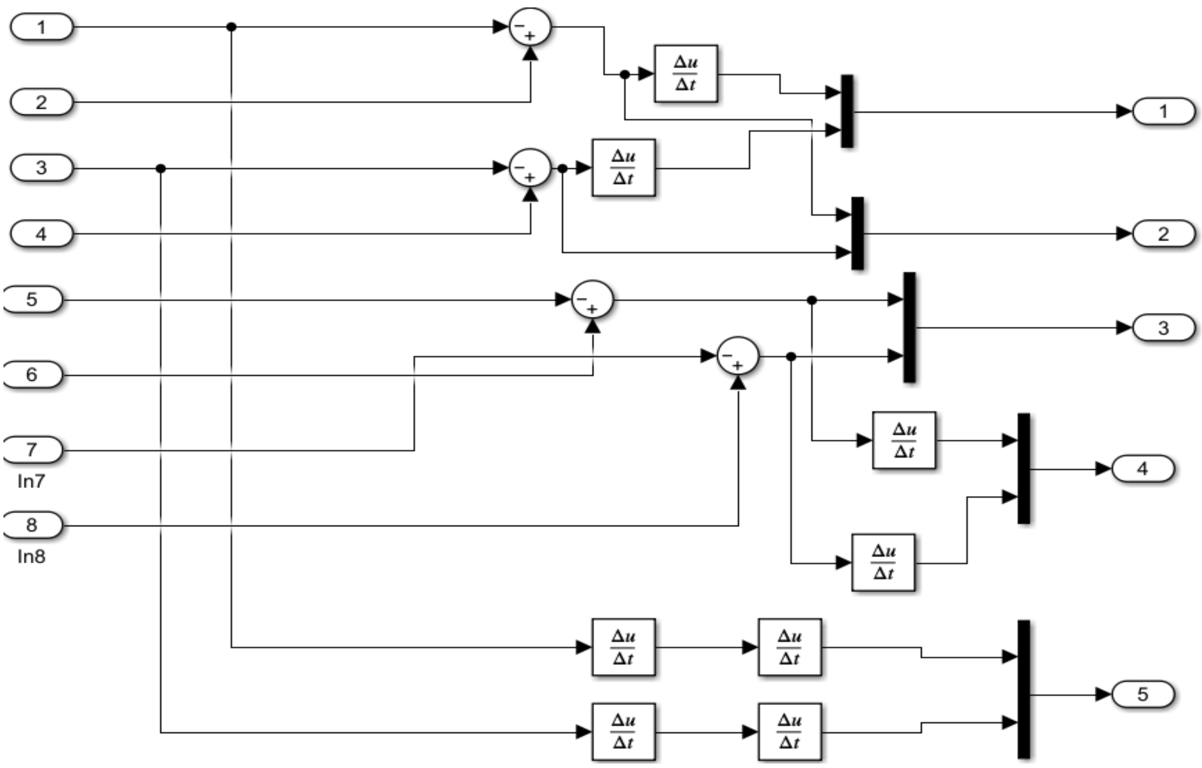


Figure B.2: Error dynamics

Appendix C

Appendix 3

C.1 S-function MATLAB code for plant dynamics

function [sys,x0,str,ts,simStateCompliance] = dynamicmodel(t,x,u,flag)

switch flag,

case 0, [sys,x0,str,ts,simStateCompliance]=mdlInitializeSizes; case 1,

sys=mdlDerivatives(t,x,u);

case 2,

sys=mdlUpdate(t,x,u);

case 3,

sys=mdlOutputs(t,x,u);

case 4,

sys=mdlGetTimeOfNextVarHit(t,x,u);

case 9,

sys=mdlTerminate(t,x,u);

otherwise

DAStudio.error('Simulink:blocks:unhandledFlag', num2str(flag));

end

function [sys,x0,str,ts,simStateCompliance]=mdlInitializeSizes

sizes = simsizes;

sizes.NumContStates = 12;

sizes.NumDiscStates = 0;

sizes.NumOutputs = 12;

sizes.NumInputs = 2;

sizes.DirFeedthrough = 0;

sizes.NumSampleTimes = 1;

sys = simsizes(sizes);

x0 = [0 0 0.4 0 0.3489 0 0.3489 0 0 0 0 0];

str = [];

ts = [0 0];

simStateCompliance = 'UnknownSimState';

function sys=mdlDerivatives(t,x,u)

m=9.2;

m1=2;

m2=1;

l2=0.6;

g=9.81;

m11=m+m1+m2;

m22=m1+m2;

$m44 = m2 * l2^2$;

m12=(m1+m2)*sin(x(5));

m13=(m1+m2)*x(3)*cos(x(5));

m14=m2*l2*cos(x(7));

m24=m2*l2*sin(x(5)-x(7));

m34=m2*l2*x(3)*cos(x(5)-x(7));

$m33 = (m1 + m2) * x(3)^2$;

M11=[m11 m12;m12 m22];

M12=[m13 m14;0 m24];

M21=[m13 0;m14 m24];

M22=[m33 m34;m34 m44];

M=[M11 M12;M21 M22];

l12=2*m22*cos(x(5))*x(6);

l13=2*m22*(cos(x(5))*x(4)-x(3)*sin(x(5))*x(6));

```
l14=-m2*l2*sin(x(7))*x(8);
l23=-m22*x(3)*x(6);
l24=-m2*l2*cos(x(5)-x(7))*x(8);
l32=2*m22*x(3)*x(6);
l33=2*m22*x(3)*x(4);
l34=m2*x(3)*l2*sin(x(5)-x(7))*x(8);
l42=2*m2*l2*cos(x(5)-x(7))*x(6);
l43=2*m2*l2*cos(x(5)-x(7))*x(4)-m2*x(3)*l2*sin(x(5)-x(7))*x(6);
L11=[0 l12;0 0];
L12=[l13 l14;l23 l24];
L21=[0 l32;0 l42];
L22=[l33 l34;l43 0];
L=[L11 L12;L21 L22];
g1=m22*g*(1-cos(x(5)));
g2=m22*g*x(3)*sin(x(5));
g3=m2*g*l2*sin(x(7));
G1=[0;g1];
G2=[g2;g3];
G=[G1;G2];
u=[u(1);u(2)];
MM1=M11-M12*inv(M22)*M21;
LL11=-M12*inv(M22)*L21+L11;
LL12=-M12*inv(M22)*L22+L12;
GG1=G1-M12*inv(M22)*G2;
MM2=-M21*inv(M11)*M12+M22;
LL21=-M21*inv(M11)*L12+L22;
LL22=-M21*inv(M11)*L11+L21;
GG2=-M21*inv(M11)*G1+G2;
H=-M21*inv(M11);
```

```
qad=[x(2);x(4)];
qud=[x(6);x(8)];
gn1=MM1;
gn2=MM2;
qadd=-inv(MM1)*(LL11*qad+LL12*qud+GG1-u);
qudd=-inv(MM2)*(LL21*qud+LL22*qad+GG2-H*u);
fa=-inv(MM1)*(LL11*qad+LL12*qud+GG1);
fu=-inv(MM2)*(LL21*qud+LL22*qad+GG2);
sys(1)=x(2);
sys(2)=qadd(1);
sys(3)=x(4);
sys(4)=qadd(2);
sys(5)=x(6);
sys(6)=qudd(1);
sys(7)=x(8);
sys(8)=qudd(2);
sys(9)=fa(1);
sys(10)=fa(2);sys(11)=fu(1);sys(12)=fu(2);
    function sys=mdlUpdate(t,x,u)
sys = [];
    function sys=mdlOutputs(t,x,u)
    sys(1) = x(1);sys(2) = x(2);sys(3) = x(3);sys(4) = x(4);sys(5) = x(5);sys(6) =
x(6);sys(7) = x(7);sys(8) = x(8);sys(9) = x(9); sys(10)=x(10);sys(11)=x(11);sys(12)=x(12);
function sys=mdlGetTimeOfNextVarHit(t,x,u)
    sampleTime = 1; sys = t + sampleTime;
    function sys=mdlTerminate(t,x,u)
    sys = [];
```

C.2 Sample Matlab code for rbf1

```
function [sys,x0,str,ts,simStateCompliance] = rbf1(t,x,u,flag)
    switch flag,
    case 0,
    [sys,x0,str,ts,simStateCompliance]=mdlInitializeSizes; case 1, sys=mdlDerivatives(t,x,u);
    case 2,
sys=mdlUpdate(t,x,u);
    case 3,
sys=mdlOutputs(t,x,u);
    case 4,
sys=mdlGetTimeOfNextVarHit(t,x,u);
    case 9,
sys=mdlTerminate(t,x,u);
    otherwise
DAStudio.error('Simulink:blocks:unhandledFlag', num2str(flag));
    end
    function [sys,x0,str,ts,simStateCompliance]=mdlInitializeSizes
sizes = simsizes;
sizes.NumContStates = 100;
sizes.NumDiscStates = 0;
sizes.NumOutputs = 1;
sizes.NumInputs = 10;
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;
sys = simsizes(sizes);
load w1file w1 ,c1,b1;
str=[];
ts=[0 0];
```

```
simStateCompliance = 'UnknownSimState';
function sys=mdlDerivatives(t,x,u)
s1=u(1);
s2=u(2);
x1p=u(3);
x2p=u(4);
x3p=u(5);
x4p=u(6);
x5p=u(7);
x6p=u(8);
x7p=u(9);
x8p=u(10);
xp=[x1p x2p x3p x4p x5p x6p x7p x8p]';
h=zeros(100,1);
for j=1:1:100
h(j) = exp(-norm(xp - c1(:,j))^2/(2*b1(j)^2));
end
    F1=0.0003*eye(100);
lamda1=[5 0;0 20];
W1=x(1:100);
s=[s1;s2];
dW1=F1*(h*s'*lamda1(:,1)-0.2*norm(s)*W1);
for i=1:1:100
sys(i)=dW1(i);
end
function sys=mdlUpdate(t,x,u)
    sys = [];
    function sys=mdlOutputs(t,x,u)
s1=u(1);
```

```
s2=u(2);
x1p=u(3);
x2p=u(4);
x3p=u(5);
x4p=u(6);
x5p=u(7);
x6p=u(8);
x7p=u(9);
x8p=u(10);
xp=[x1p x2p x3p x4p x5p x6p x7p x8p]';
h=zeros(100,1);
for j=1:1:100
h(j) = exp(-norm(xp - c1(:, j))^2/(2 * b1(j)^2));
end
W1=x(1:100);
f1=W1'*h;
sys(1)=f1;
function
sys=mdlGetTimeOfNextVarHit(t,x,u)
sampleTime = 1;
sys = t + sampleTime;
function sys=mdlTerminate(t,x,u)
sys = [];
```

C.3 Matlab code for ARBFNN-HSMC

```
function [u, s,s1, s2]=controler(x,f1,f2,f3, f4,ead,ea,eu,eud,qaddd)
m=9.2; m1=2; m2=1; l2=0.6; g=9.8; m11=m+m1+m2; m22=m1+m2;
m44 = m2 * l2^2; m12=(m1+m2)*sin(x(5)); m13=(m1+m2)*x(3)*cos(x(5));
```

m14=m2*l2*cos(x(7)); m24=m2*l2*sin(x(5)-x(7)); m34=m2*l2*x(3)*cos(x(5)-x(7));

$m33 = (m1 + m2) * x(3)^2$; l12=2*m22*cos(x(5))*x(6);

l13=2*m22*(cos(x(5))*x(4)-x(3)*sin(x(5))*x(6));

l14=-m2*l2*sin(x(7))*x(8); l23=-m22*x(3)*x(6); l24=-m2*l2*cos(x(5)-x(7))*x(8); l32=2*m22*x(

l33=2*m22*x(3)*x(4); l34=m2*x(3)*l2*sin(x(5)-x(7))*x(8); l42=2*m2*l2*cos(x(5)-

x(7))*x(6);

l43=2*m2*l2*cos(x(5)-x(7))*x(4)-m2*x(3)*l2*sin(x(5)-x(7))*x(6); g1=m22*g*(1-cos(x(5)));

g2=m22*g*x(3)*sin(x(5)); g3=m2*g*l2*sin(x(7));

M11=[m11 m12;m12 m22];

M12=[m13 m14;0 m24];

M21=[m13 0;m14 m24];

M22=[m33 m34;m34 m44];

M=[M11 M12;M21 M22];

L11=[0 l12;0 0];

L12=[l13 l14;l23 l24];

L21=[0 l32;0 l42];

L22=[l33 l34;l43 0];

L=[L11 L12;L21 L22]; G1=[0;g1]; G2=[g2;g3]; G=[G1;G2];

alpha1=[0.3 0;0 5];

alpha2=[6 0;0 16];

lamda1=[5 0;0 20];

lamda2=[-1.5 0;0 0.06];

k2=[0.0001 0;0 0.01];

k1=[2.5 0;0 30];

s1=[ead(1);ead(2)]+alpha1*([ea(1);ea(2)]);

s2=[eud(1);eud(2)]+alpha2*([eu(1);eu(2)]);

s=lamda1*s1+lamda2*s2;

MM1=M11-M12*inv(M22)*M21;

MM2=-M21*inv(M11)*M12+M22;

```
H=-M21*inv(M11);
u=(inv(lamda1*inv(MM1)+lamda2*inv(MM2)*H)*(-lamda1*([f1;f2])-lamda2*([f3;f4])
+lamda1*qaddd-lamda1*alpha1*ead-lamda2*alpha2*eud-k1*s-k2*sign(s)));
end
```