Addis Ababa University

Addis Ababa Institute of Technology

Department of Electrical and Computer Engineering

DESIGN OF FUZZY SLIDING MODE CONTROLLER FOR THE BALL AND PLATE SYSTEM

A thesis submitted to Addis Ababa Institute of Technology, School of Graduate Studies, Addis Ababa University

in partial fulfillment of the requirement for the Degree of Master of Science in Electrical Engineering (**Electrical Control Engineering).**

By

Andinet Negash Hunde

Advisor: Professor N.P. Singh

ADDIS ABABA

JULY 2011

Addis Ababa University

Addis Ababa Institute of Technology

Department of Electrical and Computer Engineering

DESIGN OF FUZZY SLIDING MODE CONTROLLER FOR THE BALL AND PLATE SYSTEM

By Andinet Negash Hunde

APPROVED BY BOARD OF EXAMINERS

_____	_____
Chairman, Department of Graduate Committee	Signature
Professor N.P. Singh _____	_____
Advisor	Signature
_____	_____
Internal Examiner	Signature
_____	_____
External Examiner	Signature

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my advisor, Professor N.P. Singh, for his guidance, support and encouragement during the course of the thesis work. He has been an excellent advisor demonstrating good patience and enormous help throughout my thesis study.

I would also like to thank the academic staff of Electrical and Computer Engineering Department, colleagues included, for their vital comments, criticism and advice on the thesis development and its content during each seminar presentations and in other informal discussions. Honestly, their involvement was undeniably so vital to play a significant role in shaping the content of the thesis in its final form.

Finally, I shall acknowledge my family for their patience and unreservedly continuous encouragement to the end of my graduate study.

_____

Andinet Negash Hunde

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

VRML	……	Virtual Reality Modeling Language
ITAE	……	Integral of Time Multiplied by Absolute Error
ISE	……	Integral of Square Error
3-D	……	Three Dimension (al)
B&P	……	Ball and Plate
DOF	……	Degree of Freedom
SISO	……	Single Input Single Output
AAIT	……	Addis Ababa Institute of Technology
FSM(C)	……	Fuzzy Sliding Mode (Control)
VSC(C)	……	Variable Structure (Control)
SMC	……	Sliding Mode Control
VR	……	Virtual Reality
AD	……	Analog to Digital
CCD	……	Charge Coupled Device
ISO	……	International Organization for Standardization
IEC	……	International Electrotechnical Commission
RMS	……	Root Mean Square
ave	……	Average
GA	……	Genetic Algorithm
PID	……	proportional–integral–derivative
PD	……	proportional–derivative
ADRC	……	Active Disturbance Rejection Control
ESO	……	Extended State Observer
LQR	……	Linear-Quadratic Regulator

LIST OF NOTATIONS

$\binom{n}{k}$	.....	binomial coefficient
$\approx$	.....	approximately equal to
$\equiv$	.....	equivalent to
$:=$	.....	define (defined as)
$\triangleq$	.....	defined as
$\forall$	.....	for every (all)
$\in$	.....	element of
$(.)^T$	.....	transpose
$(.)^{(n)}$	.....	n^{th}derivative
$\widetilde{(.)}$	.....	fuzzy set
$\widetilde{(.)}$	.....	estimated (observed) state
$(.)^*$	.....	reference signal
μ	.....	membership function
ξ	.....	damping ratio
τ	.....	torque
$k_{t,e}$	.....	torque/viscous damping constant
e	.....	error
ω, n	.....	(angular) speed
J	.....	moment of inertia, performance index

ABSTRACT

The Ball and Plate system is a non-linear, multivariable and under-actuated system which provides a suitable experimental platform for control algorithm development and investigation of both stabilization and trajectory tracking control of unstable systems.

Sliding mode control, as one of the tools available to design robust controllers, is introduced in the outer loop of a double-loop feedback configuration. Since the robustness of sliding mode control is obtained at the cost of infinite switching of the control input, undesirable phenomenon known as chattering will be a concern in practical implementations. To this end, Fuzzy Logic is used to tune the gain of the switching control component based on the distance of the system trajectory from the sliding surface. Genetic algorithm is implemented to determine the parameters of the fuzzy system in an optimal manner.

Linear algebraic method is used to design an inner loop angle controller from a set of Diophantine equations. An implementable zero-position error transfer function is selected from tabulated results of analog computer simulations of Integral of Time Multiplied Absolute Error (ITAE) optimal systems. Specifications for the desired settling time and limitation in the actuator voltage are used as constraints to obtain the coefficients of the overall transfer function.

The mathematical model of the Ball and Plate system, solved from Euler-Lagrange Equations of Motion, is investigated by realizing the designed controllers using Simulink ® and Real-Time Windows Target™ rapid prototyping software. The 3-D model of the system designed using V·Realm™ Builder based on the standards of Virtual Reality Modeling Language (VRML) is interfaced with the Simulink ® model via Simulink® 3D Animation™ product.

Simulation results show that the ball could be stabilized anywhere on the plate in 3.5 seconds and it could also track a circular trajectory of 0.4m radius at 0.8 rad/s in 10 seconds without significant chattering.

Key Words: Fuzzy Sliding Mode Control, Variable Structure Control, Linear Algebraic Method, Ball and Plate control, VRML, Simulink® 3D Animation™/MATLAB®.

CHAPTER 1
INTRODUCTION

1.1. BACKGROUND

Though the extent and range of applications might vary, various educational models are either directly developed or acquired in several universities across the world in an effort to complement the theoretical teaching and learning process. In particular, educational models are available to the research community at large both at undergraduate and post graduate study levels to facilitate the testing of control algorithms in the laboratories in technological institutions. In a similar manner these models are being exploited to a great extent in the development of engineering devices for industrial or household applications as preliminary benchmarking tools before the commencement of major mass productions.

The increasing tendency in the preferential usage of educational models in both institutional and commercial research areas has its main purpose which springs from the ability to comfortably test real systems on simplified but fairly close replica of real systems in the laboratory platforms. In many circumstances, the direct experimentation of control algorithms on real systems in the laboratory is either impossible or costly or perhaps might even prove fatal. For instance, in order to design and test a control strategy for a state of the art humanoid robot for industrial applications, it will be unwise and uneconomical to experiment with the robot itself; however, an inverted pendulum, which mimics a walking human being in a less complex manner, will be the right choice for preliminary research as the costs and risks associated with the experimentation on the inverted pendulum are acceptable.

Likewise, the Ball and Plate (B&P) system- which is the controlled object in this thesis, is gaining popularity and is arousing the attention of several researchers in a range of disciplines. The B&P systems could be described in simple terms as a system constituting of rectangular (circular) flat plate fixed movably at its center by a pivot so that the plate could execute free inclinations in two orthogonal directions while the (steel) ball keeps rolling freely on the plate. Hence, the objective would be to command the ball to be placed at a given position on the plate or to instruct the Ball to follow a given trajectory without the ball tipping off the edges of the plate. From simple intuition, it is impossible to keep the ball at a fixed

position(s) on the plate without the ball showing the tendency to roll off the edges even in the absence of disturbances like wind gust, meaning- the problem is open loop unstable.

Unlike other educational models, the B&P system has several interesting features for the study of unstable and nonlinear systems. Firstly, models like Magnetic Levitation, Inverted Pendulum and Ball and Beam are all one-dimensional with degree of freedoms (DOF): one (1), two (2), and two (2) respectively. The B&P system has four (4) DOF i.e. with two independent motions of the ball about the plane of the plate and another two motions of the combined ball and plate inclinations (rotations) about the fixed pivot at the center of the plate. Secondly, the B&P system offers an opportunity to study trajectory tracking problems in addition to the stability problems that is the only subject that could be studied in the remaining educational models mentioned above. Thirdly, research in the control laboratory at our institution is limited to SISO systems; the multivariable property of the B&P system could be utilized to extend the scope of the research. So the advantage is clearly evident.

The B&P system was firstly established in Rockwell laboratory of Czechoslovakia University in the mid 1990s [1]. Important improvements in the mechanical design by some universities in America and Asia were introduced in the years to follow.

Nowadays commercial educational model manufactures like HUMUSOFT, GUNT Hamburg, Googol Technology and TecQuipment develop an integrated system with all the associated sensors and drives.

1.2. PROBLEM DESCRIPTION

The B&P apparatus is a two dimensional electromechanical device which can be further categorized as nonlinear, multivariable (with two inputs and two outputs) and unstable system **[**1]. Furthermore, the system is under-actuated as it possesses more degrees of freedom than the number of available actuators [2]. Thus, the fact that the ball's motion is indirectly controlled by changing the plate inclination in two orthogonal directions is quite an interesting challenge to consider.

The mathematical model of the B&P system as obtained from Euler-Lagrange Equations of Motion yields an 8^{th}-order state variable system. This mathematical model which has coupling and nonlinearity terms is further simplified by a set of reasonable assumptions and Euler approximation for small angle.

For a more effective control of the Ball and Plate system, a double feedback loop structure (a loop within a loop) is utilized [1]. The inner loop is designed as an actuator (angular) position controller for the plate inclination while the outer loop is implemented so as to control the ball's (linear) position on the plate. Thus, while the controller in the outer loop computes the angle by which the plate should move to balance the ball, the inner control loop actually moves the plate by that angle [1].

The inner control loop design is based on linear algebraic method. An overall transfer function is chosen that minimizes the Integral of Time Multiplied by Absolute Error (ITAE) and a Two-Parameter configuration is used for the implementation of the compensators that are obtained from the solution of a Diophantine equation.

Because of the existence of uncertainty due to friction, measurement time delays and parameter uncertainties, practical applications require nonlinear control methods to be adopted in the design of outer control loop [1]. In order to fulfill this requirement, a controller design technique based on Sliding Mode Control is implemented in the outer control loop. The robustness in Sliding Mode Control is achieved as a result of high frequency switching in the control signal to which no practical device could respond without significant delays. Therefore, fuzzy logic is introduced in the outer control loop to tune the switching gain based on the distance of the system trajectory from the sliding surface.

Genetic algorithm is employed to determine the parameters of the fuzzy membership functions in an optimal manner.

1.3. OBJECTIVE OF THE THESIS

1.3.1. General Objectives

The general objectives of the thesis are:

i. To apply robust and optimal control methods such as Sliding Mode Control, Fuzzy Logic Control and Linear Algebraic Method to control the B&P system for both static and trajectory tracking commands.

1.3.2. Specific Objectives

The specific objectives of the thesis are:

i. To study the mathematical model of the B&P system for control design.
ii. To select the appropriate actuator parameters for the control of the B&P system.

iii. To design the inner loop angle controller using Linear Algebraic Method.

iv. To design the outer loop position controller using Sliding Mode Control and apply Fuzzy Logic techniques to eliminate chattering in the control signal introduced by the high switching action of SMC.

v. To develop the Virtual Reality of the B&P system and evaluate the performance of the proposed controller using Simulink® 3D Animation™ as the interface between the MATLAB® and the VR models.

1.4. CONTRIBUTION OF THE THESIS

The contributions of this thesis are:

i. The choice of appropriate parameters of sliding surface that make the surface asymptotically stable is an important design decision. In this thesis, the principle of design for controllers in double control loop configuration is employed to select the parameter for the sliding surface.

ii. This thesis proposes the use of a Two-parameter configuration which is based on Linear Algebraic Method /Inward Approach/ to design the inner loop controller. In this approach, an optimal implementable transfer function is chosen first to meet design specifications and then the required compensators are obtained from a set of Diophantine equations.

iii. The idea of applying the hybrid algorithm Fuzzy and Sliding Mode Control and (FSMC) is relatively new and FSMC has never been applied to the control of the B&P system so far. This thesis proposes a new method to the control of the B&P system.

iv. The B&P system is an important device to be listed in our laboratory's inventory owing to its range of applications that could be studied on the platform. This thesis contributes towards the mathematical modeling, appropriate actuator parameter selection, an exemplary controller design using Fuzzy Sliding Mode Control and/or Linear Algebraic Method and Virtual Reality (VR) model development for the B&P system.

1.5. OUTLINE OF THE THESIS

Chapter I presents the background of the B&P system, problem statement and the objective as well as the contribution of the thesis. C*hapter II* begins by introducing the model of the B&P System based on Euler-Lagrange Equations of Motion followed by a detailed account of literature survey on the B&P system. A description of concepts related to Sliding Mode Control, Fuzzy Logic, Linear Algebraic Method, Virtual Reality Modeling Language and Simulink® 3D Animation™ is presented in *chapter III*. *Chapter IV* is dedicated to the design of controllers for the B&P System. In this chapter, the modeling and parameter selection for the actuator is discussed. An inner servo angle control loop is designed using Linear Algebraic Method and implemented using the Two-Parameter Configuration. Sliding Mode Control design principles are applied to the outer position control loop followed by the introduction of Fuzzy Logic Control methods to eliminate the undesirable chattering exhibited in the control signal. The parameters of the Fuzzy sets are determined in an optimal manner using genetic algorithm. *Chapter V* is devoted to the investigation of the performance of Sliding Mode Control and comparison of the results obtained with the proposed Fuzzy Sliding Mode Control for both static and trajectory tracking problems of the B&P system. Conclusions, recommendations and suggestions for future work are all presented in *Chapter VI*.

CHAPTER 2
THE BALL AND PLATE SYSTEM

2.1. INTRODUCTION

The B&P model is an extension of the classical Ball and Beam system. It consists of a rectangular (circular) flat plate fixed movably at its center by a pivot. The control target is the plate which could rotate about 2 mutually perpendicular axes, with the ultimate aim of balancing a free rotating ball. The control problem of this system is to control the position of the B&P system for both static positions and desired path tracking. The inclination of the plate can be manipulated in two perpendicular directions, so that the tilting of the plate will make the ball move on the plate.

Static position tracking control of the Ball and Plate system is to hold the ball at a specific position on the plate. Dynamic position tracking control demands the ball to follow a given position reference. Static and dynamic position tracking control could be studied simultaneously on the Ball and Plate system. The block diagram of the B&P system is shown in Figure 2.1. Briefly the control procedure is described as follows:

1. The position sensor obtains the position of the ball on the plate. In most applications, the sensor is either a camera or a touch-screen device.
2. Position data is feedback to the control system where the reference angle is calculated either for position command or trajectory tracking problems. If a camera is used as the sensor, the position data is obtained by image processing techniques that identify the target object from its background and determine its position from the analog frame of picture.
3. The rotation of the plate along the two mutually orthogonal directions are facilitated by two actuators and thus balancing the ball to the command position or along the desired trajectory. The actuators often used for this application are either DC-motors or stepper motors and in only one instance a pneumatic device is reported [3].

The block-diagram representation of the control of the B&P system shown in Figure 2.1 is for one of the two orthogonal directions.

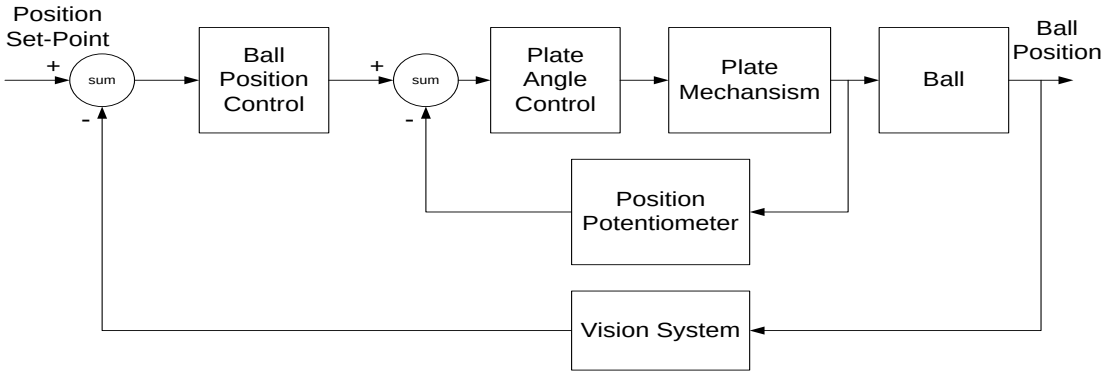


Figure 2.1: Block diagram of the B&P system

The B&P system is available from several manufactures like HUMUSOFT, GUNT Hamburg, Googol Technology and TecQuipment. A pictorial representation of the B&P system is shown below.

Figure 2.2: The B&P system developed by Googol Technology (Left) and HUMUSOFT (Right)

The Ball and Plate (B&P) system is unstable, nonlinear, multivariable, and under-actuated. Owing to these properties together with the opportunity to experiment both the stabilization of the ball to static position and trajectory tracking problems simultaneously on the same

platform, its popularity is surging in recent publications and among practicing researchers in a range of disciplines.

Thus, owing to these features the B&P model is an excellent alternative that could be used in the study of stabilization and trajectory tracking researches in such disciplines as control theory, robotics, optimal control, nonlinear control, multivariable control, etc.

2.2. MATHEMATICAL MODEL OF THE B&P SYSTEM

2.2.1. Euler-Lagrange Equations of Motion

The dynamical equations of J.L. Lagrange represent a powerful alternative to the classical Newton equations and are particularly useful for systems having many degrees of freedom. In the modeling of the Ball and Plate, the method of Lagrange shall be used. To begin with, let us define the Lagrangian function as follows [4]:

$$L(q_i, \dot{q}_i, t) = T(\dot{q}_i, t) - V(q_i, t) \quad (2.1)$$

Then, the Euler-Lagrange equations are given by:-

$$\frac{d}{dt}\left(\frac{\partial L(q_i, \dot{q}_i, t)}{\partial \dot{q}_i}\right) - \frac{\partial L(q_i, \dot{q}_i, t)}{\partial q_i} = F_{q_i}, 1 \leq i \leq n \quad (2.2)$$

where

$T(\dot{q}_i, t)$ is the kinetic energy with respect to the inertial axes

$V(q_i, t)$ is the potential enrgy

n is the degree of freedom

q_1 to q_n are the generalized co-ordinates

F_{q_1} to F_{q_n} are the generalized forces

A set of system co-ordinates is said to be complete if the values of the co-ordinates are sufficient to fix the state of the entire system. A set of system co-ordinates is said to be independent if, when all except one co-ordinate are fixed, it is still possible to vary that co-ordinate over a continuous range of admissible values. Since any complete, independent set of variables is sufficient to characterize the system state, such variables are called generalized

co-ordinates. [4] That is, a set of generalized co-ordinates is one in which each co-ordinate is independent and the number of co-ordinates is just sufficient to specify completely the configuration of the system.

2.2.2. **Nonlinear Model of the Ball and Plate System**

In order to simplify the mathematical modeling, the following assumptions are made:-[5]

i. The ball never losses contact with the plate
ii. There is no slithering(sliding) between the ball and the plate
iii. All friction forces and rotational moments are being neglected
iv. There is no limitation in the plate angle and the plate's surface.

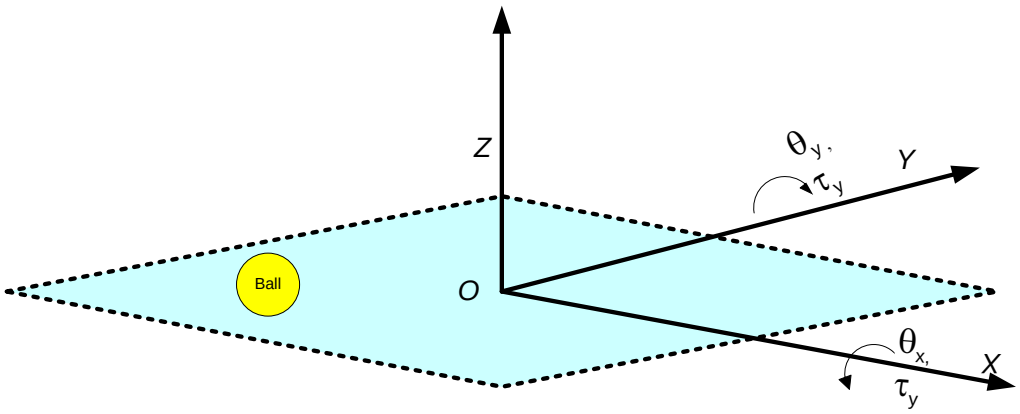


Figure 2.3: Schematic diagram of the Ball and Plate system

The kinetic energy T can be divided into four components:-

1. Translational kinetic energy of the ball

   The ball's position measured from the origin of the co-ordinate system is given by:-

$$r = x\hat{\imath} + y\hat{\jmath} + (x\sin\theta_x + y\sin\theta_y)\hat{k} \qquad (2.3)$$

   Solving for the velocity of ball using $v = \dot{r}$, the kinetic energy becomes:-

$$\frac{1}{2}mv^2 = \frac{1}{2}m\{\dot{x}\hat{\imath} + \dot{y}\hat{\jmath} + [\dot{x}\sin\theta_x + x\dot{\theta}_x\cos\theta_x + \dot{y}\sin\theta_y + y\dot{\theta}_y\cos\theta_y]\hat{k}\}^2 \qquad (2.4)$$

Using Euler's formula (i.e. $e^{ix} = \cos x + \text{i}\sin x$) to expand $\sin\theta_{x,y}$ and $\cos\theta_{x,y}$ in a Taylor series about the equilibrium point $(x, y, \theta_x, \theta_y) = (0,0,0,0)$, we get[1] $\sin\theta_x = 0\ and\ \cos\theta_y = 1$. Therefore, equation (2.4) could be reduced to the following simpler form:-

$$\frac{1}{2}mv^2 = \frac{1}{2}m\left\{\dot{x}^2 + \dot{y}^2 + \left[x\dot{\theta}_x + y\dot{\theta}_y\right]^2\right\} \qquad (2.5)$$

2. The ball's rotational kinetic energy about its center of gravity is given by:-

$$\frac{1}{2}J_B\omega^2 = \frac{1}{2}J_B\{\frac{V'}{R}\}^2 \qquad (2.6)$$

where $V' = \dot{x}\hat{\imath} + \dot{y}\hat{\jmath}$ is the velocity of the ball with respect to the origin *O* in a co-ordinate system aligned with the plane of the plate and *R* is the radius of the ball. Therefore, substitution of $V' = \dot{x}\hat{\imath} + \dot{y}\hat{\jmath}$ in equation (2.5) gives:-

$$\frac{1}{2}J_B\omega^2 = \frac{1}{2}\frac{J_B}{R^2}\{\dot{\text{x}}^2 + \dot{\text{y}}^2\} \qquad (2.7)$$

where J_B is the moment of inertia of the ball about its center.

3. The rotational kinetic energy of the combined B&P system with respect to the co-ordinate system.

$$\frac{1}{2}\{J_B\}\left\{\dot{\theta_x}^2 + \dot{\theta_y}^2\right\} + \frac{1}{2}J_{P_x}\dot{\theta_x}^2 + \frac{1}{2}J_{P_y}\dot{\theta_y}^2 \qquad (2.8)$$

where $J_{P_x}$and $J_{P_y}$are the moments of inertia of the plate about the ***X*** and ***Y*** axes respectively. $\theta_x and\ \theta_y$ are the inclinations of the rectangular plate about the ***X*** and ***Y*** axes, respectively.

On the other hand, the potential energy V of the Ball and Plate system relative to a horizontal plane through the origin of the co-ordinate system is obtained as:-

$$V = -mg(x\sin\theta_x + y\sin\theta_y) \qquad (2.9)$$

Therefore, the Lagrangian will now take the form:-

[1] $\cos x = \sum_{n=0}^{\infty}(-1)^n\frac{x^{2n}}{(2n)!}$ and $\sin x = \sum_{n=0}^{\infty}(-1)^n\frac{x^{2n+1}}{(2n+1)!}$

$$L(q_i, \dot{q}_i) = \frac{1}{2} m \left\{ \dot{x}^2 + \dot{y}^2 + \left[x\dot{\theta}_x + y\dot{\theta}_y \right]^2 \right\} + \frac{1}{2} \frac{J_B}{R^2} \{ \dot{x}^2 + \dot{y}^2 \} + \frac{1}{2} \{ J_B \} \left\{ \dot{\theta_x}^2 + \dot{\theta_y}^2 \right\} + \frac{1}{2} J_{P_x} \dot{\theta_x}^2 + \frac{1}{2} J_{P_y} \dot{\theta_y}^2 + mg(x \sin\theta_x + y \sin\theta_y) \quad (2.10)$$

For the sake of mathematical analysis, the Ball and Plate system can be simplified into a particle system consisting of two rigid bodies. The plate has three geometrical limitations in the translation along the x-axis, y-axis and z-axis. It also has a geometrical limitation in its rotation about the z-axis. It has two degrees of freedom (DOF) in the rotation about the x-axis and y-axis. The ball has a geometrical limit in the translation along the z-axis; therefore it has 2 degrees of freedom (DOF) in translations along the x-axis and y-axis. The model has four degrees of freedom where x, y, θ_x and θ_y are chosen as the generalized co-ordinates:- $q_1 = x$, $q_2 =$ y, $q_3 = \theta_x$ and $q_4 = \theta_y$. [2]

Using Lagrange's equation of motion, i.e. equation (2.2), we obtain the following equations along each of the generalized coordinates:-

$$\begin{Bmatrix} \frac{d}{dt}\left(\frac{\partial L(q_i, \dot{q}_i, t)}{\partial \dot{x}}\right) - \frac{\partial L(q_i, \dot{q}_i, t)}{\partial x} = F_x, \\ \frac{d}{dt}\left(\frac{\partial L(q_i, \dot{q}_i, t)}{\partial \dot{y}}\right) - \frac{\partial L(q_i, \dot{q}_i, t)}{\partial y} = F_y, \\ \frac{d}{dt}\left(\frac{\partial L(q_i, \dot{q}_{i,} t)}{\partial \dot{\theta}_x}\right) - \frac{\partial L(q_i, \dot{q}_i, t)}{\partial \theta_x} = \tau_x, \\ \frac{d}{dt}\left(\frac{\partial L(q_i, \dot{q}_{i,} t)}{\partial \dot{\theta}_y}\right) - \frac{\partial L(q_i, \dot{q}_i, t)}{\partial \theta_y} = \tau_y. \end{Bmatrix} \quad (2.11)$$

Substituting the expression for the Lagrangian $L(q_i, \dot{q}_i)$ obtained in equation (2.10) into equation (2.11) and noting that $F_x = F_y = 0$, we get the following results:-

$$\begin{Bmatrix} \frac{d}{dt}\left(\left\{\mathrm{m}+\frac{J_B}{R^2}\right\}\dot{x}\right)- \\ \left(m\dot{\theta}_x\{x\dot{\theta}_x+\ y\dot{\theta}_y\}+mg\sin\theta_x\ \right)\ =0 \\ \frac{d}{dt}\left(\left\{\mathrm{m}+\frac{J_B}{R^2}\right\}\dot{y}\right)- \\ \left(m\dot{\theta}_y\{x\dot{\theta}_x+\ y\dot{\theta}_y\}+\ mg\sin\theta_y\right)\ =0 \\ \frac{d}{dt}\left(mx\{x\dot{\theta}_x+\ y\dot{\theta}_y\}+(J_B\ +J_{P_x})\dot{\theta}_x\right)+ \\ (mgx\cos\theta_x)\ =\ \tau_{\theta_x} \\ \frac{d}{dt}\left(my\{x\dot{\theta}_x+\ y\dot{\theta}_y\}+(J_B\ +J_{P_y})\dot{\theta}_y\right)+ \\ \left(mgy\cos\theta_y\right)\ =\ \tau_{\theta_y} \end{Bmatrix} \quad (2.12)$$

Evaluating the derivatives and rearranging terms, we obtain four differential equations as follows:-

$$\left(m+\frac{J_B}{R^2}\right)\ddot{x}-mx\left(\dot{\theta}_x\right)^2-my\dot{\theta}_x\dot{\theta}_y=mg\sin\theta_x \quad (2.13)$$

$$\left(m+\frac{J_B}{R^2}\right)\ddot{y}-my\left(\dot{\theta}_y\right)^2-mx\dot{\theta}_x\dot{\theta}_y=mg\sin\theta_y \quad (2.14)$$

$$\left(mx^2+J_B+J_{P_x}\right)\ddot{\theta}_x+\ 2mx\dot{x}\dot{\theta}_x+mxy\ddot{\theta}_y +m(\dot{x}y+x\dot{y})\dot{\theta}_y=\tau_{\theta_x}-mgx\cos\theta_x \quad (2.15)$$

$$\left(my^2+J_B+J_{P_y}\right)\ddot{\theta}_y+\ 2my\dot{y}\dot{\theta}_y+mxy\ddot{\theta}_x +m(\dot{x}y+x\dot{y})\dot{\theta}_x=\tau_{\theta_y}-mgy\cos\theta_y \quad (2.16)$$

In these equations, m (kg) is the mass of the ball; R(m) is the radius of the ball; x (m) is the ball position along X-axis; y (m) is the ball position along Y-axis; $\dot{x}$ (m/s) and $\ddot{x}$ (m/s^2) are the velocity and acceleration respectively along X-axis; $\dot{y}$ (m/s) and $\ddot{y}$ (m/s^2) are the velocity and acceleration respectively along Y-axis; θ_x(rad) is the plate deflection angle about X-axis; $\dot{\theta}_x$ (rad/sec) is the plate deflection angular velocity about X-axis; θ_y (rad) is the plate deflection angular about Y-axis. $\dot{\theta}_y$ (rad/sec) is the plate deflection angular velocity about Y-axis; τ_{θ_x}

(Nm) is the torque exerted on the plate in X-axis direction and τ_{θ_y} (Nm) is the torque exerted on the plate in Y-axis direction.

Equations (2.13) and (2.14) describe the ball motion on the plate; they tell us that the acceleration of the ball movement depends on the angles and angular velocity of the plate inclination. Equations (2.15) and (2.16) tell how the plate inclination dynamics is influenced by the external driving forces.

2.2.3. **Simplification and Further Linearization**

Consider the state variable assignment X= $(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8)^T$ = $(x, \dot{x}, \theta_x, \dot{\theta}_x, y, \dot{y}, \theta_y, \dot{\theta}_y)^T$. From equations (2.15) and (2.16), since the mass and the moment of inertia of the ball is negligible compared to the moment of inertia of the plate we find that $J_{P_x}\ddot{\theta}_x = \tau_{\theta_x}$ or $\ddot{\theta}_x = u_x$and $J_{P_y}\ddot{\theta}_y = \tau_{\theta_y}$ or $\ddot{\theta}_y = u_y$ [13]. Then the B&P system can be represented in state-space form as follows [24, 52].

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\\ \dot{x}_5\\ \dot{x}_6\\ \dot{x}_7\\ \dot{x}_8\end{bmatrix} = \begin{bmatrix} x_2 \\ B(x_1 {x^2}_4 + x_4 x_5 x_8 + g\sin x_3) \\ x_4 \\ 0 \\ x_6 \\ B(x_5 {x^2}_8 + x_1 x_4 x_8 + g\sin x_7) \\ x_8 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix} \qquad (2.17)$$

$$y = [1\ 0\ 0\ 0\ 1\ 0\ 0\ 0] \begin{bmatrix} x_1\\ x_2\\ x_3\\ x_4\\ x_5\\ x_6\\ x_7\\ x_8 \end{bmatrix}$$

where $B = \frac{m}{\left(m + \frac{J}{R^2}\right)}$.

The Ball and Plate model is characterized by an 8th order nonlinear system, with coupling between X and Y motions; moreover it is a multivariable problem with double inputs and double outputs. In order to further simplify the model we introduce the following assumptions:-[3], [5] and [6]

1. The motor can control each step of the ball. This implies that skipping between steps doesn't occur and that the magnitude of plate moment doesn't affect the position of the rotor. In fact the system inputs are θ_x and θ_y and not the torque moments $\tau_{\theta_x}$and τ_{θ_y}, because the magnitude of the load moments do not affect the position of the motors. Due to this reason, equations (2.15) and (2.16) could be dropped out in the consecutive study of the Ball and Plate system.
2. In the steady state the plate should be in the horizontal position, where both inclination angles are equal to zero. If one assumes that the angle do not change much, ±30°, the sine function can be replaced by its argument [7].
3. In equations (2.13) and (2.14), the velocities $\dot{\theta}_x$ and $\dot{\theta}_y$ are small and hence have negligible effects when squared or multiplied together.

Based on these basic assumptions, linearized, simplified and uncoupled ordinary differential equations are obtained.

$$\left\{m + \frac{J_B}{R^2}\right\}\ddot{x} = mg\theta_x \qquad (2.18)$$

$$\left\{m + \frac{J_B}{R^2}\right\}\ddot{y} = mg\theta_y \qquad (2.19)$$

Substituting the moment of inertia of the ball, $J_B = \frac{2}{5}mR^2$ into equations (2.18) and (2.19) and noting that the state-space description is now split into two parts owing to the simplifying assumptions that led to the independence of motions in the X-axis and Y-axis, we get:

$$\frac{7}{5}\ddot{x} = g\theta_x \qquad (2.20)$$

$$\frac{7}{5}\ddot{y} = g\theta_y \qquad (2.21)$$

The state space representation of equations (2.20) and (2.21) could be realized by choosing state variables as: $x_1 = x, x_2 = \dot{x}\ and\ y_1 = y, y_2 = \dot{y}$

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}0\\ b_0\end{bmatrix}\theta_x \qquad (2.22)$$

$$\begin{bmatrix}\dot{y}_1\\ \dot{y}_2\end{bmatrix} = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}y_1\\ y_2\end{bmatrix} + \begin{bmatrix}0\\ b_0\end{bmatrix}\theta_y \qquad (2.23)$$

where $b_0 = \frac{5g}{7}$

Owing to the symmetry of the problem, from now on, we concentrate only on equation (2.22). The controllers of the B&P system are designed by considering dynamics in the X-direction independently and the results are equally applicable for motion described by equation (2.23). Furthermore, it is worth noting that the linearized approximate model still represents many typical real systems, such as the task of horizontally stabilizing an airplane during landing and in the presence of turbulent airflow [7].

2.3. LITERATURE REVIEW

A summary of literatures published regarding the B&P system is given next. Considerable care is taken to recognize and summarize publicly available research papers.

1. **Hongwei Liu and Yanyang Liang,** [1] proposed a path tracking sliding mode control algorithm for the ball position control and proved asymptotical stability of the close-loop system through Lyapunov stability theory.

   Peculiar to the thesis is the structure of the plate, instead of a single flat plate customarily used; two plates were used as shown in Figure 2.4. Plate 2 was embedded in plate 1 and rotates around X- axis which is fixed in the middle of Plate 1. Plate 1 rotates around a static Y-axis fixed to the fixture, and thus Plate 2 rotates around Y-axis too. Therefore, the steel ball can freely move on plate 2. Though, it was mentioned about the use of double loop configuration, the inner loop was not presented and the nature of the controllers used to design the inner loop are unknown. Moreover, sliding mode parameters were chosen arbitrary without any attempt to solve them from the problem at hand.

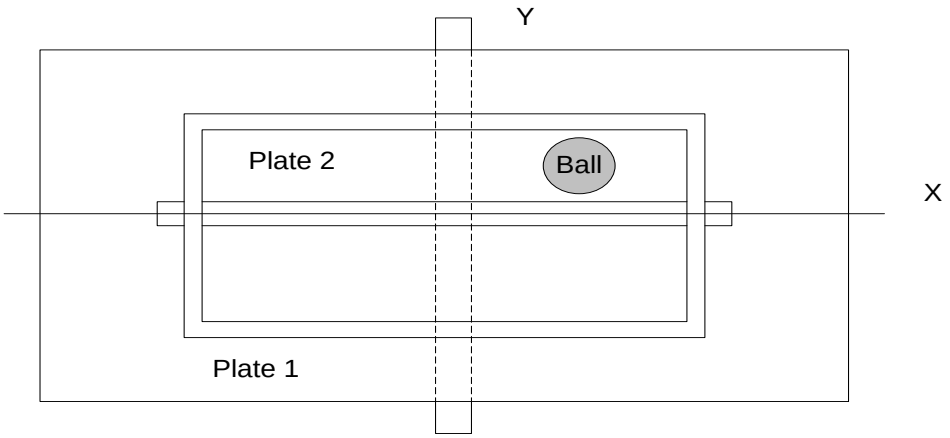


Figure 2.4: A proposed arrangement of the plate mechanism for the B&P system

A similar plate mechanism is used in [8]. Therefore, the same comment is given in (6) below about the proposed plate mechanism.

2. **Hongrui Wang, Yantao Tian, Zhen Sui, Xuefei Zhang and Ce Ding [**2**]** proposed a double-loop feedback structure for tracking control of the B&P system on the experimental platform BPVS-JLU I. The B&P system BPVS-JLU I was developed by control theory and intelligent system laboratory in Jilin University. The system consists of a CCD camera, an image processing card, two step motors and a step motor driving card, an industrial computer and the Ball and Plate system. Plate inclinations are measured by 2 photoelectric encoders. A single input rule modules fuzzy logic controller with dynamic importance degree was utilized in the external loop. As input items play unequal roles, dynamic importance degree is set to them. Input items that may improve system performance were strengthened, while input items that may degrade system performance were given small weights. A switching servo controller was used in the inner loop. Fuzzy control, bang-bang control and PD control were used in the inner feedback loop. Fuzzy logic was used to improve the servo dynamical performance. PD or bang-bang control was utilized to quicken up the system response. A switching scheme was employed to determine which algorithm needs to be chosen. Control programme was written in C++Language unlike [3] & [5].

For a given circular trajectory with radius $150\ mm$ and a desired ball tracking velocity of $3mm/s$, average position tracking error was reported as $10.761\ mm$. Although the designed controller in various aspects could be rated as fairly outstanding, the large number of Fuzzy rules employed and their individual tuning turns out to be the weaker part. Considering the inner loop alone, the plate angle error derivative was divided into 7 fuzzy subsets while the plate angle error was divided into 17 fuzzy subsets, meaning 7X17=119 rules were required! With trial-and-error method as the means to tune parameters of the fuzzy sets, it is important to notice an additional complexity added to the problem.

3. **De-hu Yuan** [3] proposed a pneumatic actuation mechanism which is composed of power-amplifying circuit board, proportional flow-controlled valve, oscillating cylinder and cam-push-rod mechanism for use on a commercially available B&P system manufactured by HUMUSOFT. Based on similar assumptions as in [5] and open-loop identification experiment, the transfer function of the B&P system was determined. Unlike [5], Yuan implemented a double-loop structure with the inner-loop and outer-loop compensated based, respectively, on PID and PD-Fuzzy controller design principles. The fuzzy controller is double-input and single-output. The position and velocity of the ball are the inputs, among which the position is measured by the touch screen and the velocity is obtained by the designed state observer. The inclination angle of the plate is the output of the controller. After normalizing the basic inputs and output of the 2-input and 1-output PD-Fuzzy controller, 9 rules were used to map the input-output relationships to the control signal (reference angle). It was stated that for a reference square wave input, tracking performance could be obtained in 3 seconds without significant overshoot. It wasn't clearly stated if the system was implemented, however, the simulation of the simplified model (excluding the model of the actuating mechanisms) was designed using Simulink®.
4. **Andrej Knuplei, Amor Chowdhur and Rajko SveEko** [5] have used commercially available product of HUMUSOFT called CE151 B&P to demonstrate stability of the ball. The apparatus uses a servo-drive with a controller and two stepping motors to control the inclination of the plate.

For measuring the ball position, an intelligent video system composed of CCD camera, picture-framing interface and program for real-time picture processing (supplied with the apparatus) are used. The picture-framing interface receives the analog video signal, which will be digitized with AD converter and saved as data-sheet onto the storage media. A computer determines, from the received data, the position of the ball and calculates the difference between reference and real value on which basis the controller generates a signal that is needed to drive the stepping motors.

The starting point for the modeling is the use of Lagrange's equation and basic simplifying assumptions also cited in other papers- these assumptions are referred while developing the mathematical model of the B&P system.

Lead compensator was chosen as a controller for the plant. Acceptable regions for the location of the two parameters, the gain and controller zero, are determined from Routh's stability criteria so as to ensure the stability of the closed loop transfer function. For optimization of the parameters the Integral of Square Error (ISE) was used. The simulation was carried out with Simulink® and Real-time Toolbox using MATLAB ®software.

Simulation results show that responses to a unit step and a sinusoidal input have a non-decaying error that was attributed to the non-linearity (irregular ball geometry) and of a limitation in sensor resolution.

5. **Dejun Liu, Yantao Tian and Huida Duan** [6] proposed, based on the nonlinear dynamic model obtained from Euler-Lagrange Equation, an observer for the uncertain items i.e. the coupling terms often neglected in the course of design. They applied the estimate so-obtained to design a sliding mode controller for the B&P system. The mathematical expression for the B&P model could also be rewritten from equation (2.17) as:

$$\ddot{x} = a\theta_x - b\dot{x} + F \qquad (2.24)$$

where $a = \frac{mg}{m+\frac{J_B}{R^2}}$, $F = \frac{m\dot{\theta}_x\dot{\theta}_y y}{m+\frac{J_B}{R^2}} + \frac{m(\dot{\theta}_x)^2 x}{m+\frac{J_B}{R^2}} + b\dot{x}; b$ is a contant. The idea is to design an observer $\check{F}$ so that when the error $F - \check{F}$ is sufficiently small the switching gain k could be designed as small as possible to decrease chattering. The sliding mode control with the estimated uncertain item, therefore, has the following form:

$$u = \frac{1}{a}\left(\lambda\dot{e} + \ddot{r} + b\dot{y} + \check{F} + ksgn(s)\right) \qquad (2.25)$$

Where r and y are respectively the input and output signals; whereas λ is a parameter of the sliding surface. The parameter λ is randomly chosen.

Simulations results were compared against bang-bang (the ordinary) sliding mode control showing satisfactory results for circular reference signal. The parameters for the B&P system are based on HUMUSOFT CE 151.

Results of the simulation do not include the effects of the actuator dynamics and cannot be compared with results from other papers.

6. **Greg Andrews, Chris Colasuonno and Aaron Herrmann** [8] based on a nonlinear dynamical model obtained from the General Equation of Motion for a Multibody system developed the mathematical model of the system. Further linearizations were made so as to apply linear control theory.

$$\boldsymbol{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{B}\left(\dot{\boldsymbol{\theta}}\right) + \boldsymbol{C}\left(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}\right)\dot{\boldsymbol{\theta}} + \boldsymbol{G}(\boldsymbol{\theta}) = \boldsymbol{\tau} \qquad (2.26)$$

where $\boldsymbol{M}$ is the mass-inertia matrix of the system, $\boldsymbol{B}$ is the friction term, $\boldsymbol{C}$ is the velocity coupling matrix, and $\boldsymbol{G}$ is the gravity loading vector.

The control scheme for the Ball on Plate Balancing System consists of two loops: an inner loop to control the motor torque based on a desired plate angle, and an outer loop to command a plate angle based on a desired ball location. Several control techniques including LQR, pole placement and Kalman filter were applied to the design of a custom-built B&P system. To better visualize the simulation results, a 3-D animation was generated using MATLAB's Virtual Reality Toolbox. The practical implementation is such that one of the motors was attached to the primary axis, while

the other motor is attached to the secondary axis which in turn is attached and rotated by the primary axis. An identical design was also proposed in [1] where one actuator turns the plate and the other actuator which increases the load moment of inertia of the first actuator. In general, this arrangement requires an expensive motor with higher torque requirement than is normally necessary and eliminates the advantages obtained as a result of the symmetrical design of the two independent motions.

7. **Huida Duan ,Yantao Tian and Guangbin Wang** [9] presented a paper on a control technique known as Active Disturbance Rejection Control (ADRC). The proposed ADRC was applied to the trajectory tracking control of the B&P system. In the ADRC framework, the disturbance and coupling are treated as additional state variables, which are then estimated using Extended State Observer (ESO) and compensated for in real time. For instance, consider a general second order plant as shown in equation (2.27) [10].

$$\ddot{y} = f(t, y, \dot{y}, \omega) + bu \qquad (2.27)$$

where ω is the external disturbance and b is a given (known) constant. Here $f(t, y, \dot{y}, \omega)$ represents the nonlinear time-varying dynamics of the plant that is not known. That is, for this plant, only the order and the parameter b are given. Instead of following the traditional design path of modeling and linearizing $f(t, y, \dot{y}, \omega)$ and then designing a linear controller, the ADRC approach seeks to actively compensate for the unknown dynamics and disturbances. In order to estimate $f(t, y, \dot{y}, \omega)$, without knowing its analytical form, equation (2.27) is augmented as:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 + bu, x_3 \triangleq f(t, y, \dot{y}, \omega), \\ \dot{x}_3 &= h(t), h(t) \triangleq \dot{f}(t, y, \dot{y}, \omega) \\ y &= x_1 \end{aligned} \qquad (2.28)$$

The reason for increasing the order of the plant is to make $f(t, y, \dot{y}, \omega)$ a state set such that a state observer can be used to estimate it. One such observer is given as:-

$$\dot{z}_1 = z_2 + l_1(x_1 - z_1)$$

$$\dot{z}_2 = z_3 + l_2(x_1 - z_1) + bu \qquad (2.29)$$

$$\dot{z}_3 = l_3(x_1 - z_1)$$

Turning our attention to the paper we started reviewing, for the purpose of simulation, the B&P design parameters were selected based on the HUMUSOFT CE 151 model. Though the results obtained were interesting, several parameters were needed to be tuned such as three observer gains and six other parameters used in the design.

8. **Nima Mohajerin, Mohammad Bagher Menhaj** [11] applied Fuzzy Logic techniques to control the static position of the Ball on the Plate. The widely acclaimed HUMUSOFT CE 151 B&P apparatus was used and the design idea was to (ideally) divide the plate into small rectangular regions so as to approximate the location of the rolling ball. These regions were used as an indication of the measure of the distance of the ball from the center of the plate. Control action was adjusted based on a Fuzzy mapping of the distance of the ball to the inclination angle of the plate. 11 rules were used to map the 11 input fuzzy variables with Gaussian membership functions that represent the crisp distance of the ball's position into 11 fuzzy variables with Gaussian membership functions that represent the inclination angle of the plate. It is clear that for the combined control of the ball in both orthogonal directions, a total of 22 rules are required. Determination of Fuzzy rules and membership functions is complicated and reliant on 'expert' knowledge. Therefore, with the lack of universally accepted design norms, the tuning of Fuzzy parameters such as the width of the membership functions is often obtained based on intuition (trial-and-error) and hence the method is time consuming.

9. **Miad Moarref, Mohsen Saadat and Gholamreza Vossoughi** [12] introduced an actuating system composed of stepper motors and a five link spatial parallel

mechanism. Since the motors were not directly connected to the plate, kinematical relations between the two were derived. Based on the dynamical equations of the B&P, double loop control structure and treating the multivariable B&P position control problem as two separate SISO systems, supervisory fuzzy logic was proposed to tune PD gains obtained from a root-locus design. The rules used to tune each gain of the PD controller were 49. Error and error derivative were the two fuzzy variables defined to tune the two PD gains, k_p, k_D. Seven (7) triangular membership functions were used requiring 7x7=49 rules and 49x2=98 rules to tune both PD gains for just one of the SISO systems. In the same paper, SMC design principles were applied on the B&P system. Based on the Lyapunov method, control law for the system was devised with the inevitable chattering being handled by a linear-piece-wise approximation of the switching (signum) function in the boundary layer. Plots show that the ball could be stabilized approximately in 8 seconds. Though control chatter was eliminated due to the replacement of the discontinuous signum function by the continous saturating function, the robustness properties are now severely compromised. The sliding surface will not be a line any more, it will be a band /region/ determined by the thickness of the boundary layer. On the other hand, the total number of rules (98x2=196) used to tune both SISO systems, are descriptive of the difficulties involved in applying Fuzzy Logic for control purposes.

10. **Xinghe Fan, Naiyao Zhang and Shujie Teng** [13] implemented a hierarchical fuzzy control scheme on the educational model of HUMUSOFT B&P apparatus. The scheme was composed of three levels. The lowest level is a fuzzy tacking controller; the middle level is a fuzzy supervision controller that takes actions in extreme situations; and the top level is a fuzzy planning controller that determines the desired trajectory. In order to optimize the ball's moving trajectory, genetic algorithm was introduced to adjust the output membership functions of the fuzzy planning controller. The lowest level fuzzy tracking controller alone requires adjusting 16x2=32 fuzzy rules. The challenges here are similar to the ones given for [12].

CHAPTER 3
METHODOLOGY

3.1. INTRODUCTION

Sliding mode control has been applied for nonlinear systems which face model uncertainties, parameter variations, and disturbances [14]. The main reason for this success stems from the fact that during the design of a switching surface, a predetermined dynamics can be defined and as long as the system is in the sliding mode, it follows the predefined trajectory. While this implies that the system dynamics cannot be affected by bounded disturbances, such enhancement has been achieved at the expense of chattering in the control action. Chattering is undesirable as it introduces drastic changes in the control variable which consequently can damage the actuator, and more importantly, may excite unmodeled high frequency dynamics.
To overcome these problems, fuzzy logic has been introduced to reduce the control chatter by decelerating the system state in the reaching mode as it approaches the switching surface. Various researches on the use of Fuzzy Logic in the design of a chatter free sliding surface are reported. Furthermore, contrary to this procedure, a different design approach which uses Sliding Mode Control principles to help design Fuzzy Logic Control known as Sliding Mode Fuzzy Control (SMFC) is reported in literatures. The discussion in this thesis is limited to the more popular Fuzzy Sliding Mode Control (FSMC).

3.2. SLIDING MODE CONTROL

Sliding Mode Control (SMC) is recognized as a powerful control technique that is capable of making a closed-loop system robust with respect to bounded plant parameter variations and bounded external disturbances. In the most basic theoretical description, a sliding mode control uses a bounded control, switching infinitely fast, to drive the state space trajectory onto a surface that is designed independent of the plant parameters. Surface parameters are directly related to the closed-loop dynamics, and the state space motion on this surface is called sliding mode. Since the surface parameters are chosen independent of the plant parameters, the closed-loop response can be made theoretically invariant to plant variations.

Despite such a power, actual use of variable structure control in practical engineering systems is somewhat limited. The reluctance to apply sliding mode control is often due to the concern that chattering exists in practical variable structure system (VSC) [15].

Let us consider a general system description with uncertainties and external disturbances of the form:

$$\begin{aligned}\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x},t) + \Delta\boldsymbol{f}(\boldsymbol{x},\boldsymbol{\theta},t) \\ + \{\boldsymbol{B}(\boldsymbol{x},t) + \Delta\boldsymbol{B}(\boldsymbol{x},\boldsymbol{\theta},t)\}\boldsymbol{u}(t) \\ + \boldsymbol{d}(\boldsymbol{x},\boldsymbol{\theta},t)\end{aligned} \qquad (3.1)$$

where $\boldsymbol{x}(t) \in R^n$ is the state vector, $\boldsymbol{u}(t) \in R^m$ is the control vector.

$\boldsymbol{f}(\boldsymbol{x},t),\ \Delta\boldsymbol{f}(\boldsymbol{x},\boldsymbol{\theta},t) \in R^n, \boldsymbol{B}(\boldsymbol{x},t), \Delta\boldsymbol{B}(\boldsymbol{x},\boldsymbol{\theta},t) \in R^{nxm},\ \Delta\boldsymbol{f}(\boldsymbol{x},\boldsymbol{\theta},t)$ and $\Delta\boldsymbol{B}(\boldsymbol{x},\boldsymbol{\theta},t)$ represent the system uncertainties, $\boldsymbol{\theta}$ is an unknown parameter vector and $\boldsymbol{d}(\boldsymbol{x},\boldsymbol{\theta},t)$ includes unmodelled dynamics and external disturbances [16]. It has been shown [17] that for certain $\Delta\tilde{\boldsymbol{f}}, \Delta\widetilde{\boldsymbol{B}}$ *and* $\Delta\widetilde{\boldsymbol{d}}$ if the matching conditions are satisfied

$$\Delta\boldsymbol{f}(\boldsymbol{x},\boldsymbol{\theta},\boldsymbol{t}) = \boldsymbol{B}(\boldsymbol{x},t)\Delta\tilde{\boldsymbol{f}}(\boldsymbol{x},\boldsymbol{\theta},t) \qquad (3.2)$$

$$\Delta\boldsymbol{B}(\boldsymbol{x},\boldsymbol{\theta},t) = \boldsymbol{B}(\boldsymbol{x},t)\Delta\widetilde{\boldsymbol{B}}(\boldsymbol{x},\boldsymbol{\theta},t)$$

$$\boldsymbol{d}(\boldsymbol{x},\boldsymbol{\theta},t) = \boldsymbol{B}(\boldsymbol{x},t)\Delta\widetilde{\boldsymbol{d}}(\boldsymbol{x},\boldsymbol{\theta},t)$$

the system will be invariant (better than just robust) to modeling error and disturbances. Under these conditions, equation (3.1) becomes [16]:

$$\begin{aligned}\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x},t) + \{\boldsymbol{B}(\boldsymbol{x},t)\}\boldsymbol{u}(t) \\ + \boldsymbol{B}(\boldsymbol{x},t)\boldsymbol{w}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{\theta},t)\end{aligned} \qquad (3.3)$$

where, $\boldsymbol{w}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{\theta},t)$ represents the total plant uncertainties, which is unknown and mostly compensated by setting parameters of $\boldsymbol{K} \in R^{mxm}$(positive definite diagonal matrix) to large values. However, large values of $\boldsymbol{K}$, could increase chattering [16]. Therefore, let us, consider a given control system represented by the state-variable form:-

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(\boldsymbol{x}, t) + \{\boldsymbol{B}(\boldsymbol{x}, t)\}\boldsymbol{u}(t) \quad (3.4)$$

The dimensions of $\boldsymbol{x}$ and $\boldsymbol{u}$ are given above.

The statement of the Sliding Mode Control problem can be stated as:

(1) Find m switching functions, $\boldsymbol{s}(\boldsymbol{x})$

(2) Find a variable structure control

$$\boldsymbol{u}(\boldsymbol{x}, t) = \begin{cases} \boldsymbol{u}^{+}(\boldsymbol{x}, t) \; when \; \boldsymbol{s}(\boldsymbol{x}) > 0 \\ \boldsymbol{u}^{-}(\boldsymbol{x}, t) \; when \; \boldsymbol{s}(\boldsymbol{x}) < 0 \end{cases} \quad (3.5)$$

such that the reaching modes satisfy the reaching condition, namely, reach the set $\boldsymbol{s}(\boldsymbol{x}) = 0$ (the switching surface) in finite time.

The physical meaning of above statement is as follows [16]:

(1) Design a switching surface $\boldsymbol{s}(\boldsymbol{x}) = \boldsymbol{0}$ to represent a desired system dynamics, which is of lower order than the given plant.

(2) Design a variable structure control $\boldsymbol{u}(\boldsymbol{x}, t)$ such that any state x outside the switching surface is driven to reach the surface in finite time. On the switching surface, the sliding mode takes place, following the desired system dynamics. In this way, the overall VSC system is globally asymptotically stable.

The Sliding Mode Control design for the B&P system in this thesis is both for the stabilization and trajectory tracking problems; therefore, a more useful restatement of the of the control problem is given as follows:-

Find a control law so that the state $\boldsymbol{x}$ can track a desired state $\boldsymbol{x}_d$ in the presence of model uncertainties and disturbances.

Let as in [16] $e = x - x_d$ be the tracking error in the variable x, and let $\boldsymbol{e} = \boldsymbol{x} - \boldsymbol{x_d} = [\boldsymbol{e}, \dot{\boldsymbol{e}}, \dots, \boldsymbol{e^{n-1}}]^{\boldsymbol{T}}$ be the tracking error vector. Furthermore, let us define a time-varying surface $\boldsymbol{s}(\boldsymbol{x}, t) = 0$, where

$$\boldsymbol{s}(\boldsymbol{x}, t) = \left(\frac{d}{dt} + \boldsymbol{\lambda}\right)^{\boldsymbol{n-1}} \boldsymbol{e} \quad (3.6)$$

and λ is a strictly positive constant. The interesting point about this surface is that it has a lower order than the original plant and in the case of a plant with order $n = 2$, it becomes a switching line. [14] It is important to note that while the system is in sliding mode, (i.e. while the state trajectories join the sliding surface) the dynamics neither depends on the plant parameters nor the disturbance. This so-called "invariance" property looks promising for designing feedback control for the dynamic plants operating under uncertainty conditions [19].

The major advantage of sliding mode is, therefore, low sensitivity to plant parameter variations and disturbances which eliminates the necessity of exact modeling. Furthermore, Sliding mode control enables the decoupling of the overall system motion into independent partial components of lower dimensions and, as a result, reduces the complexity of feedback design. Due to these properties the intensity of the research at many scientific centers of industry and universities is maintained at high level, and sliding mode control has been proved to be applicable to a wide range of problems [20].

The two questions that need to be addressed next are:

(1) How can we maintain the motion on the sliding surface once it has begun (sliding condition)
(2) How can we ensure that all state trajectories off the sliding surface be forced to it in finite time so as to make the surface globally asymptotically stable. (reaching condition)

3.2.1. **Control Law**

Sliding Mode Control Law

In the design of the sliding mode control, one must first find the equivalent control law, $\boldsymbol{u}_{eq}$, which will keep the state of the system on the sliding surface. The equivalent control law is found by recognizing that: $\dot{\boldsymbol{s}}|_{\boldsymbol{u}=\boldsymbol{u}_{eq}} = 0$. Differentiating $\boldsymbol{s}(\boldsymbol{x}, t)$with respect to t along the trajectories of equation (3.4) gives [21]

$$\dot{\boldsymbol{s}}(\boldsymbol{x},t) = \boldsymbol{e}^{(\boldsymbol{n})} + \sum_{\boldsymbol{k=1}}^{\boldsymbol{n-1}} \binom{n-1}{k} \boldsymbol{\lambda}^{\boldsymbol{k}} \boldsymbol{e}^{(\boldsymbol{n-k})} = \boldsymbol{A}(\boldsymbol{x},t) + \boldsymbol{B}(\boldsymbol{x},t)\boldsymbol{u}(t) - \boldsymbol{x}^{(n)}{}_{d} + \sum_{\boldsymbol{k=1}}^{\boldsymbol{n-1}} \binom{n-1}{k} \boldsymbol{\lambda}^{\boldsymbol{k}} \boldsymbol{e}^{(\boldsymbol{n-k})} \quad (3.7)$$

Thus, the equivalent control becomes:

$$\boldsymbol{u_{eq}} = \{\boldsymbol{B}(\boldsymbol{x},t)\}^{-1} \left\{ \boldsymbol{x}^{(n)}{}_{d} - \boldsymbol{A}(\boldsymbol{x},t) - \sum_{k=1}^{n-1} \binom{n-1}{k} \boldsymbol{\lambda}^{\boldsymbol{k}} \boldsymbol{e}^{(\boldsymbol{n-k})} \right\} \quad (3.8)$$

For stabilization problems, the derivative of the reference position is zero. Therefore, using the chain rule a more compact relation can be found [21]

$$\dot{\boldsymbol{s}}(\boldsymbol{x},t) = \frac{\partial \boldsymbol{s}}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}} = \frac{\partial \boldsymbol{s}}{\partial \boldsymbol{x}} \{\boldsymbol{A}(\boldsymbol{x},t) + \boldsymbol{B}(\boldsymbol{x},t)\boldsymbol{u}(t)\} \quad (3.9)$$

and setting $\dot{\boldsymbol{s}}|_{\boldsymbol{u}=\boldsymbol{u}_{eq}} = 0$,

$$\boldsymbol{u}_{eq} = -\left[\frac{\partial \boldsymbol{s}}{\partial \boldsymbol{x}} \boldsymbol{B}(\boldsymbol{x},t)\right]^{-1} \frac{\partial \boldsymbol{s}}{\partial \boldsymbol{x}} \boldsymbol{A}(\boldsymbol{x},t) \quad (3.10)$$

The non-singularity of the matrix $\left[\frac{\partial \boldsymbol{s}}{\partial \boldsymbol{x}} \boldsymbol{B}(\boldsymbol{x},t)\right]$ is mandatory to realize the reaching law.

Switching Control Law

The condition under which the state will move toward and reach a sliding surface is called a reaching condition. The system trajectory under the reaching condition is called the reaching mode or reaching phase. Three approaches for specifying the reaching condition have been proposed [17].

(1) The Direct Switching Function Approach: The earliest reaching condition proposed was

$$\dot{\boldsymbol{s}}_i > 0, when\ \boldsymbol{s}_i < 0 \qquad (3.11)$$

$$\dot{\boldsymbol{s}}_i < 0, when\ \boldsymbol{s}_i > 0$$

$$\equiv \dot{\boldsymbol{s}}_i \boldsymbol{s}_i < 0$$

The reaching condition is global but does not guarantee a finite reaching time.

(2) The Lyapunov Function Approach: By choosing a Lyapunov function candidate $V(\boldsymbol{x}, t) = \boldsymbol{s}^T \boldsymbol{s}$, a global reaching condition is given by

$$\dot{V}(\boldsymbol{x}, t) < 0\ when\ \boldsymbol{s} \neq 0 \qquad (3.12)$$

Finite reaching time is guaranteed by modifying equation (3.12) to

$$\dot{V}(\boldsymbol{x}, t) < -\epsilon\ when\ \boldsymbol{s} \neq 0, where\ \epsilon > 0 \qquad (3.13)$$

(3) The Reaching Law Approach: Based on the reaching law method, the Reaching Law Approach directly specifies the dynamics of the switching function. The most common structures in the reaching law are the constant rate, the constant plus proportional and the power rate reaching law which are given in their respective orders in equation (3.14)

$$\dot{\boldsymbol{s}} = -Ksgn(\boldsymbol{s}) \qquad (3.14)$$

$$\dot{\boldsymbol{s}} = -Ksgn(\boldsymbol{s}) - Q\boldsymbol{s}$$

$$\dot{s}_i = -k_i |s_i|^{\alpha} sgn(s_i)$$

$$0 < \alpha < 1, i = 1\ to\ m$$

where $sgn(s)$ is given by:

$$sgn(s) = \begin{Bmatrix} -1\ if\ s < 0 \\ +1\ if\ s > 0 \end{Bmatrix}$$

The reaching law approach not only establishes the reaching condition but also specifies the dynamic characteristics of the system during the reaching phase. Additional merits of this approach include simplification of the solution for VSC and providing a measure for the reduction of chattering. Hence, based on the reaching law approach $\dot{\boldsymbol{s}} = -Ksgn(\boldsymbol{s})$, the switching control law could be shown to be:

$$\boldsymbol{u} = -\left[\frac{\partial \boldsymbol{s}}{\partial \boldsymbol{x}}\boldsymbol{B}(\boldsymbol{x},t)\right]^{-1}\left\{\frac{\partial \boldsymbol{s}}{\partial \boldsymbol{x}}\boldsymbol{A} + ksgn(\boldsymbol{s})\right\} \qquad (3.15)$$

$$= -[\boldsymbol{B}(\boldsymbol{x},t)]^{-1}\left\{-\boldsymbol{x}^{(n)}{}_d + \boldsymbol{A}(\boldsymbol{x},t) + \sum_{k=1}^{n-1}\binom{n-1}{k}\boldsymbol{\lambda}^{\boldsymbol{k}}\boldsymbol{e}^{(\boldsymbol{n}-\boldsymbol{k})} + ksgns(\boldsymbol{s})\right\}$$

It could be seen from comparison of equations (3.15) with (3.8) & (3.10), that the equivalent control is a special case of the switching control with $k = 0$.

In the ideal sliding mode control, as soon as the state trajectories of the system hit the sliding surface, the state trajectories will slide along the surface. However, because of the non-idealities in the real system, a realized sliding mode control may chatter rather than sliding along the sliding surface [21].

The chattering phenomenon is generally perceived as motion which oscillates about the sliding manifold. There are two possible mechanisms which produce such a motion. First, in the absence of switching non-idealities such as delays, i.e., the switching device is switching ideally at an infinite frequency; the presence of parasitic dynamics in series with the plant causes a small amplitude high-frequency oscillation to appear in the neighborhood of the sliding manifold. These parasitic dynamics represent the fast actuator and sensor dynamics which, according to control engineering practice, are often neglected in the open-loop model used if the associated poles are well damped, and outside the desired bandwidth of the feedback control system. The interactions between the parasitic dynamics and VSC generate a nondecaying oscillatory component of finite amplitude and frequency, and this is generically referred to as chattering. Unmodeled dynamics can also be the result of model reduction employed in the design process [15].

Second, the switching non-idealities alone can cause such a high-frequency oscillation [21]. Because of various non-idealities such as hysteresis of switching operation, time delay of control system, and sampling of digital implementation, the state trajectory of a sliding mode control system does chatter rather than slide along the sliding surface. This chattering phenomenon in turn will excite high-frequency unmodelled dynamics which was not considered in control design.

Many analytical design methods were proposed to reduce the effects of chattering—for it remains to be the only obstacle for sliding mode to become one of the most significant discoveries in modern control theory [32].

Several philosophies have been used over the years to reduce or eliminate chattering; only two of the most frequently used ones will be reviewed here [15].

(1) The Continuation Approach: The ideal relay characteristic is practically impossible to implement, so one approach to reducing the chatter is to replace relay control by a saturating or a hyperbolic tangent function as shown in Figure 3.1. The saturating function is defined as [23]:

$$sat(s) = \begin{cases} +1 \; when \; s > L \\ \dfrac{s}{L} \; when \; |s| \leq L \\ -1 \; when \; s < -L \end{cases}$$

where L defines the threshold for entering the boundary layer.

The implication of Continuation Approach is that in the state space, a boundary layer around the switching surface is introduced. Within this boundary layer, the control is chosen to be a continuous approximation of the switching function. A consequence of the continuation method is that invariance is lost. The system possesses robustness that is a function of the boundary layer width. In conclusion, the continuation approach eliminates the high-frequency chattering at the price of losing the invariance property. A high degree of robustness can still be maintained within a narrow boundary layer, but significant delays in the control signal may dictate the need for a "thicker" boundary layer. The invariance and robustness properties of the system no longer exist in the latter case.

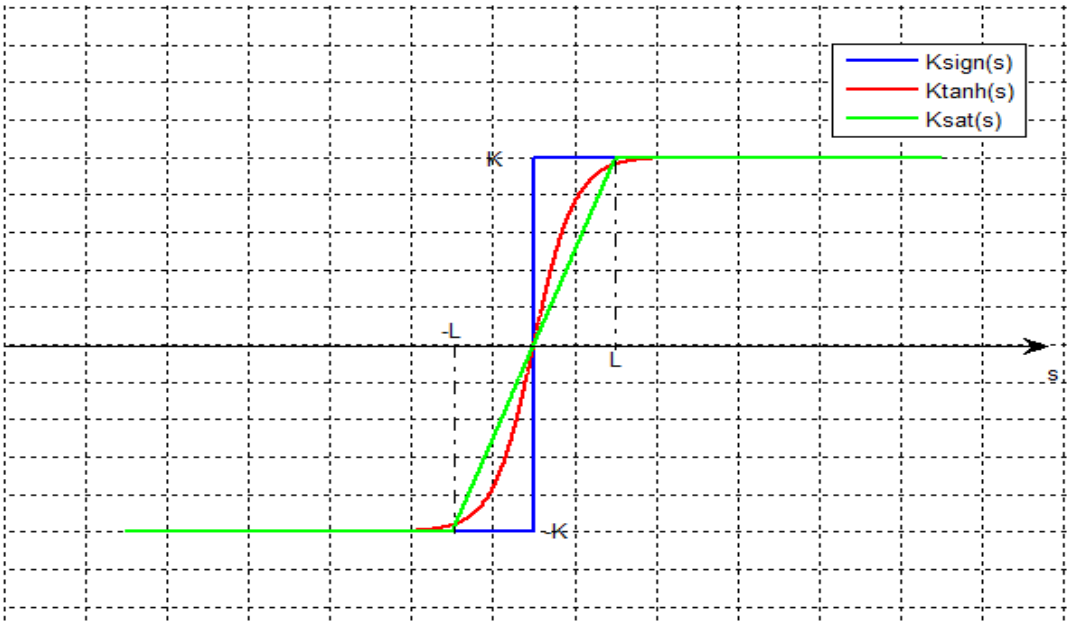


Figure 3.1: Continuous approximations for signum function

(2) Tuning the Reaching Law Approach: Chattering can be reduced by tuning parameters q_i, and k_i, in the reaching law.

$$\dot{s}_i = -k_i sgn(s_i) - q_i s_i \; i = 1, \cdots, m. \qquad (3.16)$$

Near the switching surface, $s_i \approx 0$, so $|\dot{s}_i| \approx q_i$. By choosing the gain q_i small, the momentum of the motion will be reduced as the system trajectory approaches the switching surface. As a result, the amplitude of the chatter will be reduced. However, q_i, cannot be chosen equal to zero because the reaching time would become infinite; the system fails to be a sliding mode control system. A large value of k_i, increases the reaching rate when the state is not near the switching surface.

In an effort to address the reduction of chattering in the proposed Sliding Mode Control, the second method (Tuning the Reaching Law Approach) is used. The closer the state trajectories are to the sliding surface, the smaller should be the switching gain to reduce chattering. On the other hand, the farther away the state trajectories are from the sliding surface, the larger will be the switching gain in order to reach the surface in less time. The question, then, will be: what would be the most appropriate technique that could be used in the absence of explicit

mathematical relationships between the sliding surface and the switching gain to reach the surface and remain there without excessive delay and chattering?

To this end, several techniques are applied to adaptively tune the gain of the switching function. In this thesis Fuzzy Logic is exclusively used in order to approximate the relationship between the sliding surface and the switching gain, as discussed below.

3.3. Fuzzy Logic and Genetic Algorithm [24]

The leading theory in quantifying uncertainty in scientific models from the late nineteenth century until the late twentieth century had been probability theory. However, the gradual evolution of the expression of uncertainty using probability theory was challenged first in 1937 by Max Black with his studies in vagueness, then with the introduction of fuzzy sets by Lofti Zadeh in1965.

Uncertainty can be manifested in many forms: viz. it can be fuzzy (not sharp, unclear, imprecise, or approximate), it can be vague (not specific, amorphous), it can be ambiguous (too many choices, contradictory), it can be of the form of ignorance (not knowing something), or it can be a form due to natural variability (conflicting, random, chaotic, or unpredictable).

Zadeh extended the notion of binary membership to accommodate various ‘‘degrees of membership’’ on the real continuous interval [0, 1], where the endpoints of 0 and 1 conform to no membership and full membership respectively. The sets on the universe X that can accommodate ‘‘degrees of membership’’ were termed by Zadeh as ‘‘fuzzy sets.’’ A fuzzy set $\tilde{A}$ is a function that maps its elements to a real number value on the interval [0 1]. The functional mapping is given by $\mu_{\tilde{A}}(x) \in [0\ 1]$ and the symbol $\mu_{\tilde{A}}(x)$ is the degree of membership of an element x in the fuzzy set $\tilde{A}$.Therefore, $\mu_{\tilde{A}}(x)$ is a value on the unit interval that measures the degree to which element x belongs in a fuzzy set $\tilde{A}$.

3.3.1. Fuzzification

Fuzzification is the process of making a crisp quantity ‘fuzzy’. We do this by simply recognizing that many of the quantities that we consider to be crisp and deterministic are actually not deterministic at all: they carry considerable uncertainty. If the form of uncertainty

happens to arise because of imprecision, ambiguity, or vagueness, then the variable is probably fuzzy and can be represented by a membership function.

In the real world, several devices such as a digital voltmeter generate crisp data, but these data are subject to experimental error. The information shown in Figure 3.2 shows one possible range of errors for a typical voltage reading and the associated membership function that might represent such imprecision.

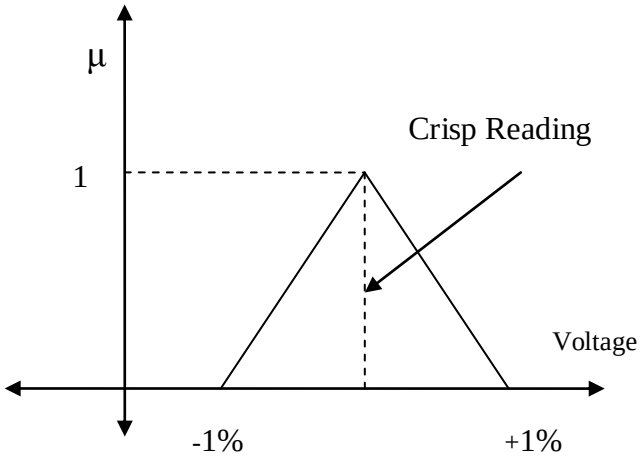


Figure 3.2: Membership function showing imprecision in 'crisp voltage reading'

Since the membership function essentially embodies all fuzziness for a particular fuzzy set, its description is the essence of a fuzzy property or operation. Just as there are an infinite number of ways to characterize fuzziness, there are an infinite number of ways to graphically depict the membership functions that describe this fuzziness. The most common forms of membership functions are those that are normal, monotonic and symmetrical.

Because of the importance of the ''shape'' of the membership function, a great deal of attention has been focused on development of these functions. Three of the six methods described in literature to assign membership values or functions to fuzzy variables are Intuition, Neural Network and Genetic Algorithm. In this thesis membership functions for the fuzzy variables are chosen based on genetic algorithm.

3.3.2. **Genetic Algorithm**

Genetic algorithms use the concept of Darwin's theory of evolution. Darwin's theory basically stressed the fact that the existence of all living things is based on the rule of ''the survival of the fittest.'' Darwin also postulated that new breeds or classes of living things come into existence through the processes of reproduction, crossover, and mutation among existing organisms. These concepts in the theory of evolution have been translated into algorithms to search for solutions to problems in a more ''natural'' way. First, different possible solutions to a problem are created. These solutions are then tested for their performance i.e., how good a solution they provide. Among all possible solutions, a portion of the good solutions is selected, and the others are eliminated i.e. the survival of the fittest. The selected solutions undergo the processes of reproduction, crossover, and mutation to create a new generation of possible solutions which are expected to perform better than the previous generation. This process of production of a new generation and its evaluation is repeated until there is convergence within a generation. The benefit of this technique is that it searches for a solution from a broad spectrum of possible solutions, rather than restricting the search to a narrow domain where the results would be normally expected. Genetic algorithms try to perform an intelligent search for a solution from a nearly infinite number of possible solutions.

All genetic algorithms contain three basic operators: reproduction, crossover, and mutation, where all three are analogous to their name sakes in genetics.

First, an initial population of n strings, for n parameters, of length L is created. The strings are created in a random fashion, i.e., the values of the parameters that are coded in the strings are random values created by randomly placing the zeros and ones in the strings. Each of the strings is decoded into a set of parameters that it represents. This set of parameters is passed through a numerical model of the problem space. The numerical model gives out a solution based on the input set of parameters. On the basis of the quality of this solution, the string is assigned a fitness value. The fitness values are determined for each string in the entire population of strings. With these fitness values, the three genetic operators are used to create a new generation of strings, which is expected to perform better than the previous generations i.e. they have better fitness values. The new set of strings is again decoded and evaluated, and

a new generation is created using the three basic genetic operators. This process is continued until convergence is achieved within a population.

Among the three genetic operators, reproduction is the process by which strings with better fitness values receive correspondingly better copies in the new generation, i.e., we try to ensure that better solutions persist and contribute to better offspring (new strings) during successive generations. This is a way of ensuring the ‘‘survival of the fittest’’ strings. The total number of strings in each generation is kept a constant for computational economy and efficiency hence; strings with lower fitness values are eliminated.

The second operator, crossover, is the process in which the strings are able to mix and match their desirable qualities in a random fashion. After reproduction, crossover proceeds in three simple steps. First, two new strings are selected at random. Second, a random location in both strings is selected. Third, the portions of the strings to the right of the randomly selected location in the two strings are exchanged. In this way information is exchanged between strings, and portions of high-quality solutions are exchanged and combined.

Reproduction and crossover together give genetic algorithm most of its searching power. The third genetic operator, mutation, helps to increase the searching power. During the creation of a generation it is possible that the entire population of strings is missing a vital bit of information that was important for determining the correct or the most nearly optimum solution. Future generations that would be created using reproduction and crossover would not be able to alleviate this problem. Here mutation becomes important. Occasionally, the value at a certain string location is changed, i.e., if there is a ‘one’ originally at a location in the bit string, it is changed to a ‘zero’, or vice versa. Thus mutation ensures that the vital bit of information is introduced into the generation. Mutation, as it does in nature, takes place very rarely.

3.3.3. **Defuzzification**

Despite the fact that the bulk of the information we assimilate every day is fuzzy, most of the actions or decisions implemented by humans or machines are crisp or binary. Therefore, in various applications and engineering scenarios, there is a need to ‘‘defuzzify’’ the fuzzy results we generate through a fuzzy systems analysis. In other words, we may eventually find

a need to convert the fuzzy results to crisp results. Defuzzification is the conversion of a fuzzy quantity to a precise quantity, just as fuzzification is the conversion of a precise quantity to a fuzzy quantity. Some of the techniques used in the defuzzification process are: Center of Largest Area, Max membership Function, Centroid Method, Weighted Average Method, Mean Max Principle, Center of Sums and First (or last) of Maxima. However, only one of these methods shall be used and discussed next.

Centroid Method: This procedure (also called center of area, center of gravity) is the most prevalent and physically appealing of all the defuzzification methods; it is given by the algebraic expression:

$$x^* = \frac{\int \mu_{\tilde{A}}(x).xdx}{\int \mu_{\tilde{A}}(x)dx} \qquad (3.17)$$

where, x^* denotes the defuzzified value and $\mu_{\tilde{A}}(x)$ is the membership of x in fuzzy set $\check{A}$.

3.3.4. **Principles of Fuzzy Control Design**

A number of assumptions are implicit in a fuzzy control system design. Six basic assumptions are commonly made whenever a fuzzy rule-based control policy is selected.

i. The plant is observable and controllable: state, input, and output variables are usually available for observation and measurement or computation.
ii. There exists a body of knowledge comprised of a set of linguistic rules, engineering common sense, intuition, or a set of input–output measurements data from which rules can be extracted.
iii. A solution exists.
iv. The control engineer is looking for a ‘‘good enough’’ solution, not necessarily the optimum one.
v. The controller will be designed within an acceptable range of precision.
vi. The problems of stability and optimality are not addressed explicitly; such issues are still open problems in fuzzy controller design.

The fuzzy controller used in this thesis can be depicted by a block diagram shown in Figure 3.3. The knowledge base module in the figure contains knowledge about all the input and

output fuzzy partitions. It includes the membership functions defining the input variables to the fuzzy rule base system and the output variables, or control actions, to the plant under control. The rule base contains rules in an antecedent-consequent form that sets the foundation for approximate (imprecise) reasoning. An expression of a typical rule has the format as given in equation (3.18).

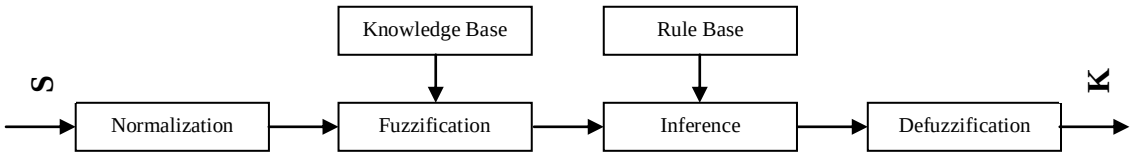


Figure 3.3: Block diagram of Fuzzy Logic Control system

$$IF\ x\ is\ \tilde{A}, THEN\ y\ is\ \tilde{B} \qquad (3.18)$$

where $x\ and\ y$ are crisp quantieis, $\tilde{A}\ and\ \tilde{B}$ denote fuzzy sets

The steps in designing a simple fuzzy control system are as follows:

i. Identify the variables (inputs, states, and outputs) of the plant.
ii. Partition the universe of discourse or the interval spanned by each variable into a number of fuzzy subsets, assigning each a linguistic label.
iii. Assign or determine a membership function for each fuzzy subset
iv. Assign the fuzzy relationships between the inputs' or states' fuzzy subsets on the one hand and the outputs 'fuzzy subsets on the other hand, thus forming the rule-base.
v. Choose appropriate scaling factors for the input and output variables in order to normalize the variables to the [0, 1] or the [−1, 1] interval.
vi. Fuzzify the inputs to the controller.
vii. Use fuzzy approximate reasoning to infer the output contributed from each rule.
viii. Aggregate the fuzzy outputs recommended by each rule.
ix. Apply defuzzification to form a crisp output

3.4. **Linear Algebraic Method** [25]

In the Linear Algebraic method or the inward approach, we first find an overall transfer function of the system to meet design specifications and then implement it using suitable configuration. This is contrary to the root-locus or frequency-domain methods where one must first choose a configuration and a compensator with open parameters. The analyst then searches for parameters such that the resulting overall system will meet design specifications. This approach is essentially a trial-and-error method; therefore, one usually chooses the simplest possible feedback configuration namely, a unity feedback configuration and start from the simplest compensator namely, a gain. If the design objective cannot be met by searching the gain, the designer then chooses a different configuration or a compensator of degree 1 i.e. phase-lead or phase-lag network and repeat the search. This approach is called the outward approach as the design starts from the internal compensators and then finds an overall system to meet design specifications. In the inward approach, however, the designer first searches an overall transfer function to meet design specifications, and then chooses a configuration and computes the required compensators.

The choice of an overall transfer function is not entirely arbitrary. Given a proper plant transfer function:

$$G(s) = \frac{N(s)}{D(s)} \qquad (3.19)$$

An overall transfer function $G_o(s) = \frac{N_o(s)}{D_o(s)}$ is said to be implementable if there exists a configuration with no plant leakage such that $G_o(s)$ can be built using only proper compensators. The "no plant leakage" constraint implies that all power must pass through the plant and that no compensator be introduced in parallel with the plant. Furthermore, the resulting system is required to be well posed and totally stable. By well-posed we mean, the closed-loop transfer function of every possible input-output pair of the system is proper (to avoid amplification of high frequency noise. The necessary and sufficient conditions for $G_o(s)$ to be implementable are that

(1) $G_o(s)$ is stable

(2) $G_o(s)$ contains the non-minimum-phase zeros of $G(s)$

(3) The pole-zero excess of $G_o(s)$ is equal to or larger than that of $G(s)$.

The poles of $G(s)$ or the roots of $D(s)$ are shifted by feedback and it is immaterial whether $D(s)$ is Hurwitz or not. However, if a root of $N(s)$ does not appear in $N_o(s)$, the only way to achieve this is to introduce the same root by feedback to cancel it. This cancellation is unstable pole-zero cancellation if the root of $N(s)$ is in the closed right half s-plane. Therefore, all the non-minimum-phase zeros of $G(s)$ must appear in $N_o(s)$. If the pole-zero excess inequality is met, all poles and all minimum-phase zeros of $G_o(s)$ can be arbitrarily assigned. The overall implementable transfer function $G_o(s)$ shall be chosen to fulfill both the steady state and transient behaviors of the system, $G(s)$.

3.4.1. **Transient and Steady State Requirements**

Transient requirement

The ITAE has the best selectivity-the largest changes as ξ varies compared to IAE and ISE performance indices. In addition the ITAE also yields a system with a faster response than the other criteria. In this thesis, an implementable overall system is chosen to minimize the ITAE performances index.

A constraint on the actuating signal or on the bandwidth of the resulting systems must be imposed; otherwise, it is possible to design an overall system to have a performance index as small as desirable and the corresponding actuating signal will approach infinity. Therefore, the problem will be to find an optimal implementable $G_o(s)$ with desired tracking performances as in equation (3.22) so as to:

$$Minimize\ J \coloneqq \int_0^\infty t|e(t)|dt \qquad (3.20)$$

$$subject\ to\ u(t) < M, \forall\ t$$

For plants with minimum-phase zeros, standard forms for ITAE optimal systems were found through analog computer simulations and tabulations of these results are available in several control text-books.

Steady state requirement

The control system with the overall transfer function must be designed so that the plant output $y(t)$ eventually tracks a reference input $r(t)$ without error. That is, if the chosen transfer function $G_o(s)$ given by:

$$G_o(s) = \frac{\beta_o + \beta_1 s + \beta_2 s^2 + \cdots + \beta_m s^m}{\alpha_o + \alpha_1 s + \alpha_2 s^2 + \cdots + \alpha_n s^n} \quad (3.21)$$

with α_n>0 and $n \geq m$ is said to achieve asymptotic tracking if

$$\lim_{t\to\infty} |y(t) - r(t)| = 0 \quad (3.22)$$

We require $G_o(s)$ to be stable, which in turn requires that α_i>0, $\forall\, i$. The condition for $G_o(s)$ to achieve asymptotic tracking depends on the type of $r(t)$ available at the input port, for instance if $r(t)$ is a ramp function, the conditions are $G_o(s)$ be stable, and $\alpha_0 = \beta_0$, $\alpha_1 = \beta_1$.

3.4.2. **Implementation by Two-Parameter Configuration**

In this section, given a plant transfer equation and an implementable model $G_o(s)$, we design an overall system so that the overall transfer function matches or equals $G_o(s)$.

If the denominator $D(s)$ and the numerator $N(s)$of a plant transfer function have no common factors, then it is possible to achieve any model matching. In this case, the conditions of achieving matching depend on the degree of the compensators. The larger the degree of the compensator, the more parameters we have for matching.

Here, the two-parameter configuration is used to implement the overall transfer function $G_o(s)$ and to calculate the desired compensators. Let us consider the unity feedback system shown in Figure 3.4.

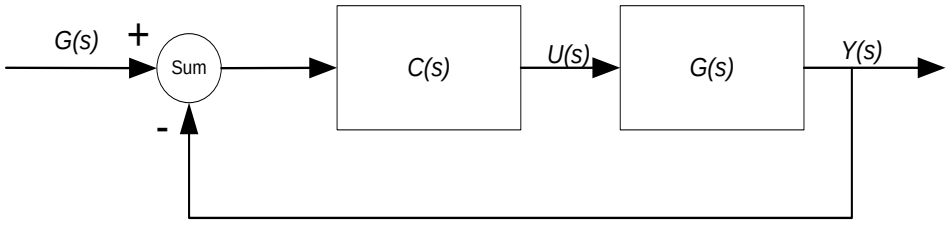


Figure 3.4: A unity feedback system to design compensator by the inward approach

The actuating signal is of the form

$$\begin{aligned} U(s) &= C(s)\big(R(s) - Y(s)\big) \\ &= C(s)R(s) - C(s)Y(s) \end{aligned} \quad (3.23)$$

Now, we look for the most generic form of compensator by introducing $C_1(s) = \frac{L(s)}{A_1(s)}$ as the feedforward compensator and $C_2(s) = \frac{M(s)}{A_2(s)}$ as feedback compensator, where $L(s), M(s), A_1(s)$ and $A_2(s)$ are polynomials. In fact $A_1(s)$ and $A_2(s)$ need not be the same. It turns out that even if they are chosen to be the same, the two compensators can be used to achieve any model matching. Therefore, we assume that $A_1(s) = A_2(s) = A(s)$ and the compensators become:-

$$\begin{aligned} U(s) &= C_1(s)R(s) - C_2(s)Y(s) \\ &= \frac{L(s)}{A(s)}R(s) - \frac{M(s)}{A(s)}Y(s) \\ &= \left[\frac{L(s)}{A(s)} - \frac{M(s)}{A(s)}\right]\begin{bmatrix} R(s) \\ Y(s) \end{bmatrix} \end{aligned} \quad (3.24)$$

Thus the compensator is written as,

$$\begin{aligned} C(s) := [C_1(s) - C_2(s)] &= \left[\frac{L(s)}{A(s)} - \frac{M(s)}{A(s)}\right] \\ &= A^{-1}[L(s) - M(s)] \end{aligned} \quad (3.25)$$

The compensator shall be implemented as a unit to avoid possible unstable pole-zero cancellation; furthermore the following configuration that requires the minimum number of integrators is used to design the controller.

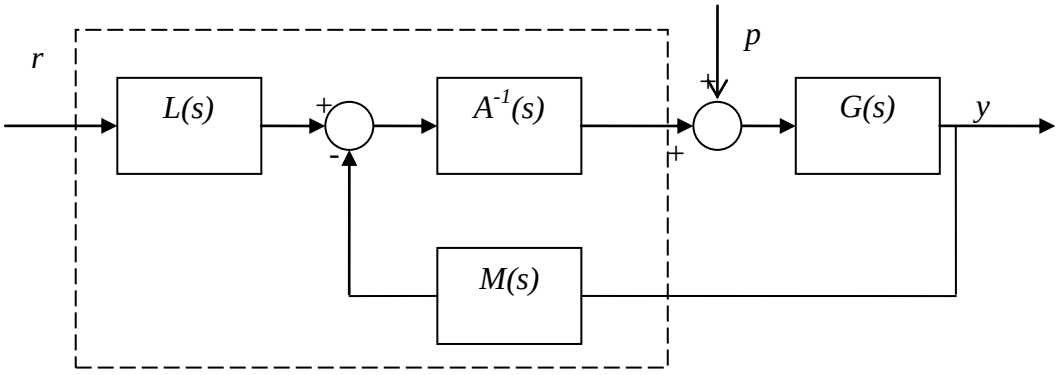


Figure 3.5: The Two-parameter feedback configuration for compensator design

Using Mason's gain formula,

$$\frac{Y(s)}{R(s)} = \frac{L(s)A^{-1}(s)G(s)}{1 + A^{-1}(s)M(s)G(s)} \quad (3.26)$$
$$= \frac{L(s)N(s)}{A(s)D(s) + M(s)N(s)}$$

The problem of finding the appropriate compensators can be stated as:

If $N(s)$ *and* $D(s)$ *are coprime and* $deg\big(N(s)\big) < deg\big(D(s)\big) = n$ *and given an implementable* $G_o(s) = \frac{N_o(s)}{D_o(s)}$, *find proper compensators* $\frac{L(s)}{A(s)}$ *and* $\frac{M(s)}{A(s)}$ *such that* $G_o(s) = \frac{N_o(s)}{D_o(s)} = \frac{L(s)N(s)}{A(s)D(s)+M(s)N(s)}$

The solution proceeds in the following three steps:

Step 1: compute

$$\frac{G_o(s)}{N(s)} = \frac{N_o(s)}{D_o(s)} = \frac{L(s)N(s)}{A(s)D(s) + M(s)N(s)} := \frac{N_p(s)}{D_p(s)} \quad (3.27)$$

where $N_p(s)$ and $D_p(s)$ are coprime.

Step 2: introduce an arbitrary Hurwitz polynomial $\bar{D}_p(s)$ so that the degree of $D_p(s)\ \bar{D}_p(s)$ is at least $2n - 1$.

Step 3: rewrite equation (3.27) as:

$$G_o(s) = \frac{N(s)N_p(s)}{D_p(s)} \quad (3.28)$$
$$= \frac{N(s)\big[N_p(s)\bar{D}_p(s)\big]}{D_p(s)\bar{D}_p(s)}$$
$$= \frac{L(s)N(s)}{A(s)D(s) + M(s)N(s)}$$

Now we set

$$L(s) = N_p(s)\bar{D}_p(s) \quad (3.29)$$
$$A(s)D(s) + M(s)N(s) = D_p(s)\bar{D}_p(s)$$
$$:= F(s)$$

If we write $A(s), M(s)$and $F(s)$ as polynomials:

$$A(s) = A_o + A_1(s) + \cdots + A_m s^m \quad (3.30)$$
$$M(s) = M_o + M_1(s) + \cdots + M_m s^m$$
$$F(s) = F_o + F_1(s) + \cdots + F_{n+m} s^{n+m}$$

where $m \geq n - 1$, then $A(s) and\ M(s)$ can be solved from equation (3.31).

$$\left(\begin{array}{cc|cc|c|cc} D_o & N_o & 0 & 0 & \cdots & 0 & 0 \\ D_1 & N_1 & D_o & N_o & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ D_n & N_n & D_{n-1} & N_{n-1} & \cdots & D_0 & N_0 \\ 0 & 0 & D_n & N_n & \cdots & D_1 & N_1 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & D_n & N_n \end{array}\right)\left(\begin{array}{c} A_o \\ \underline{M_o} \\ A_1 \\ \underline{M_1} \\ \vdots \\ A_m \\ M_m \end{array}\right) = \begin{pmatrix} F_o \\ F_1 \\ F_2 \\ \vdots \\ F_{n+m} \end{pmatrix} \quad (3.31)$$

It was shown that [25] the resulting compensators are always proper and the two-parameter configuration is stable and well-posed with no "plant leakage", implying that the two-parameter configuration can be used to implement any implementable overall transfer function.

3.5. VRML AS A 3-D MODELING TOOL [26]

The Virtual Reality Modeling Language (VRML) is a file format for describing interactive 3-D objects and worlds. VRML is designed to be used on the Internet, intranets, and local client systems. VRML is also intended to be a universal interchange format for integrated 3-D graphics and multimedia. VRML may be used in a variety of application areas such as engineering and scientific visualization, multimedia presentations, entertainment and educational titles, web pages, and shared virtual worlds.

In VRML, a 3-D scene is described by a hierarchical tree structure of objects (nodes). Every node in the tree represents some functionality of the scene. There are 54 different types of nodes. Some of them are *shape nodes* (representing real 3-D objects), and some of them are *grouping nodes* used for holding child nodes. Here are some examples:

- Box node — Represents a box in a scene.
- Transform node — Defines position, scale, scale orientation, rotation, translation, and children of its subtree (grouping node).
- Material node — Corresponds to material in a scene.
- DirectionalLight node — Represents lighting in a scene.

Each node contains a list of fields that hold values defining parameters for its function. Nodes can be placed in the top level of a tree or as children of other nodes in the tree hierarchy.

When a value is changed in the field of a certain node, all nodes in its sub-tree are affected. This feature allows one to define relative positions inside complicated compound objects.

In the hierarchical structure of a VRML file, the position and orientation of child objects are specified relative to the parent object. The parent object has its local coordinate space defined by its own position and orientation. Moving the parent object also moves the child objects relative to the parent object.

Regarding measurement units used in VRML All lengths and distances are measured in *meters*, and all angles are measured in *radians*.

Before we conclude this section, it is worth mentioning the conventions of the coordinate systems used in VRML and MATLAB. VRML uses the *world coordinate system* in which the *y*-axis points upward and the *z*-axis places objects nearer or farther from the front of the screen. In VRML, rotation angles are defined using the right-hand rule. Imagine your right hand holding an axis while your thumb points in the direction of the axis toward its positive end. Your four remaining fingers point in a counterclockwise direction. This counterclockwise direction is the positive rotation angle of an object moving around that axis.

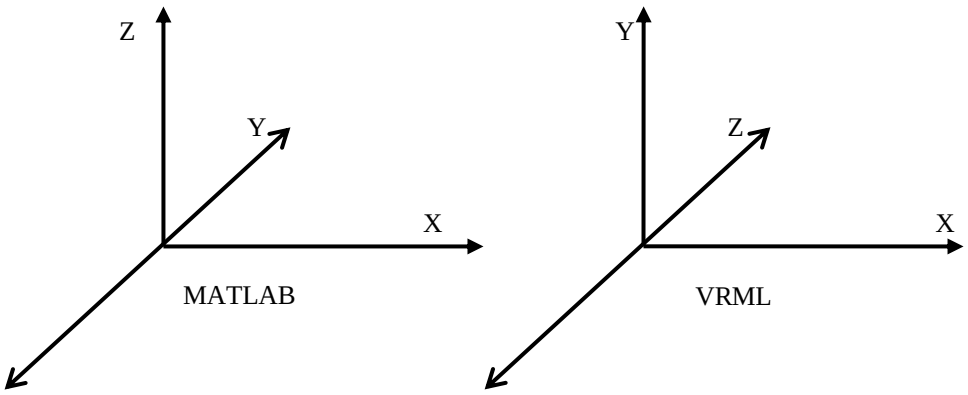


Figure 3.6: Comparison of MATLAB and VRML coordinate systems

3.6. SIMULINK® 3D ANIMATION [27]

The Simulink®3D Animation™ product uses VRML97 technology to deliver a 3-D visualization solution for MATLAB users. The Simulink®3D Animation™ product provides an interface to connect an existing virtual world, defined with VRML, to the Simulink® and MATLAB® environments. It is a useful contribution to a wide use of VRML97 in the field of technical and scientific computation and interactive 3-D animation.

The Simulink®3D Animation™ product is a solution for interacting with virtual reality models of dynamic systems over time. It extends the capabilities of the MATLAB and Simulink software into the world of virtual reality graphics. The necessary steps to use the product are:

Step 1: Create virtual worlds or three-dimensional scenes using standard Virtual Reality Modeling Language (VRML).

Step 2: Create and define dynamic systems with the MATLAB and Simulink products.

Step 3: View moving three-dimensional scenes driven by signals from the Simulink environment.

The Simulink®3D Animation™ library provides blocks to directly connect Simulink signals with virtual worlds. This connection lets us to visualize the Simulink model as a three-dimensional animation. We can implement most of the software features with Simulink blocks. Once these blocks are included in a Simulink diagram, we can select a virtual world and connect Simulink signals to the virtual world. The software automatically scans a virtual world for available VRML nodes that the Simulink software can drive. All the VRML node properties are listed in a hierarchical tree-style viewer. Then, we select the degrees of freedom to control from within the Simulink interface. After connecting these inputs to appropriate Simulink signals, we can view the simulation with a VRML viewer.

The Simulink®3D Animation™ product contains a viewer that is the default viewing method for virtual worlds. On the other hand, to create the virtual worlds that are desired to be

connected to Simulink block diagrams, we use a VRML editor. For this purpose, the Simulink®3D Animation™ product includes one of the classic VRML authoring tools, V-Realm Builder by Ligos. With the addition of this VRML authoring tool, the product provides a complete authoring, development, and working environment for carrying out 3-D visual simulations.

CHAPTER 4
CONTROLLER DESIGN

4.1. INTRODUCTION

One of the most effective control schemes for the Ball and Plate system is the double feedback loop structure, a loop within a loop [1]. The inner loop serves as an actuator angular position controller for the plate inclination; while the outer loop controls the ball's linear position on the plate. While the controller in the outer loop computes the angle by which the plate should move to balance the ball, the inner loop controller actually moves the plate by that angle [1].

In practical application, nonlinear control methods must be adopted in the outer control loop in order to get high controller performance. High performance is required because of the existence of uncertainty caused by frictions, backlash effect in transmission, measurement time delays and parameter uncertainties [1]. In order to successfully implement the design, the inner loop must have a much faster response than the outer loop. Since the inner loop can only be designed to be as fast as physical limitations of motor maximum torque, speed, etc. allow, we must intentionally design the outer loop to be slower [8].

4.2. PARAMETER SELECTION OF THE ACTUATOR

Let us consider Figure 4.1 that shows an actuator system with permanent magnet DC motor and the gear. For many applications gears exhibit backslash because of the loose fit between two or more meshed gears. We shall, however, idealize the behavior of gears and assume that there is no backslash [27].

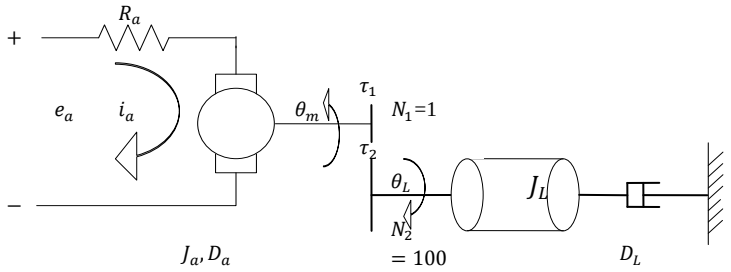


Figure 4.1: An actuator system with the DC motor and the gear

To find the transfer function between the angle θ_L and the voltage e_a, we shall first find the relationship between the input torque τ_1 and the delivered torque τ_2 and the electrical equations as given in equation (4.1) [27]

$$\left\{\begin{array}{c} \tau_m = J_a \dfrac{d^2\theta_m}{dt^2} + f_1 \dfrac{d\theta_m}{dt} + \tau_1 \\ \tau_2 = J_L \dfrac{d^2\theta_L}{dt^2} + f_2 \dfrac{d\theta_L}{dt} \\ R_a I_a + L_a I_a s = e_a - K_e \theta_m s \\ \tau_m = k_m I_a \end{array}\right\} \quad (4.1)$$

Solving equation (4.1) for the relationship between θ_L and e_a we get:

$$\frac{\theta_L}{e_a} = \frac{K_t \dfrac{N_1}{N_2}}{\left[J_{eq} L_a s^3 + \left(J_{eq} R_a + D_{eq} L_a\right) s^2 + \left(D_{eq} R_a + K_t K_e\right) s\right]} = T \quad (4.2)$$

$$where\ J_{eq} = J_a + J_L \left(\frac{N_1}{N_2}\right)^2\ and\ \ D_{eq} = D_a + D_L \left(\frac{N_1}{N_2}\right)^2$$

The block diagram representation of the motor (and the load) together with the gear is presented in Figure 4.2.

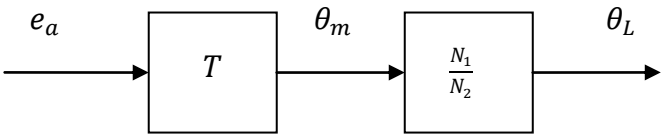


Figure 4.2: Representation of the transfer function between load angle (angle at secondary gear) and armature voltage.

The motor parameters are selected based on the requirement of the load torque, moment of inertia and angular speed. The following parameters relating to the Ball and the Plate are obtained from HUMUSOFT CE151-Ball & Plate Apparatus [13].

Table 4.1: Parameters of the Ball and Plate system

No.	Description	parameter	Value	Unit
1	Mass of the ball	m	0.11	Kg
2	Radius of the ball	R	0.02	m
3	Plate dimension (square)	$l\ x\ w$	1	m^2
4	Moment of Inertia of the plate	$J_{P_{x,y}}$	0.5	Kgm^2
5	Moment of Inertia of the ball	J_B	$1.76x10^{-5}$	Kgm^2
6	Maximum Speed	v	4	mm/s

Noting that the angular acceleration, angular speed and linear speed are all small; the load torque could be obtained from equation (2.15) and the values of the B&P system parameters given in Table 4.1 as follows:

$$\tau_L = \tau_{\theta_x} = \left(mx^2 + J_B + J_{P_x}\right)\ddot{\theta}_x + 2mx\dot{x}\dot{\theta}_x + mxy\ddot{\theta}_y + m(\dot{x}y + x\dot{y})\dot{\theta}_y - mgx\cos\theta_x \cong mgx\cos\theta_x = 0.5396Nm \quad (4.3)$$

The required angular speed is estimated from animated videos using VRML for both trajectory tracking and stabilization problems. It was found that $n_L = 20\, rev/min$ is the desired actuator speed for our application.

On the other hand, the moment of inertia of the load is obtained from Table 4.1 as:

$$J_L = J_P + J_B \approx 0.5 kgm^2 \qquad (4.4)$$

Based on the values of τ_L, J_L and n_L, we choose DC motors which fulfill these minimal requirements from a certain manufacturer's catalogue (Harmonic Drives LLC, in this case) and the motors shall be examined in detail through further analysis.

Table 4.2: Summary of actuator parameter after preliminary and detailed analysis

Type	J_a(Kgm^2)	τ_{ave}(Nm)	τ_N (Nm)	τ_{acc}(Nm)	τ_{max}(Nm)	$n_L(rev/min)$
RHS-14 3003	1.8	27.63	7.8	60.75	28	30
RHS-20 3007	1.2	20.42	24	45.05	84	30
RFS-20 3012	1.9	28.83	30	63.37	84	30
RHS-25 6012	0.53	12.38	20	27.5	100	60
RHS-25 3012	2.1	31.23	40	68.6	160	30
RFS-25 6018	1.1	19.22	30	42.43	100	60
RFS-25 3018	4.5	60.06	60	131.44	160	30
RHS-32 6018	1.4	22.83	30	50.28	220	60
RHS-32 3018	5.8	75.68	60	165.47	340	30
RFS-32 6030	3.1	43.25	50	94.79	220	60
RFS-32 3030	12.0	150.15	100	327.79	340	30

Assuming, a 40% duty cycle operation, with the duration of each time interval chosen as $t_1 = 0.1sec, t_2 = 0.2sec, t_3 = 0.1sec\ and\ t_4 = 0.6sec$, we calculate the acceleration torque $\tau_1$and RMS torque τ_{ave}

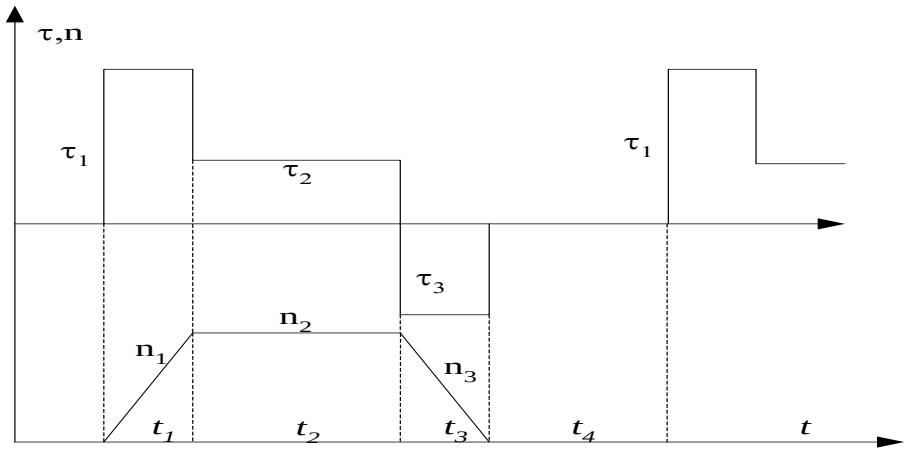


Figure 4.3: Schematic of torque and speed evolution with time

and select those actuators that satisfy $\tau_1 < \tau_{max}$and $\tau_{ave} < \tau_N$, where:

$$\tau_1 = \tau_L + 2\pi \frac{(J_A + J_L)n_L}{t_1} \quad (4.5)$$

$$\tau_{ave} = \sqrt{\frac{{\tau_1}^2 t_1 + {\tau_2}^2 t_2 + {\tau_3}^2 t_3}{t_1 + t_2 + t_3 + t_4}} \quad (4.6)$$

Based on the above analysis, the actuator chosen is RHS-25 3012. To function with accurate position and speed control, the DC servo actuators are equipped with control units. Safety features are incorporated to protect the control units and the motor from accidental damage. The nature of the input signal could either be analog or digital (from a programmable controller) or data communication input. The data communication input control unit uses RS-232C interface for connection of the unit to an external computer where the controller codes are found.

Ball position data is available through an absolute encoder with a resolution of 1024 ppr (at the motor shaft) or 102400 ppr at the output of the actuator. We summarize the important parameters for the RHS-25 3012 actuator in Table 4.3. The overall configuration with all the constituting parts is given in Appendix B.

Table 4.3: Actuator parameters

No.	Item	Parameter	Unit	Value
1	Rated output power	P_{rated}	W	123
2	Rated voltage	V_{rated}	V	75
3	Rated current	I_{rated}	A	1.9
4	Rated output torque	τ_{rated}	Nm	40
5	Rated output speed	ω_{rated}	rpm	30
6	Maximum output torque	τ_{max}	Nm	160
7	Maximum output speed	ω_{max}	rpm	40
8	Torque constant	K_t	Nm/A	21
9	Inertia at the output shaft	J_a	Kgm^2	2.1
10	Viscous damping constant	D_m	Nm/rpm	0.52
11	Gear ratio	1:R	-	100
12	Armature resistance	R	Ω	1.2
13	Armature inductance	L	mH	1.1
14	Weight (gearbox, motor, encoder)	W	kg	5.7
15	Dimension	$l\ x\ \Phi$	mmXmm	309xΦ130

Since the inductance is very small, its effect is neglected [28] in the mathematical model of the motor. Hence, equation (4.2) simplifies to:

$$\frac{\theta_L}{e_a} = \frac{K_t \frac{N_1}{N_2}}{[(J_m R_a)s^2 + (D_m R_a + K_t K_e)s]} \quad (4.7)$$

Noting that $J_{eq} = J_a + J_L \left(\frac{N_1}{N_2}\right)^2 = 2.1\text{x}10^{-4} + 0.5\text{x}10^{-4}$ and $D_{eq} = 4.9656\text{x}10^{-4}$ and substituting the values for R_a (armature resistance), k_t (torque constant) and k_e(viscous damping constant) from Table 4.3 into equation (4.31) we obtain:

$$\frac{\theta_L}{e_a} = \frac{21x10^{-2}}{[(3.12x10^{-4})s^2 + (431.13)s]} \quad (4.8)$$

$$= \frac{6.73x10^2}{s(s + 138.183x10^4)} \approx \frac{4.87x10^{-4}}{s}$$

The effect of the pole at $s = -138.183x10^4$ on the transient behavior is negligible and therefore the transfer function is approximated by a first order differential equation. Though, we can safely neglect the impact of the pole, we must retain its effect on the steady state performance of the system [28].

Under the assumption that there is little coupling between the ball movement along the two orthogonal directions, controller design for the B&P system could be divided into two independent problems. As the two problems share the same structure, only one direction will be introduced next.

4.3. DESIGN OF INNER CONTROL LOOP

The double feedback loop structure, a loop within a loop is one of the most effective control schemes for the Ball and Plate system [1]. The inner loop serves as an angular position controller for the plate inclination; while the outer loop controls the ball's linear position on the plate. The overall block diagram representation of the system is shown in Figure 4.4.

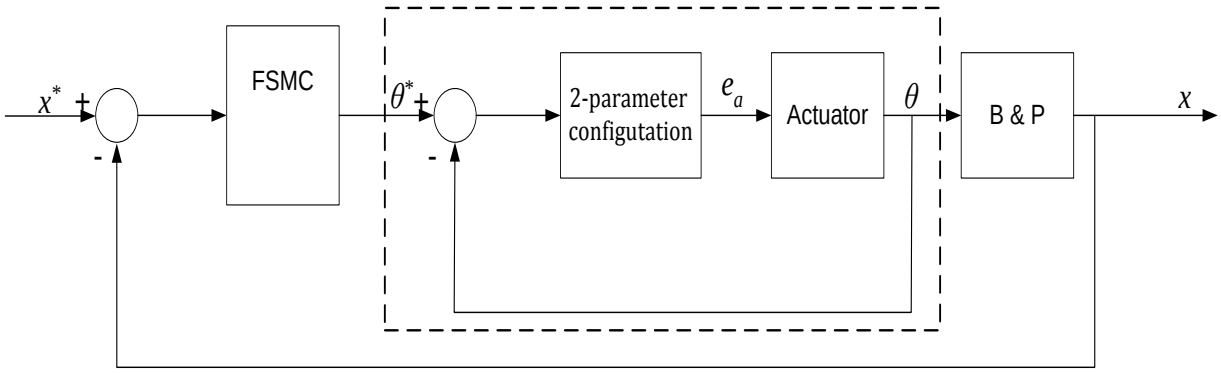


Figure 4.4: The proposed double feedback loop for the B&P system

Using the simplified actuator transfer function as given in equation (4.8) and the controller form as illustrated in Figure 3.5, the inner control loop is shown in Figure 4.5. We note that $r = \theta^*$ and $y = \theta$).

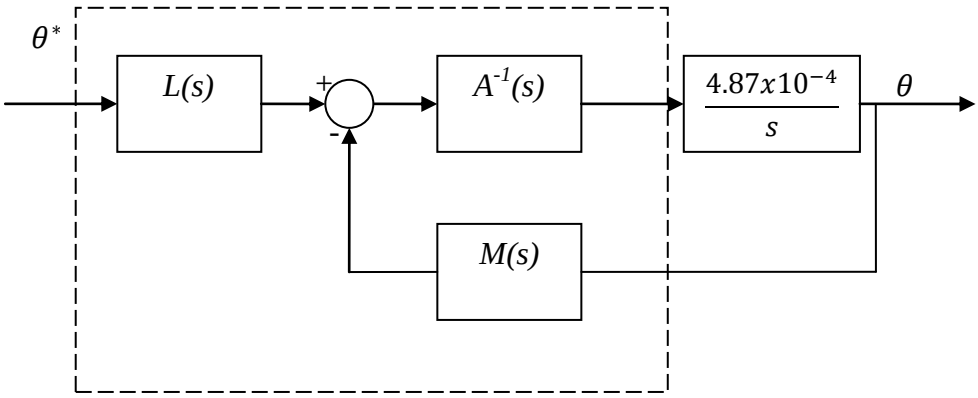


Figure 4.5: Implementation of 2-parameter configuration for the inner control loop

We choose, $G_o(s)$, the ITAE optimal overall transfer function with zero position error as [25]:

$$G_o(s) = \frac{{\omega_o}^2}{s^2 + 1.4\omega_0 s + {\omega_o}^2} \qquad (4.9)$$

In order to determine the value of ω_o, we require that a response due to a step input should settle in $0.4\ sec$. Through iterative simulation, we find that the value $\omega_o = 20\ rad/sec$ gives the desired steady state response.

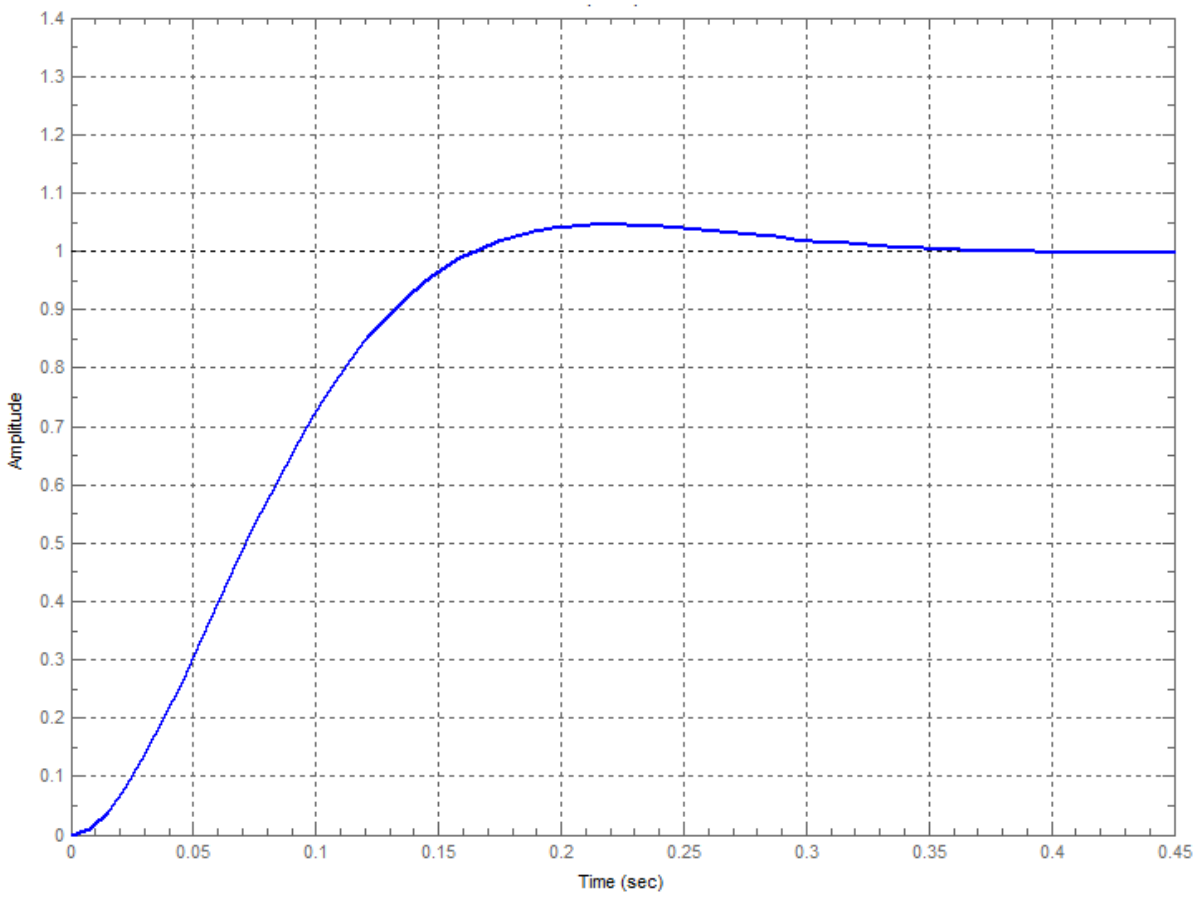


Figure 4.6: Step response of the overall transfer function $G_o(s)$

For the given value of ω_o, $G_o(s)$ becomes:

$$G_o(s) = \frac{400}{s^2 + 28s + 400} \qquad (4.10)$$

and the step response of the actuating signal is obtained as follows:

$$G_o(s) = \frac{y}{r} = \frac{L(s)A(s)^{-1}G(s)}{1 + A(s)^{-1}G(s)M(s)} = \frac{L(s)G(s)}{A(s) + G(s)M(s)} \qquad (4.11)$$

$$\frac{U(s)}{R(s)} = \frac{L(s)A(s)^{-1}}{1 + A(s)^{-1}G(s)M(s)} = \frac{L(s)}{A(s) + G(s)M(s)}$$

$$\frac{U(s)}{R(s)} = \frac{y}{r} * \frac{1}{G(s)} = \frac{G_o(s)}{G(s)}$$

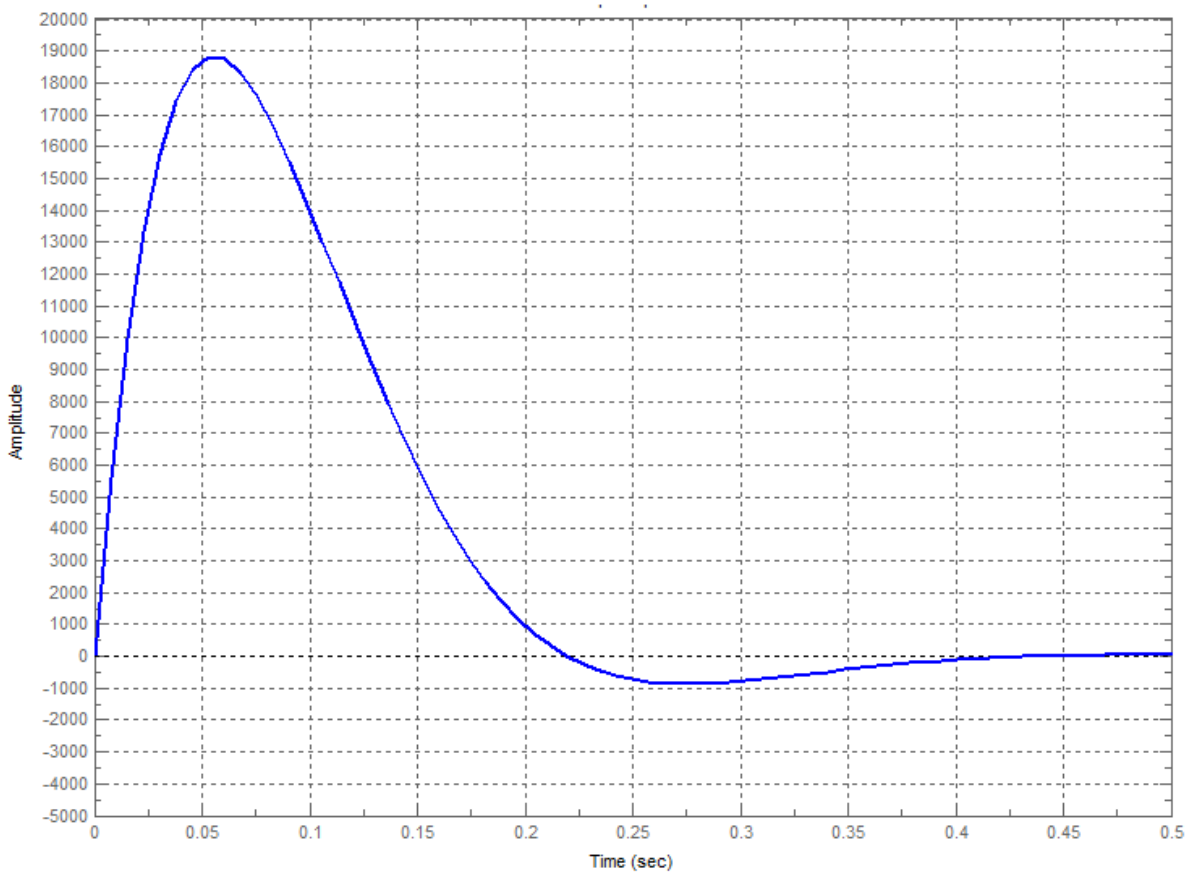


Figure 4.7: The response of the actuating signal due to a step input

The maximum value of the step response is found to be approximately 18804.08v!, which is unrealistic. Therefore, we demand that the actuating signal $U(s)$ due to a step input cannot exceed the rated motor voltage. To fulfill this constraint, we rewrite the actuator model with open parameters as $\frac{k_m}{s}$ and search the value of k_m through computer simulation in doing so we use the following condition:

$$step\left(\frac{G_o(s)}{G(s)}\right) < 75 \qquad (4.12)$$

$$step\left(\frac{400}{s^2+28s+400} * \frac{s}{k_m}\right) < 75$$

With the value of $k_m = 0.123$, we obtain $U(s) = 74.4519v$. Since the actuating system could only provide $4.87x10^{-4}$, the required gain $\frac{0.123}{4.87x10^{-4}} \approx 253$ should be obtained using a

preamplifier. However, this will not affect the fact that the inner loop settles in $0.4\,sec$. The step response of the actuating signal with the preamplifier is given in Figure 4.8.

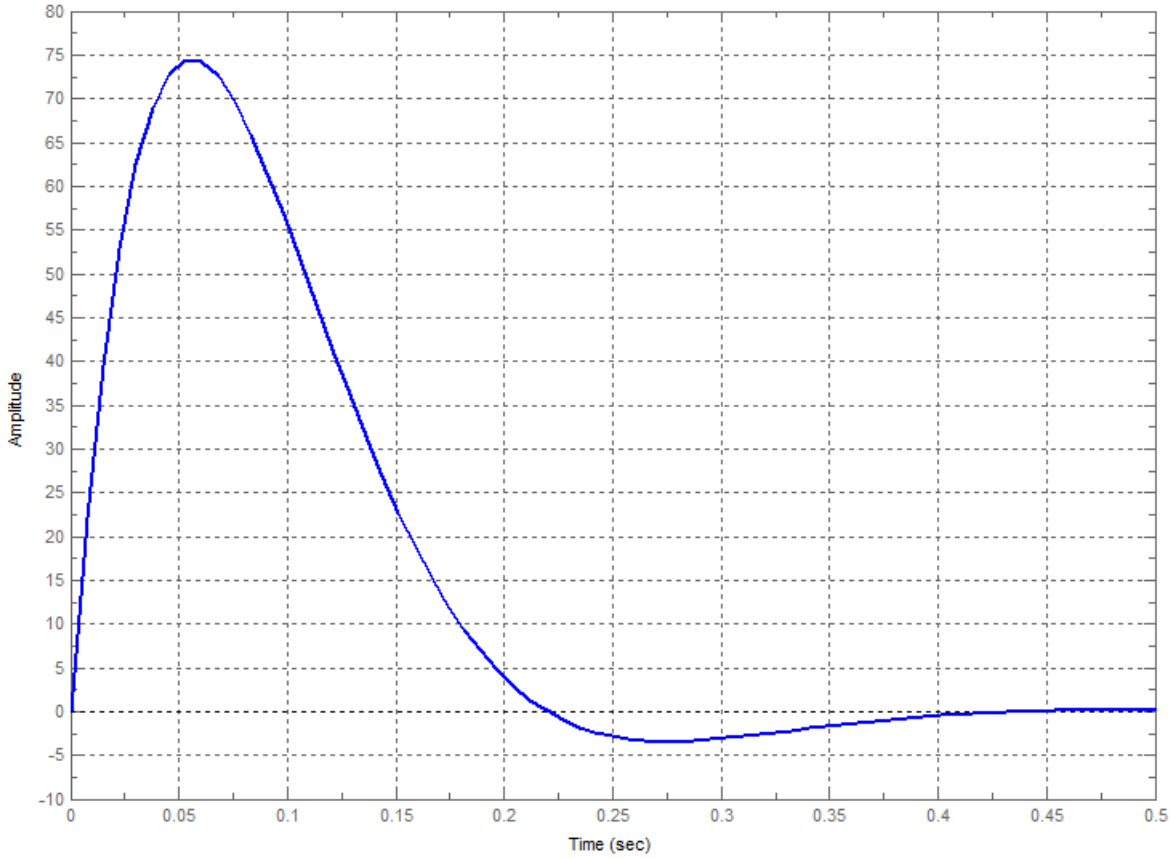


Figure 4.8: The step response of the actuating signal with the preamplifier

The overall transfer function is given as:

$$G_o(s) = \frac{400}{s^2 + 28s + 400}, G(s) = \frac{0.123}{s} = \frac{N(s)}{D(s)} \qquad (4.13)$$

Dividing $G_o(s)$ by $N(s)$, we get:

$$\frac{G_o(s)}{N(s)} = \frac{3252}{s^2 + 28s + 400} = \frac{N_p(s)}{D_p(s)} \qquad (4.14)$$

We introduce a Hurwitz polynomial $\bar{D}_p(s)$ with its roots inside acceptable pole-zero cancellation region so that the $\deg\left(D_p(s)\bar{D}_p(s)\right) \geq 2n - 1 = 1$. We note that the degree of $D_p(s)$ is already 2. Therefore, we choose $\bar{D}_p(s) = 1$.

Using equation (3.29),

$$L(s) = N_p(s)\bar{D}_p(s) = 3252 \qquad (4.15)$$

$$A(s)D(s) + M(s)N(s) = s^2 + 28s + 400$$

We next solve the resulting Diophantine equation for the values of the compensator from equation (4.15) and the actuator model in equation (4.13). Thus, we have:

$$\begin{bmatrix} 0 & 0.123 & 0 & 0 \\ 1 & 0 & 0 & 0.123 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} A_o \\ M_o \\ A_1 \\ M_1 \end{bmatrix} = \begin{bmatrix} 400 \\ 28 \\ 1 \end{bmatrix} \qquad (4.16)$$

Letting $M_1 = 1$, we obtain:

$$A(s) = 27.877 + s \qquad (4.17)$$

$$M(s) = 3{,}252 + s$$

$$L(s) = 3{,}252$$

Comparing the chosen overall transfer function ($s^2 + 28s + 400$) with a standard second order transfer function ($s^2 + 2\xi\omega_n s + {\omega_n}^2$), we obtain $\omega_n = 20$ and $\xi = 0.7$.

At this point, it would be appropriate to decide of how large or small the sampling time should be for discrete-time implementation of the controller. Clearly, a fast sampling rate improves the performance of a digital controller but at the cost of a more expensive computer. With small sampling rate, on the other hand, we might end up with an unstable system. Theoretically, according to the Nyquist sampling theorem, a sampling frequency twice the desired bandwidth is recommended. We shall, however, select a sampling frequency 35 [29] times the natural frequency, then the resulting sampling time would be $0.009\ sec$.

The summary of the parameters of the designed controller for the inner control loop is diagrammatically shown in Figure 4.9. The design of the outer loop controller shown in dashed lines in this figure is discussed in the following sub-section.

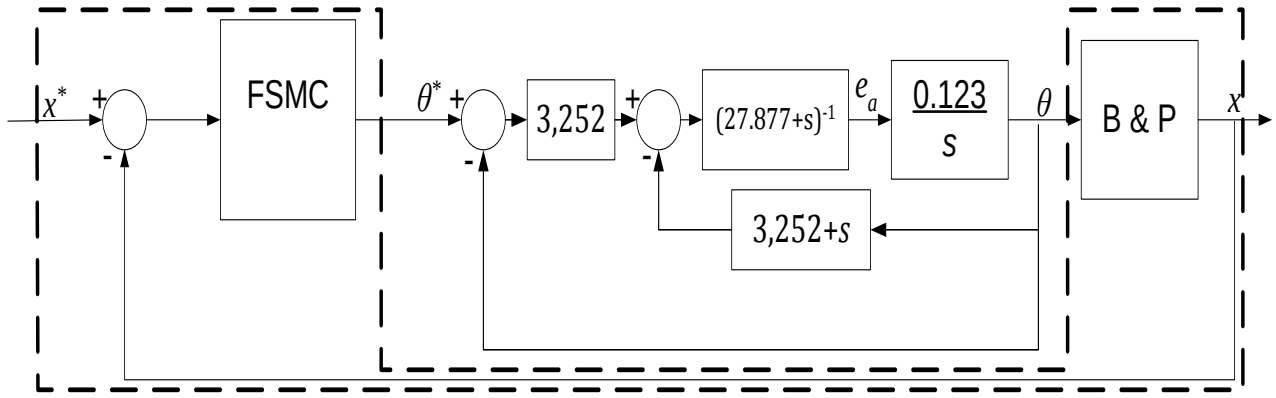


Figure 4.9: The complete design of the inner control loop

It is important to note that, though the 2-parameter configuration so selected uses minimum number of amplifiers and does not involve pole-zero cancellations, it cannot be realized on Simulink® for the reason that all blocks need to be represented by proper transfer functions. Therefore, the design could be realized in the following equivalent form as shown in Figure 4.10.

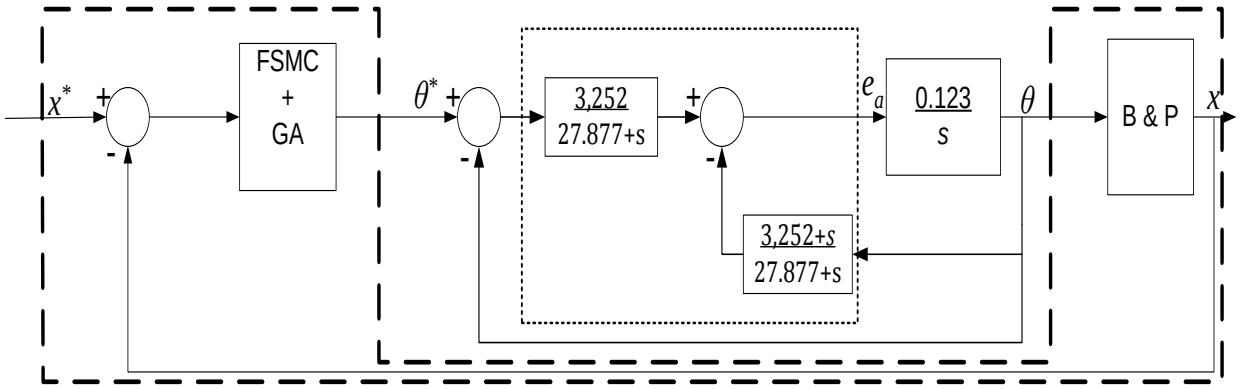


Figure 4.10: Alternative realization of the 2-parameter configuration for simulation purpose

4.4. DESIGN OF OUTER CONTROL LOOP

We shall design the controller for the outer loop by assuming that the inner loop controller is ideal with no transport delay as given in Figure 4.11[30].

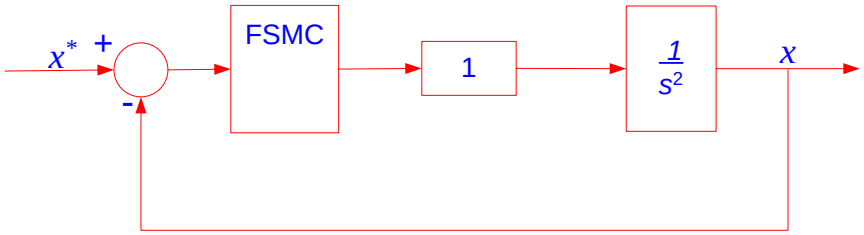


Figure 4.11: Schematic layout of outer loop controller design

Design of Sliding Mode Control

Based on equation (3.6), the sliding surface for a SISO second order plant shown in Figure 4.11 is given as:

$$s = \dot{e} + \lambda e \qquad (4.18)$$

The value for λ could be obtained by observing that during 'perfect' sliding, equation (4.18) reduces to $s = 0$ resulting in a first order differential equation in $e(t)$. Since the settling time of the inner loop is chosen as $0.4\ seconds$, the outer loop controller that is governed by the dynamics of the error $e(t)$ cannot settle any faster than $0.4\ seconds$. Furthermore, the requirement for the stability of the system trajectory confined to the sliding surface implies that $\lambda > 0$. Hence, we choose $\lambda = 1.25$ so that the response due to a unit step input settles in about $4\ seconds$. Here the rise time is taken to be equal to five time constants at which instant the response reaches and remains within 1% of its final value [25].

Noting that the B&P system is formulated in the transform domain, we obtain the control signal based on sliding mode control from equation (3.15) as:

$$u = -{b_o}^{-1}\{-\ddot{x}_d + \lambda\dot{e} + ksgn(s)\} \qquad (4.19)$$

To determine the value of k, it is assumed that the inclination angle of the plate will not exceed 30^0 [13] and that the first and second time derivative of the reference trajectory is negligible. In fact, for set-point tracking problems these derivative terms are zero. In addition the maximum velocity of the ball is given as $4\ mm/sec$ [13]. Taking the above assumptions

we find that the appropriate value of k that will not result in too large settling time is if it is selected as 2.

Fuzzy Logic and Genetic Algorithm

In non-ideal conditions, the system trajectories chatter rather than sliding along the sliding surface. Since chattering is not a desired phenomenon, we shall fuzzify the relationship between the gain k and the 'distance' s using fuzzy logic. The larger the distance is the larger must be the value of k to commence sliding as fast as possible. The smaller the distance is the smaller must be the gain k so as not to introduce too much chattering.

Three fuzzy sets named S, M, L respectively for small, medium and large are chosen to fuzzify the sliding surface. The subscripts P and N are short for positive and negative, respectively.

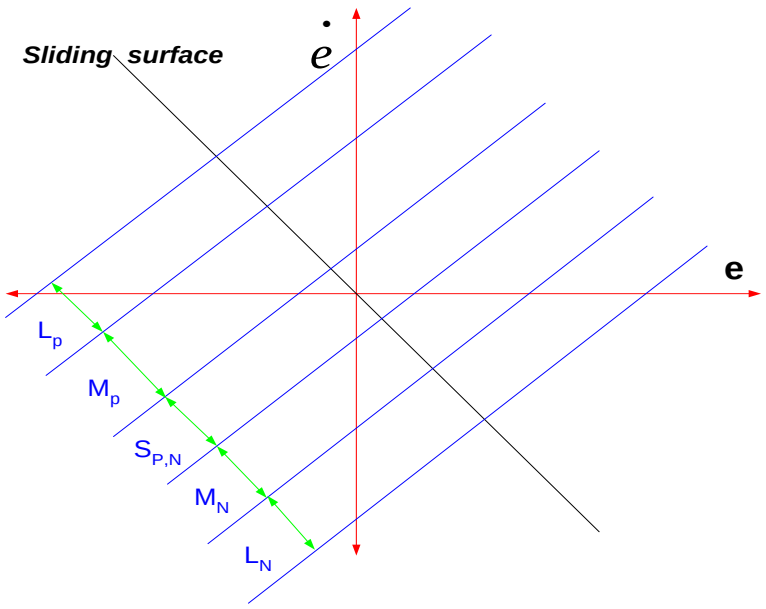


Figure 4.12: Fuzzification of the sliding surface

Similarly the switching gain k is fuzzified into three fuzzy sets named S, M, L respectively for small, medium and large. Approximate sketch of the fuzzy sets using Fuzzy Logic Toolbox for the input variable s are shown in Figure 4.13:

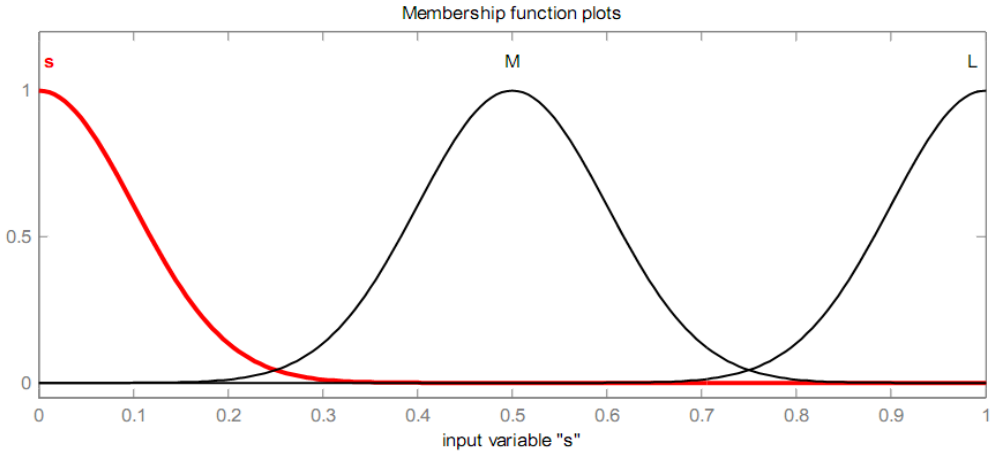


Figure 4.13: Approximate 'shapes' of the fuzzy sets for the normalized variable s

In literature, different approaches were used to design the hybrid controller consisting of sliding mode control and fuzzy logic control. A summary of the proposed design approaches is shown in the Table 4.6.

Table 4.6: Summary of various methods used in the hybrid FSMC

No.	Input	Fuzzy Rules	Output
1	$\begin{Bmatrix} e \\ \dot{e} \end{Bmatrix}$	$\longrightarrow$	k
2	$\begin{Bmatrix} e \\ \dot{e} \\ \int e \end{Bmatrix}$	$\longrightarrow$	k
3	$\begin{Bmatrix} s \\ \Delta s \end{Bmatrix}$	$\longrightarrow$	u
4	$\begin{Bmatrix} e \\ \Delta e \end{Bmatrix}$	$\longrightarrow$	$ksat(s)$
5	s	$\longrightarrow$	k
6	$s\dot{s}$	$\longrightarrow$	Δk
7	$\lvert s \rvert$	$\longrightarrow$	k

The rules used to map the input and output fuzzy sets are as follows:

$$rule\ 1: If\ |s(t)|\ is\ \tilde{S}, then\ k(t) is\ \tilde{S} \qquad (4.20)$$
$$rule\ 2: If\ |s(t)|\ is\ \tilde{M}, then\ k(t) is\ \tilde{M}$$
$$rule\ 3: If\ |s(t)|\ is\ \tilde{L}, then\ k(t) is\ \tilde{L}$$

However, using fuzzy logic alone the shape of the membership functions could be determined using trial-and-error methods only. In our discussion 'shape' shall mean the location of each fuzzy set along the universe of discourse, the width of the membership functions and the amount of overlap with other fuzzy sets.

In this context, we shall prefer to find the optimal shape of the membership functions using genetic algorithm. The source code for the programme is developed on MATLAB® platform and is included in Appendix D for the purpose of reference.

The design objective is to find the best (the fittest) population that will minimize the squared error in approximating the sliding line as given in equation (4.21). We note that equation (4.21) represents a line in the phase plane.

$$\dot{e} = -1.25e \qquad (4.21)$$

Each of the 6 fuzzy sets (3 for the input fuzzy set s and 3 for output fuzzy set k) are allowed to move along the universe of discourse and vary their width, i.e. both the mean and the variance of the Gaussian membership function representing the fuzzy sets vary. Since we have 12 variables to be encoded into the chromosome (2 for each fuzzy set) and if we represent each variable using 6 bits and, then we have an individual population 72 bits long.

100 populations are randomly generated each obtained by concatenating a randomly generated bit string of length 72. These are next scaled into the basic ranges for the variables $s\ (0 \leftrightarrow 0.4)$and $k(0 \leftrightarrow 2)$. The design parameters used in the simulation are given in Table 4.7.

A representative bit string (chromosome) looks like:
'110110101000101100001111010000000010010110101000101100001111010000000010'.

Table 4.7: Selected design parameters for fuzzy and genetic algorithm

No.	Design Parameter	Value
1	Population size (Number of individual)	*100*
2	Membership function	*Gaussian:* m, σ
3	Number of input fuzzy sets (s)	*3*
4	Number of output fuzzy sets (k)	*3*
5	Total number of variable to be determined	*2x6=12*
6	Length of bits to encode each variable	*6*
7	Chromosome length	*12x6=72*
8	Optimization function	$minimize(e^2)$
9	Reproduction rate	*0.15*
10	Crossover rate	*0.8*
11	Mutation rate	*0.05*
12	Fuzzy inference rule	*max*

Since the initial 100 populations are randomly generated as concatenated bits of strings of length 72 each and as some of the program codes require random number generation, one cannot expect the exact same shapes for the membership function to be displayed among different simulation runs. Nonetheless, similar patterns are to be expected among these outputs which provide sufficient design guideline in order to get approximate shapes for the fuzzy membership functions.

CHAPTER 5
SIMULATION RESULT AND DISCUSSION

5.1. INTRODUCTION

The 3-D model for the Ball and Plate system designed on V-Realm Builder is shown in Figure 5.1.

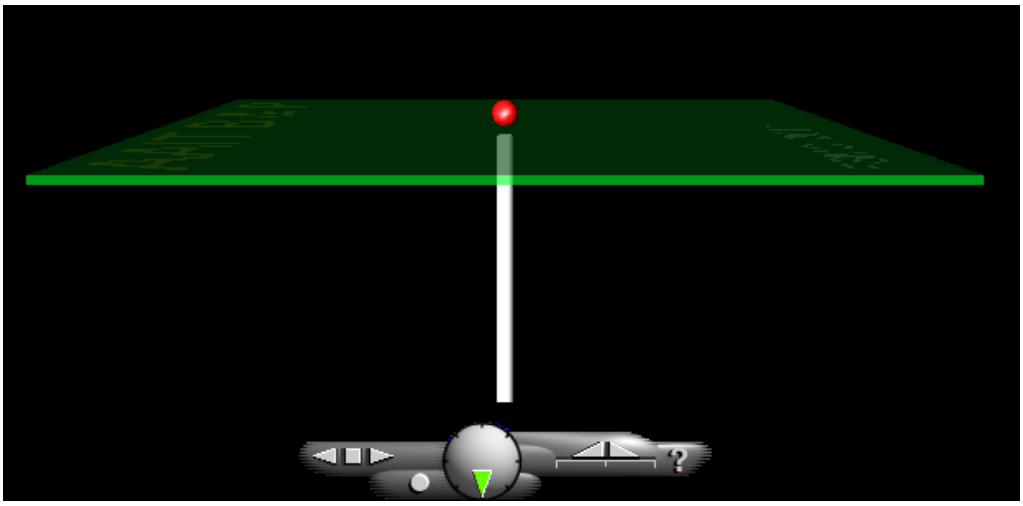


Figure 5.1: The 3-D model of the Ball and Plate with the supporting shaft

The two perpendicular plate inclinations about the X and Z axes are controlled by signals from the Simulink® model; however, the rotation about the Y axis is restricted. The translation of the ball is realized along the $X-Z$ plane of the plate; while no translation is allowed along the Y axis.

Since Simulink® signals are available only for 2-Dimentional plot; we have to transform the position and angle data of the model into 3-Dimentional coordinates so as to display an animating virtual reality (VR) of the B&P system. While doing so we consider the differences in the coordinate system convention between that of MATLAB® and VRML as shown in Figure 3.6.

The complete Simulink® model implemented on MATLAB® is shown in Figure 5.3. The model is used in the study of static and dynamic position tracking problems whose simulation results are given in the next sections.

The interface between the virtual reality (VR) of the B&P system as shown in Figure 5.1 and the Simulink® model as shown in Figure 5.2 is enabled through Simulink® 3D Animation™ product. This interface, consisting of the 2-D to 3-D coordinate transformation blocks and the VR Sink is shown using dashed lines in Figure 5.3.

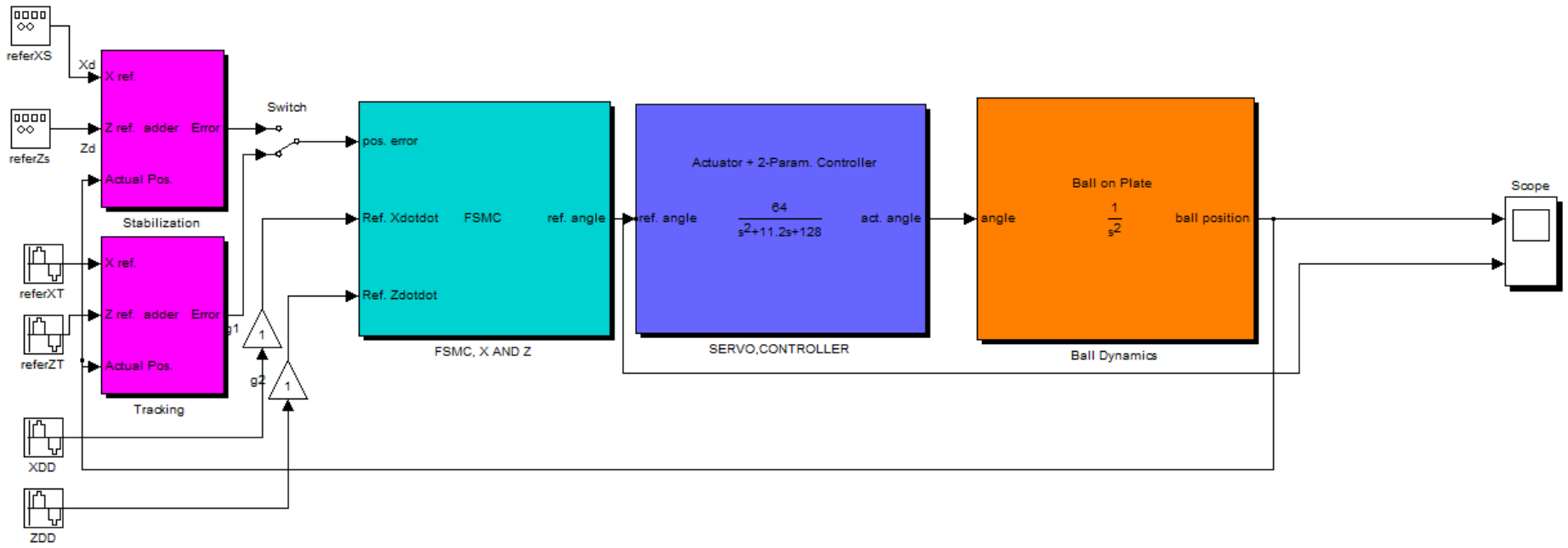


Figure 5.2: Simulink® realization of the B&P system

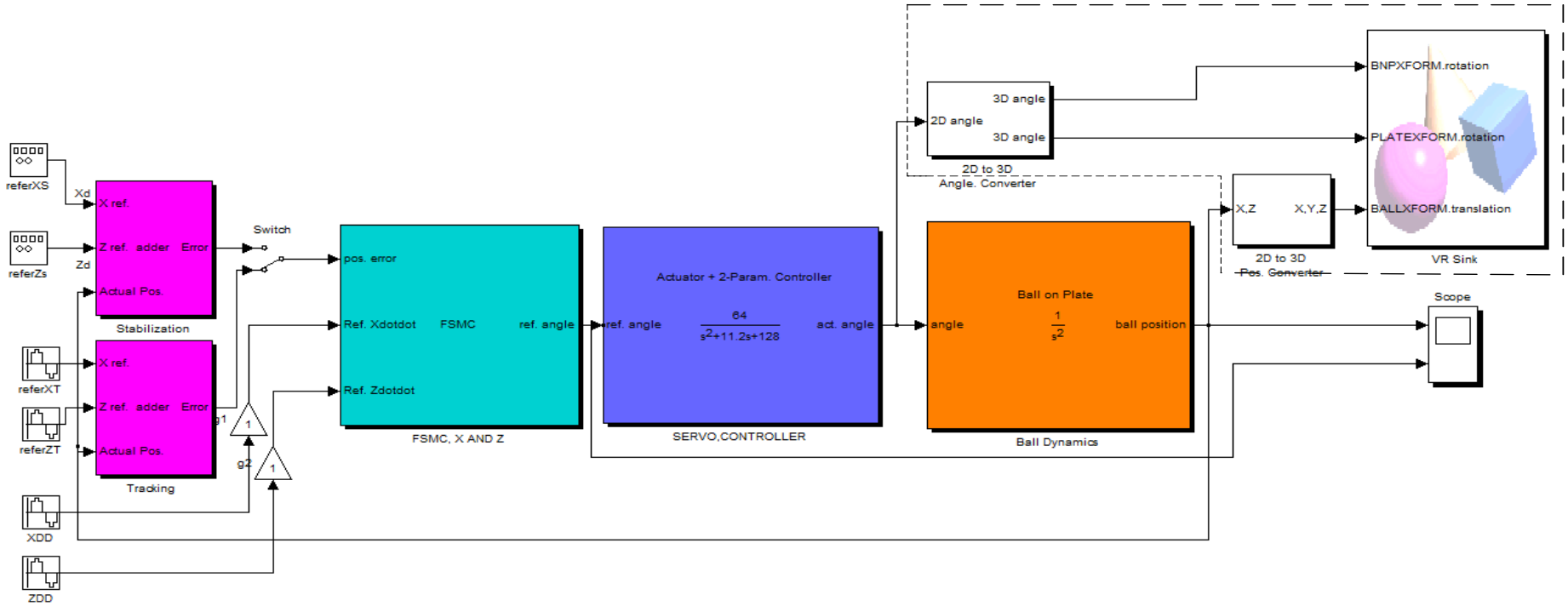


Figure 5.3: The complete Simulink® realization of the B&P system

5.2. PERFORMANCE OF STATIC POSITION TRACKING

The ball is initially placed at the origin $(0,0)$ of the (X, Z) coordinate system which is taken to be the center of the plate. The ball is then required to translate and reposition itself at a reference coordinate on the surface of the plate. Different cases have been considered to test the performance of the designed FSMC and SMC.

case i. The evolution of the sliding surface and the control θ_x with time as the ball executes free translation to reach and stay in the vicinity of the set-point $(-0.3, 0.4)$ with FSMC is shown in Figure 5.4. The 'variable' nature of the control which varies depending on the sign of s is evident.

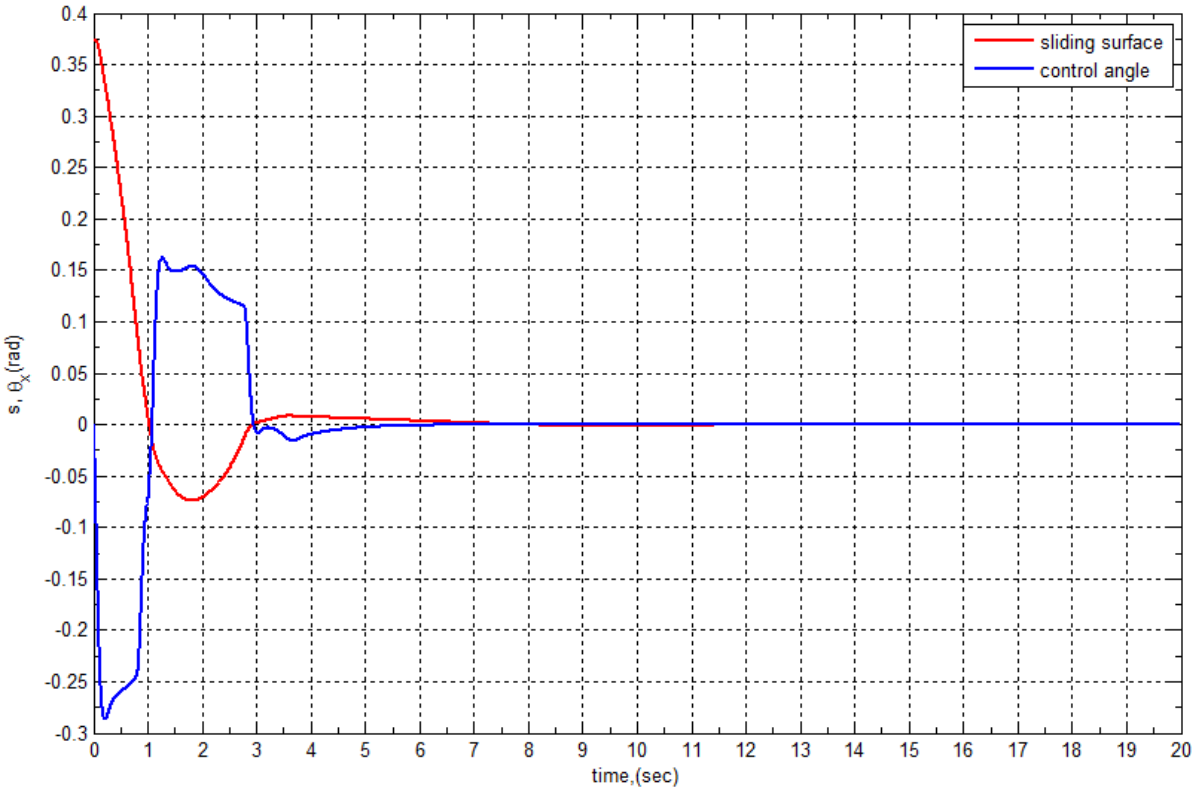


Figure 5.4: Plot of s and θ_x with the designed FSM controller.

Simulation results with SMC for the same setting is shown in Figure 5.5. It is seen that there is too much chattering in the control signal θ_x even after the error dynamics governing the sliding surface reaches within 2% of its steady state value. Due to the chattering in the control

signal, the ball keeps on oscillating about the origin indefinitely and it may eventually be trapped in a limit cycle.

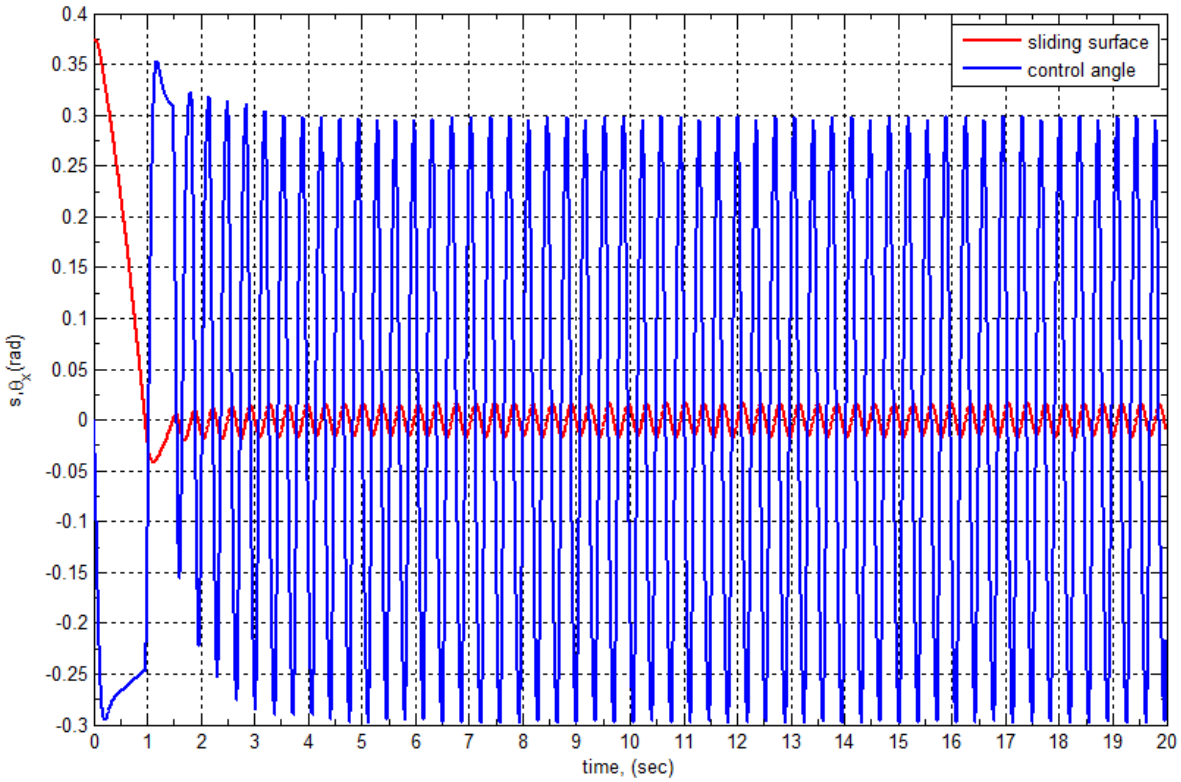
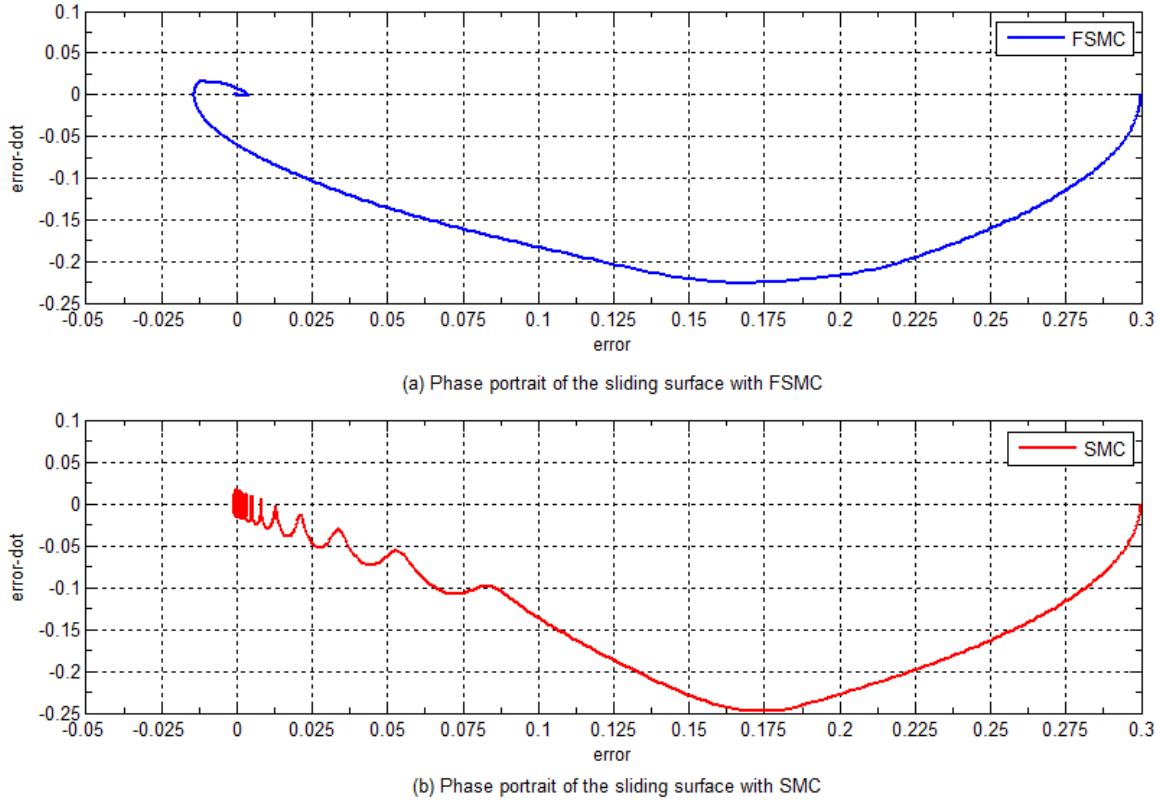


Figure 5.5: Plot of s and θ_x with SM controller.

case ii. The portrayal of the sliding surface in the phase plane with both SM and FSM controllers is shown in Figure 5.6. From the comparison of the two sub-plots, SMC exhibits too much chattering in the control signal and a closer observation of the darker region around the origin of the phase plane shows that a system trajectory governed by SMC will continue in a persisting motion of elliptical paths around the origin for an indefinite time.



(a) Phase portrait of the sliding surface with FSMC

(b) Phase portrait of the sliding surface with SMC

Figure 5.6: Comparison of the phase portrait of the sliding surface designed by (a) FSMC (b) SMC

case iii. The simulation result for the linear position of the ball versus the command position and the angular position of the plate versus the reference angular position is presented in Figure 5.7 and 5.8. From Figure 5.7 it is seen that the ball is able to reach the steady state values of the desired positions in 2 and 3 $seconds$ for the X and Z directions, respectively.

In Figure 5.8, the reference and the actual inclination angle are shown only for the X direction of motion. It can be seen that the steady state value is achieved in approximately 3 seconds.

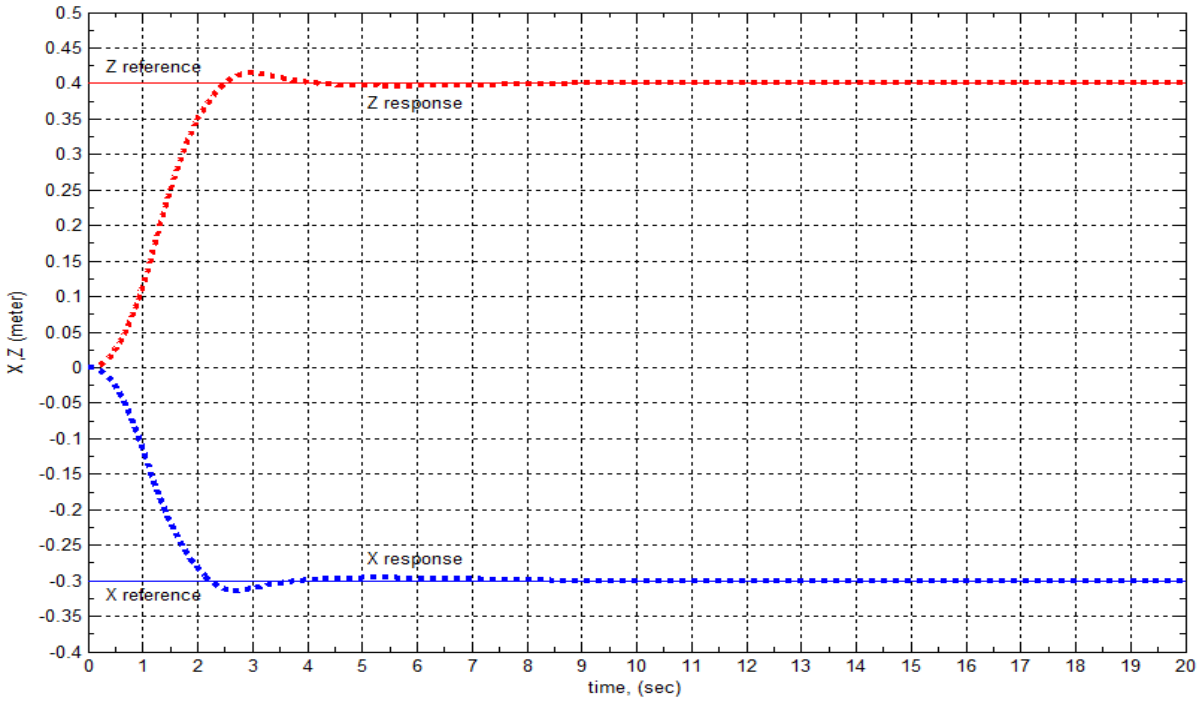


Figure 5.7: The response of the linear position of the ball with FSM controller

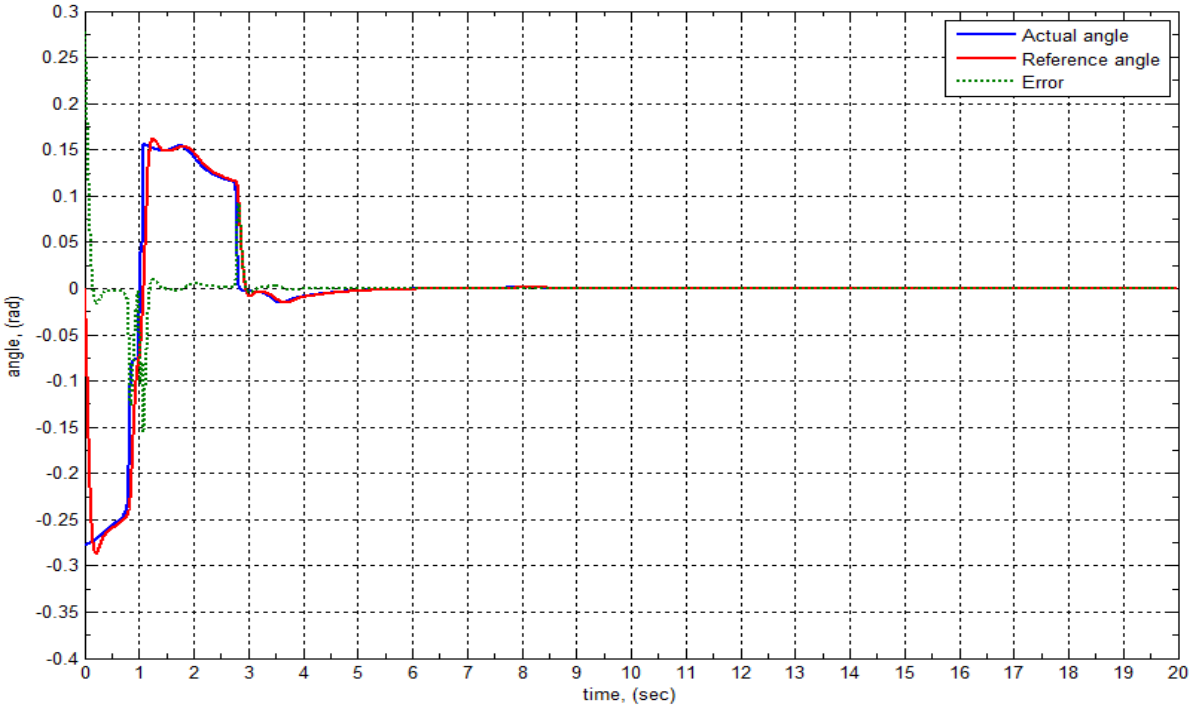


Figure 5.8: The response of the plate inclination and the error with FSMC

In Figure (5.9) and (5.10), corresponding results are plotted with the controller designed using SM design principles. The excessive chattering present in the control is evident from the Figure 5.10. It is interesting to observe from animated displays the persistent oscillation of the plate in the VR model. This shows that the performance of FSMC is superior to that of SMC and it is also an ideal choise for implementation purposes.

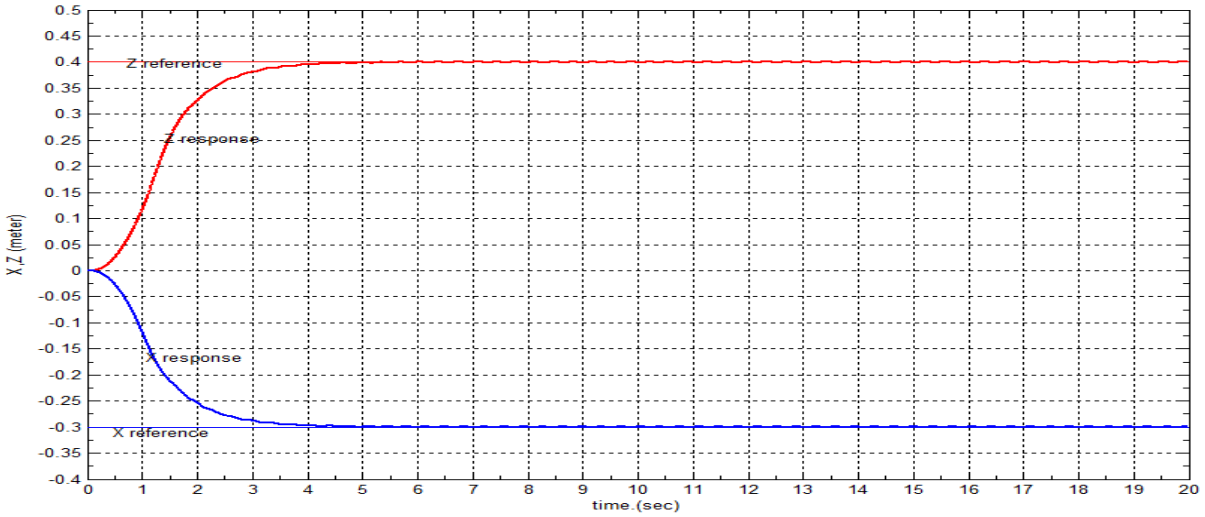


Figure 5.9: The response of the linear position of the ball with SM controller

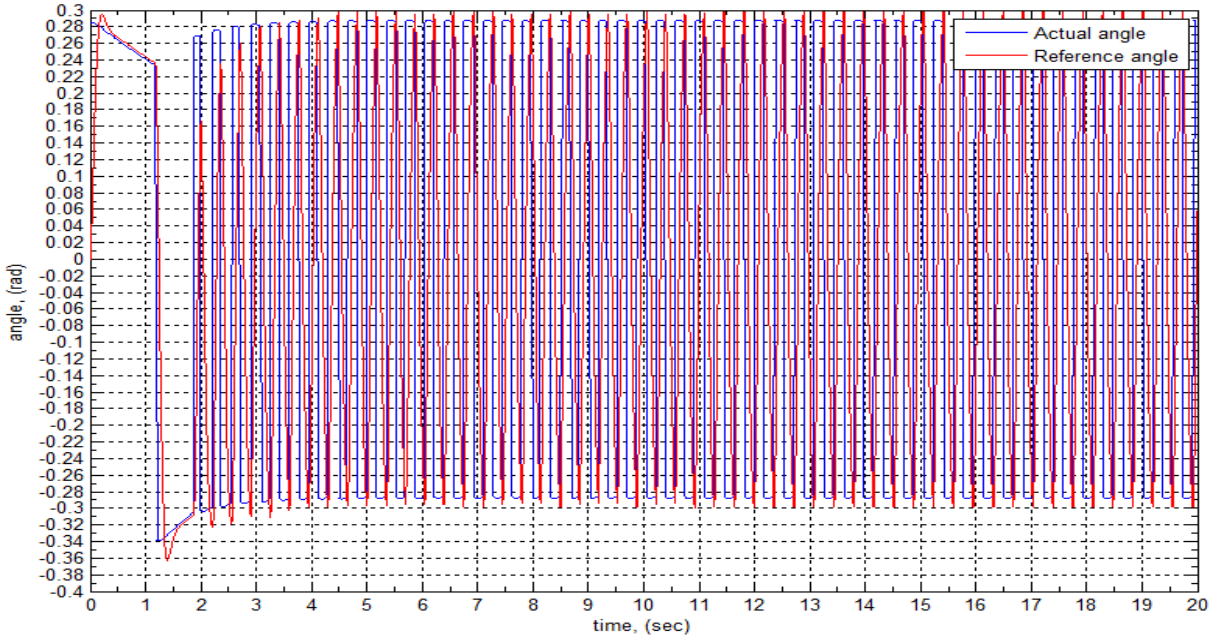


Figure 5.10: The response of the plate inclination in the *X* direction with SMC

case iv. In order to test the performance of the controllers in the presence of external disturbances, a sinusoidal disturbance signal with amplitude of 0.5 and frequency 100 rad/sec is applied additively at the input to the plant.

The plot of the error between the time response of the ball's position in the presence of external disturbance and with no external disturbance in Figure 5.11 shows that even in the presence of disturbance the designed controller is able to perform satisfactorily.

The robustness of the designed controller to external disturbances such as a wind gust is shown in Figure 5.12. The disturbance is taken as an impulse with duration 0.1 and amplitude 0.3. It is seen from Figure 5.11 and 5.12 that even in the presence of disturbances, the robustness property of the sliding mode control is maintained.

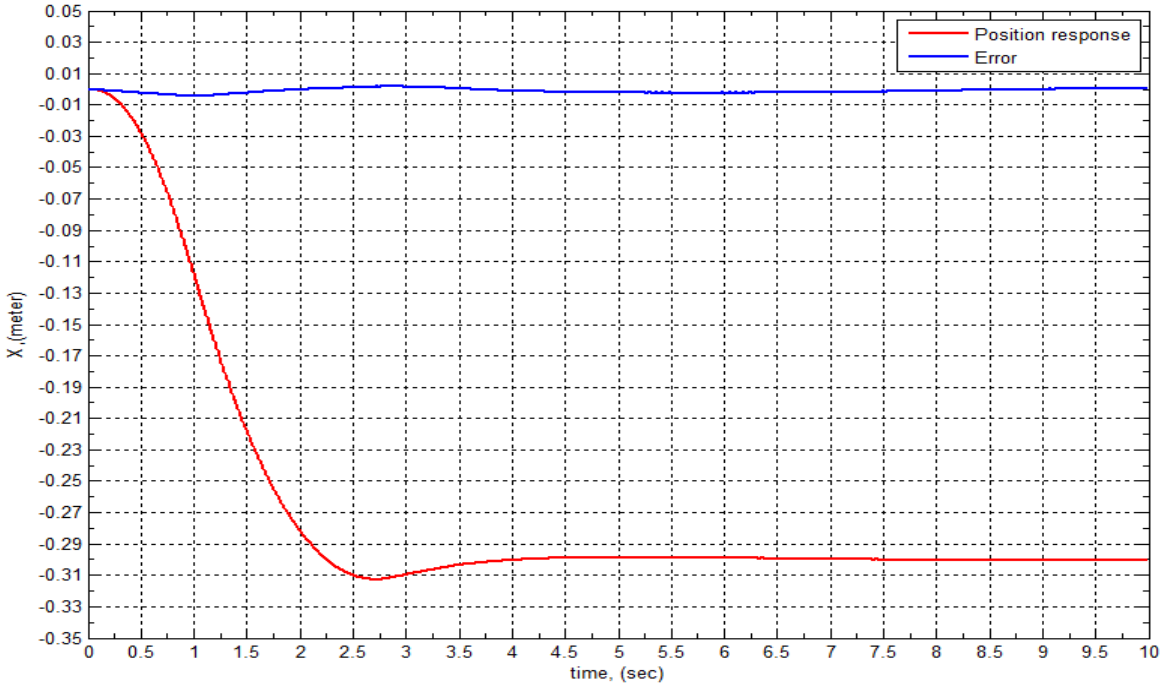


Figure 5.11: The time response of the ball's position with FSMC in the presence of a sinusoidal external disturbance

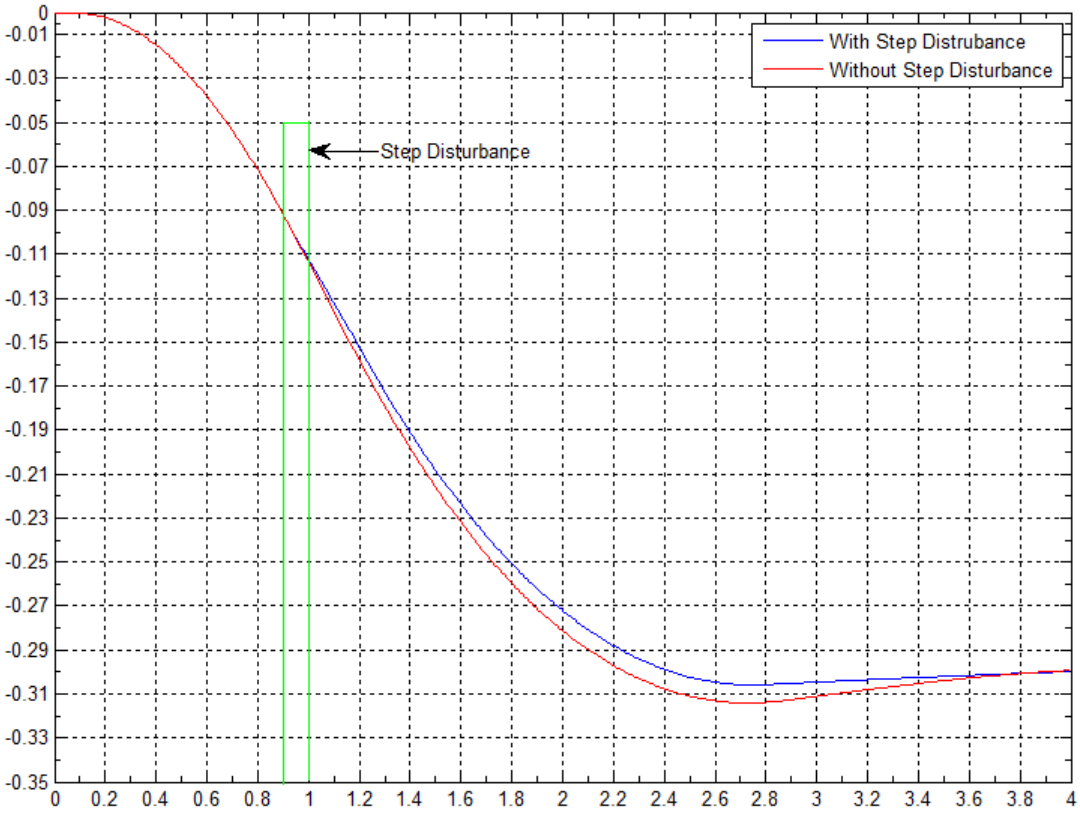


Figure 5.12: The time response of the ball's position with FSMC in the presence of an impulse external disturbance

Since the sliding surface parameters are designed independently of the system parameters, the system designed using sliding mode as the controller is robust to parameter variations. This is demonstrated in Figure 5.13 with a ± 2 uncertainty in the value of $b_o = 7$. We note that the ball could still reach and remain within 2% of its steady state value of $x = 0.3$ for all the cases $b_o = 9, b_o = 7$ and $b_o = 5$ assuring that the controller designed using (fuzzy) sliding mode control could result in a 'robust' system.

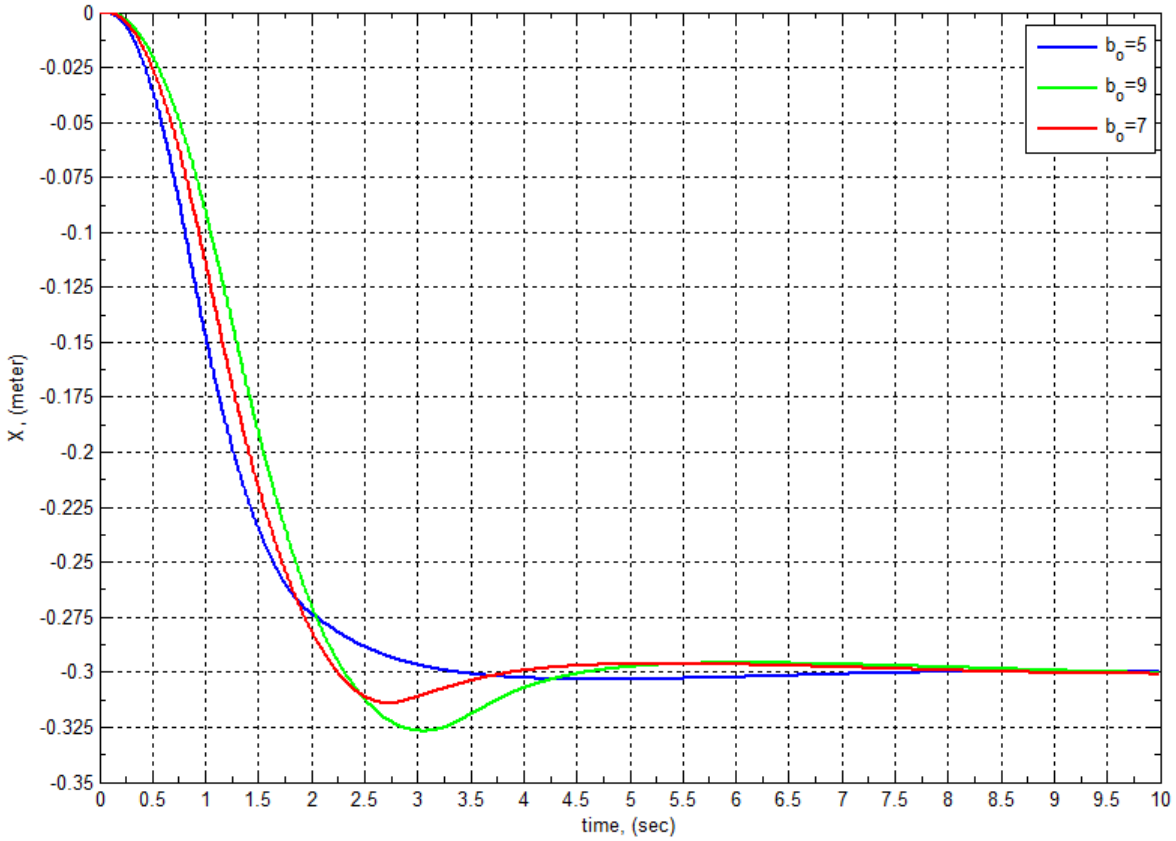


Figure 5.13: The position control of the ball under parameter perturbation

5.3. PERFORMACE OF DYNAMIC POSITION TRACKING

A circular trajectory of 0.4m radius is taken as a reference signal to demonstrate the performance of trajectory tracking capability of the ball. The angular frequency of the sinusoidal inputs in the X and Z- directions is considered to be 0.8 rad/sec.

case i. The plot of the time response of the sliding surface in Figure 5.14 shows that with FSMC, the required reference circle could be tracked with a negligible error. However, for SMC it is seen from the same figure that the error is in the order of 0.06 implying that the reference trajectory cannot be tracked with an acceptible limit of error.

The time response of the control angle with both FSM and SM controllers is shown in Figure 5.15 (a) and Figure 5.15 (b), respectively. In the second sub-plot, as was the case with static position tracking results, there is too much oscillation in the control angle.

Due to symmetry, results are shown for plate inclination in the X direction only.

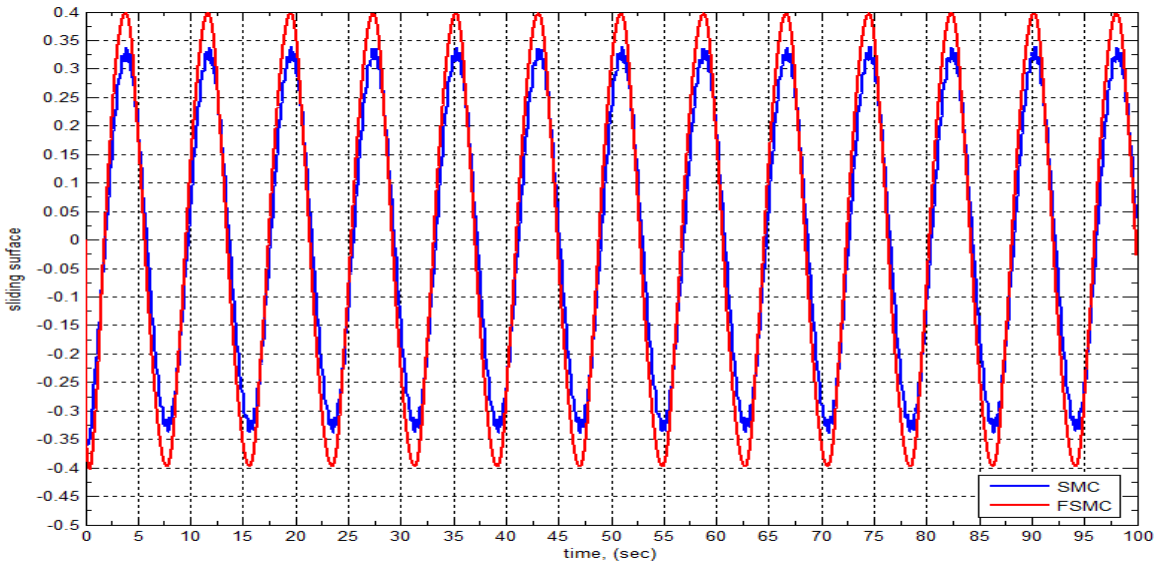


Figure 5.14: The evolution of the sliding surface with time.

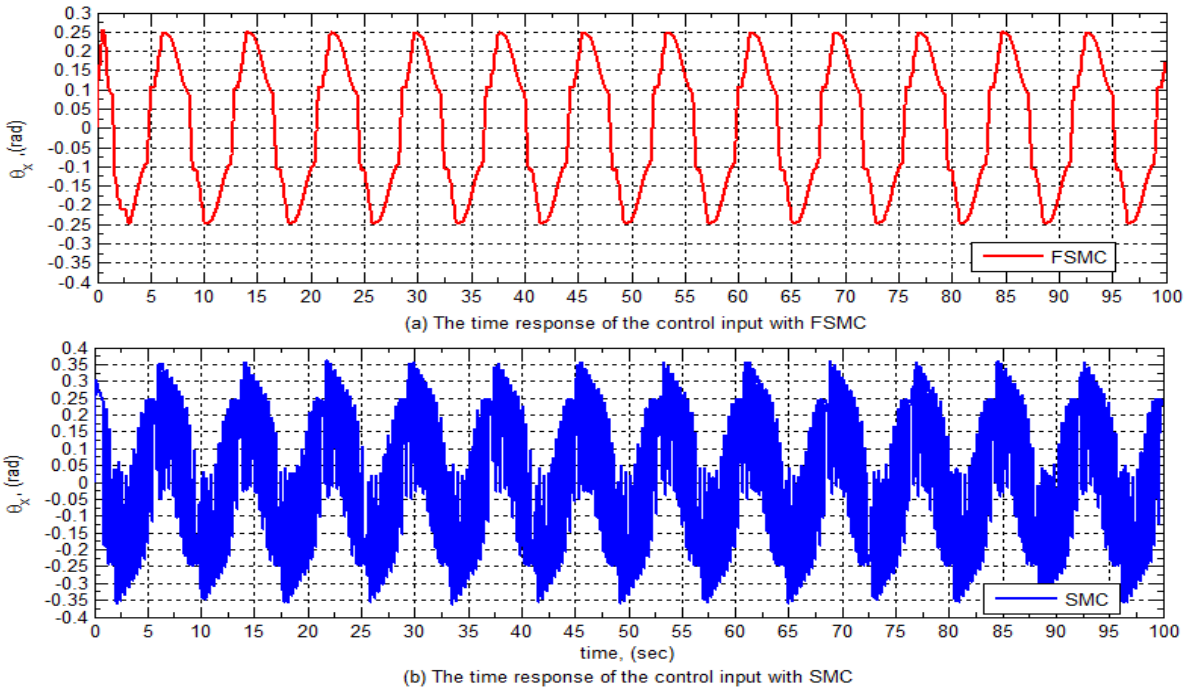


Figure 5.15: Comparison of the time response of the control input for (a) FSMC and (b) SMC

case ii. The variation of the ball's position and the reference trajectory as a function of time are shown in Figure 5.16 and 5.17. It is observed that with the controller designed using FSM, the steady state tracking error of the ball is less than 0.02. However, the ball could not track the reference position with a steady state error of less than 0.02 using SMC.

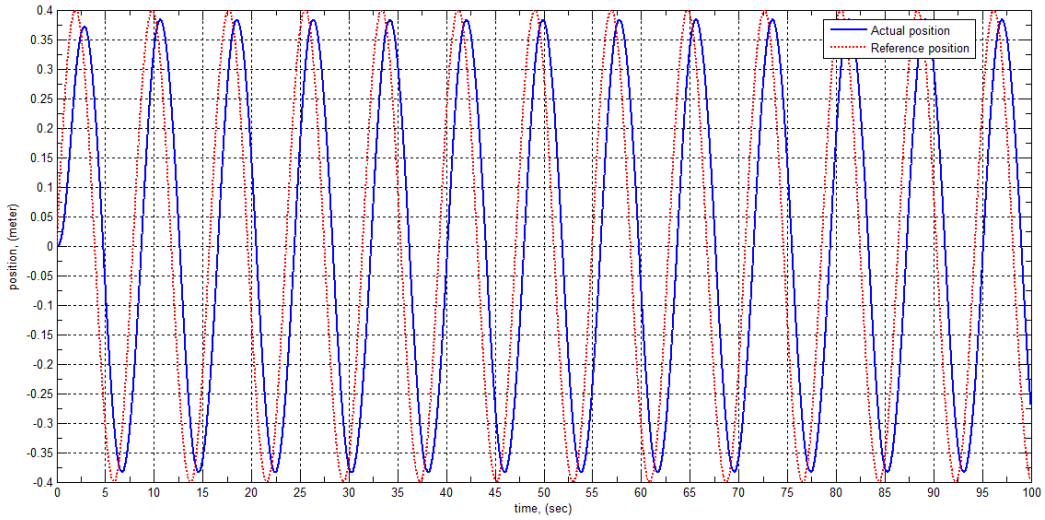


Figure 5.16: The position of the ball with FSMC

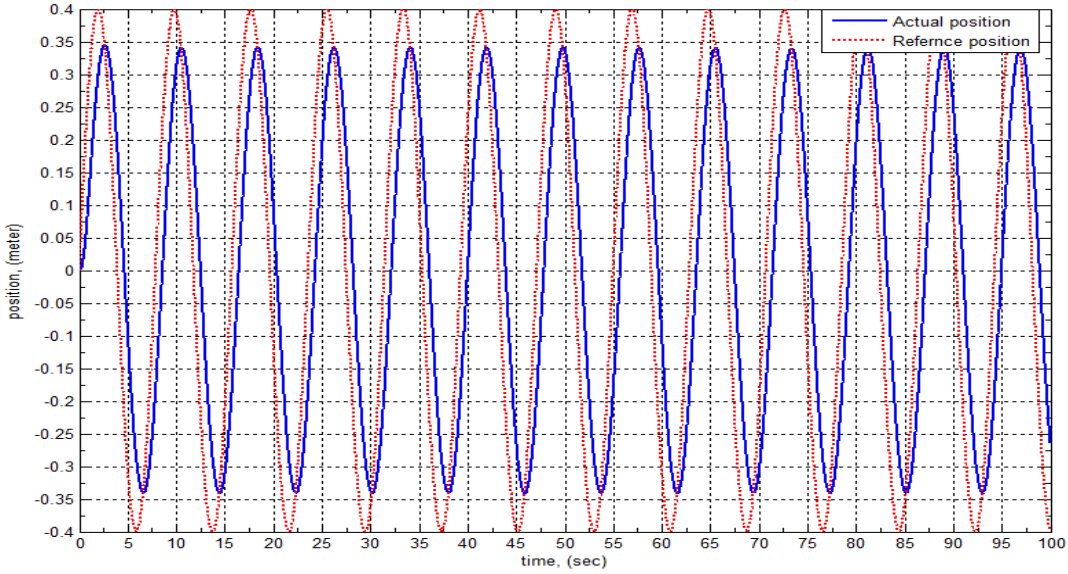


Figure 5.17: The position of the ball with SMC

case iii. The plot of X-position versus Z-position is presented in Figure 5.19. The ball is allowed to track a circular trajectory at a frequency of 0.8 rad/sec. It is shown that the controller designed with FSMC has better performance than that designed using SMC. If the ball is required to track a circular trajectory at a lower frequency, it is possible to get an acceptable tracking performance. In Figure 5.19, the frequency is set at 0.2 rad/sec and the ball is able to execute a complete motion along the reference circle in approximately 34 $seconds$. Moreover, the tracking error could be reduced if we further decrease the tracking speed but at the cost of a longer time for a complete revolution. It is further observed from Figure 5.18 at a frequency of 0.8 rad/sec, the necessary time needed for a complete revolution around a circle of radius $0.4m$ is 10 $seconds$.

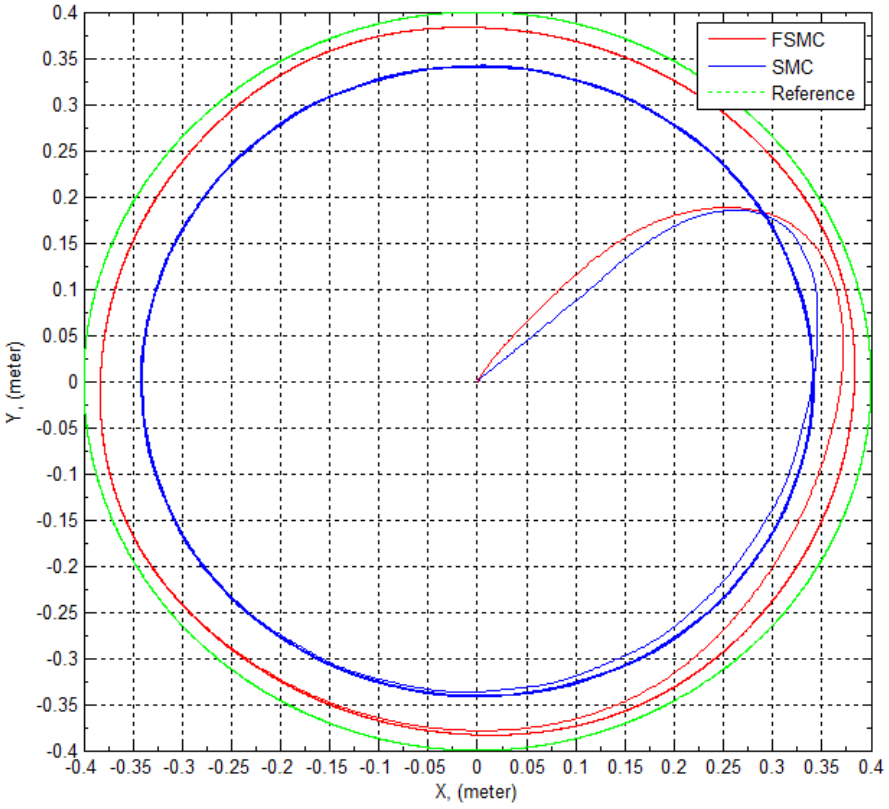


Figure 5.18: Circular trajectory tracking performance at 0.8 rad/sec with FSMC and SMC

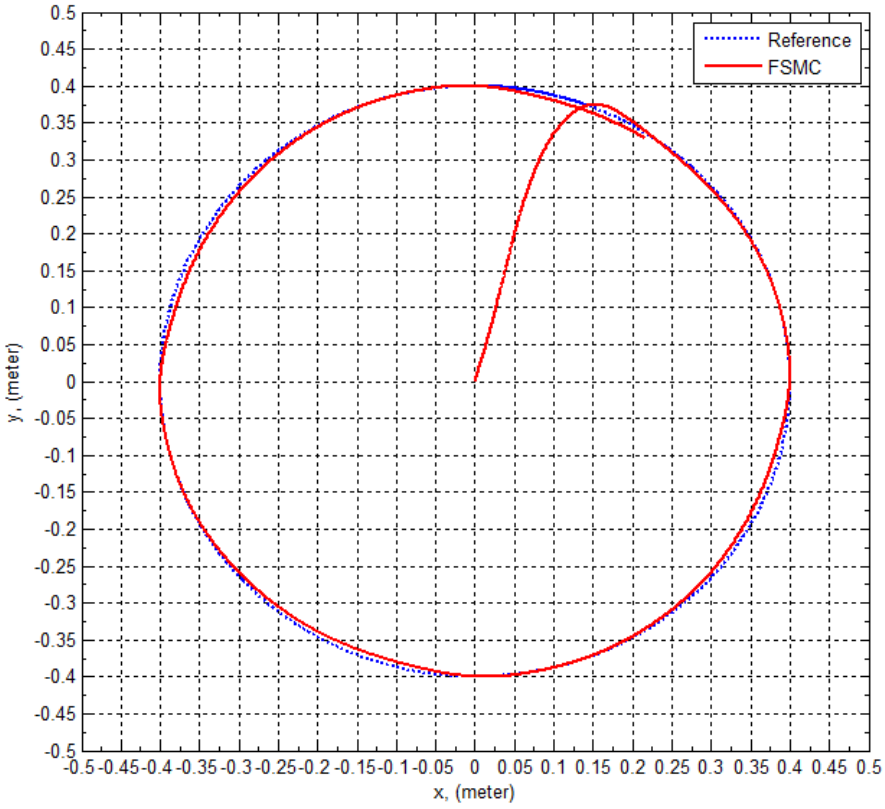


Figure 5.19: Circular trajectory tracking performance at 0.2 rad/sec with FSMC and SMC

case iv. As can be seen from Figure 5.20, the switching gain changes its magnitude with time. Initially due to a large error between the command and the actual position, a larger gain is required to bring the state to the sliding surface as soon as possible. However, once motion on the sliding surface commences, the switching gain decreases eventually; thus, reducing the control input required. However, for sliding mode control, the gain keeps oscillating between -2 and2 as shown in Figure 5.21.

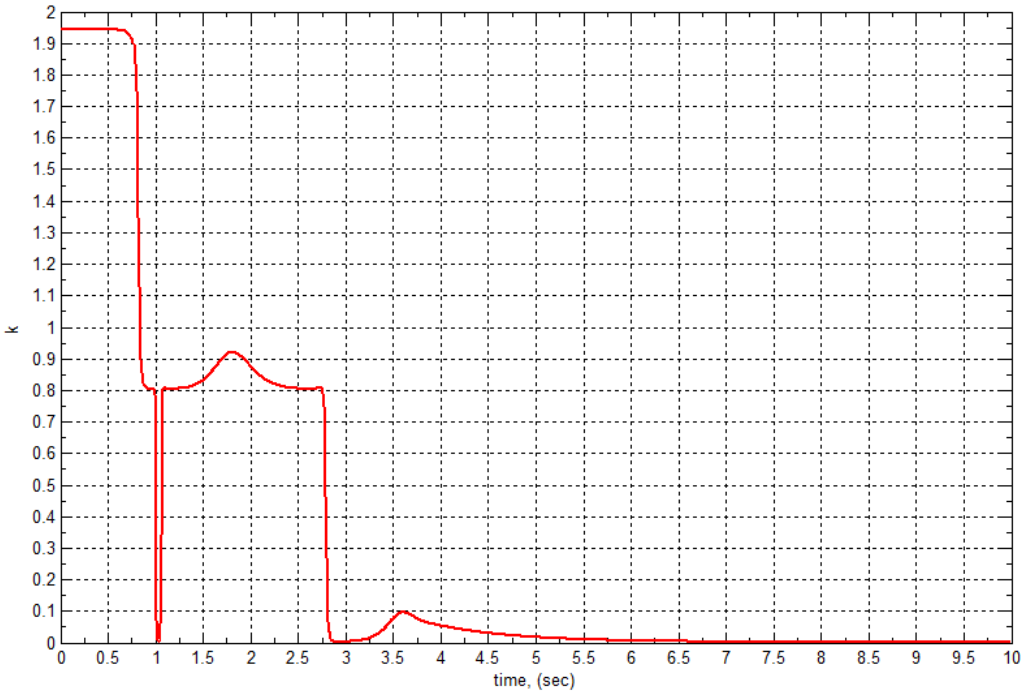


Figure 5.20: The variation of the switching gain with FSMC

Figure 5.21: The variation of the switching gain with SMC

CHAPTER 6

CONCLUSIONS, RECOMMENDATIONS AND FUTURE WORK

6.1. CONCLUSIONS

In this thesis, Fuzzy Sliding Mode controller is developed to study the control problem of the B&P system. The stability of the designed controller has been verified using Lyapunov's Direct Method. In addition various evaluation criteria have been chosen to compare the performance of the controller designed using FSMC with that of SMC.

It is observed that the ball is able to reach steady state position in less than 3 $seconds$ for static position tracking commands using FSMC. The chattering observed in the corresponding results with SMC is evident of the difficulty involved in the practical design of controllers using this technique. For the case of SMC, it is interesting to see the 3-D animation of the VR model of the system showing persistent oscillation of the plate about the origin. The performance of FSMC controller is found to be superior to that of the SMC. Moreover, in terms of practical implementations, FSMC is shown to be useful.

The effect of external disturbance in the performance of the controllers is demonstrated using a sinusoidal disturbance signal of amplitude 0.5 and frequency 100 rad/sec. Even in the presence of disturbance, it is seen that sliding mode control yields superior control in the feedback system owing its robustness property.

Furthermore, considering a ± 2 uncertainty in the value of plant parameter b_o, it is confirmed that the ball could still remain within 2% of its steady state value of the command position assuring that the controller designed using (Fuzzy) Sliding Mode Control could result in a 'robust' feedback system.

A circular trajectory of 0.4m radius is taken as a reference signal to demonstrate the performance of trajectory tracking capability of the ball. The angular frequency of the sinusoidal inputs in both X and Z directions is taken as 0.8 rad/sec. For the controller

designed using FSMC, the ball could track the reference position within an error of less than 0.02; however, this cannot be achieved using SMC.

The tracking performance is further evaluated by changing the tracking speed. At a frequency of $0.2\ rad/sec$, the necessary time for a complete revolution is $35\ seconds$. At an angular frequency $0.8\ rad/sec$ the ball is able to execute a complete motion along the reference circle in $10\ seconds$ but with more tracking error.

From the simulation results of FSMC, it is shown how the switching gain eventually approaches zero. On the other hand, with sliding mode controller, the gain keeps on switching infinitely fast between the values $+2$ and -2.

6.2. RECOMMENDATION

In this thesis, the multivariable property of the B&P system is handled by assuming independent motions in the two orthogonal directions. However, the coupling terms and the nonlinearities which led to the modeling of the system as a single-input single-output (SISO) problem could be considered to investigate the multivariable property of the B&P system.

In this thesis, a double-feedback loop configuration facilitated the breakdown of the controller design into simpler ones. However, the problem can be further explored by using a single control loop. This increases the order of the mathematical model of the system and thereby the sliding surface.

6.3. FUTURE WORK

In the future, it is recommended to explore further applications of the method in the study of large scale systems, discrete-time systems, distributed-parameter systems, time-delay systems, stochastic systems, non-minimum phase system etc. Recent publications in the study of SMC include advanced methods to alleviate chattering problem using 2-sliding surfaces and non-linear sliding surfaces.

REFERENCES

[1] Hongwei Liu and Yanyang Liang, *Trajectory tracking sliding mode control of ball and plate system*, School of Information Engineering Southwest University of Science and Technology, 2010.

[2] Hongrui Wang, Yantao Tian, Zhen Sui, Xuefei Zhang and Ce Ding, *Tracking Control of Ball and Plate System with a Double Feedback Loop Structure*, IEEE, 2007.

[3] De-hu Yuan. *Pneumatic servo ball & plate system based on touch screen and oscillating cylinder*. SJTU-SMC Technology Center. Shanghai Jiao Tong University, china.

[4] Peter E. Wellstead, *Introduction to Physical System Modeling*. Control Systems Principles, 2000.

[5] Andrej Knuplei, Amor Chowdhur and Rajko SveEko, *Modeling and Control design for the ball and plate system*, University of Maribor. Faculty of Electrical Engineering and Computer Science.

[6] Dejun Liu, Yantao Tian and Huida Duan, *Ball and Plate Control System based on sliding mode control with uncertain items observe compensation*, IEEE, 2009.

[7] Wei Wang, *Control of a Ball and Beam System*, School of Mechanical Engineering, The University of Adelaide, 2007.

[8] Greg Andrews, Chris Colasuonno and Aaron Herrmann, *Ball on Plate Balancing System*, Rensselaer Polytechnic Institute, 2004.

[9] Huida Duan ,Yantao Tian and Guangbin Wang, *Trajectory Tracking Control of Ball and Plate System Based on Auto-Disturbance Rejection Controller*, Asian Control Conference, 2009.

[10] Qing Zheng, *Stability Analysis and Application in Disturbance Decoupling Control*, Master of Engineering in Electrical Engineering National University of Singapore,2003.

[11] Nima Mohajerin and Mohammad Bagher Menhaj, *Position Control of Ball and Plate System Using Fuzzy Logic Controller*, Amirkabir University of Technology, Electrical and Electronics Engineering Department, Systems and Control Group.

[12] Miad Moarref, Mohsen Saadat and Gholamreza Vossoughi, *Mechatronic Design and Position Control of a Novel Ball and Plate System*, IEEE, 2008.

[13] Xinghe Fan, Naiyao Zhang and Shujie Teng, *Trajectory planning and tracking of ball and plate system using hierarchical fuzzy control scheme*. Department of Automation, Tsinghua University, China.

[14] F. Ashrafzadeh, E.P. Nowicki, R. Boozarjomehry and J.C. Salmon, *Optimal Synthesis of Fuzzy Sliding Mode Controllers*, IEEE, 1996.

[15] John Y. Hung and James C. Hung, *Chatter Reduction in Variable Structure Control*. IEEE, 1994.

[16] Andreja Rojko and Karel Jezernik, *Adaptive Fuzzy Sliding Mode Control of Robot Manupilator*, University of Maribor, Faculty of Electrical Engineering and Computer Science, 2000.

[17] John Y. Hung, Weibing Gao and James C. Hung, *Variable Structure Control: A Survey*, IEEE, 1993.

[18] Jean-Jacques E. Slotine and Weiping Li, *Nonlinear Applied Control*, Prentice-Hall, Inc., 1991.

[19] V.I.Utkin, *Sliding Mode Control: Mathematical Tools, Design and Applications*, Lecture note, 2004.

[20] (Edited) Asif Sabanovic, Leonid M. Fridman and Sarah Spurgeon, *Variable Structure Systems from principles to implementation*, The Institution of Engineering and Technology, 2004.

[21] Chung-Chun Kung and Chia-Chang Liao, *Fuzzy-Sliding Mode Controller Design For Tracking Control Of Non-Linear System*, Department of Electrical Engineering, Tatung Inst. of Technology.

[22] K.David Young, Vadim I.Utkin and Umit Ozguner, *A Control Engineer's Guide to Sliding Mode Control*, IEEE,1999.

[23] Feijun Song and Samuel M. Smith, *A Comparison of Sliding Mode Fuzzy Controller and Fuzzy Sliding Mode Controller*. Ocean Engineering Department, IEEE, 2000.

[24] Timothy J. Ross, *Fuzzy Logic with Engineering Applications*, Second Edition, John Wiley & Sons, Ltd. 2004.

[25] Chi-Tsong Chen, *Analog and Digital Control System Design: transfer-Function, State-Space and Algebraic Methods*, Saunders College Publishing.

[26] International Standard ISO/IEC 14772-1:1997.

[27] The MathWorks, Inc. *Simulink® 3D Animation™*,1984-2009.

[28] Bill Drury, *The Control Techniques Drives and Controls Handbook*, 2nd edition, The Institution of Engineering and Technology, 2009.

[29] Richard C. Dorf and Robert H. Bishop, *Modern Control Systems*, Eleventh edition, Pearson Education International, 2008.

[30] Gene F. Franklin, J. David Powell and Michael L. Workman, *Digital Control of Dynamic Systems*, Addison-Wesley Longman, inc., 1998.

APPENDIX

APPENDIX A: PROOF OF THE STABILITY OF THE B&P SYSTEM

The perfomance of the overall system depends largerly on the outer loop controller. Hence, we verify the stability of the B&P system using Lyapunov's Direct Method.

Assume that there exists a scalar function V of the state x with continuous first order derivative so that:

$$V(x,t) = \frac{1}{2}(s(x,t))^2 \qquad \text{A.1}$$

For conveinance we shall drop the dependence of V and s on x and t in the following discussion. From equation (A.1), V is a candidate Lyapunov function i.e. V is a posetive definate function of the states. Now, if the time derivative $\dot{V}$ is negative semi-definite, then from Lyapunov's Direct Method it can be guaranteed that the equilibruim at the origin is globally asymptotically stable [18].

From equation (4.18), and substituting $e = x - x_d$,we find that:

$$\dot{s} = \ddot{x} - \ddot{x}_d + \lambda\dot{e} \qquad \text{A.2}$$

From equation (2.20), letting $b_o = \frac{5}{7}g$ we get:

$$\dot{s} = b_o u - \ddot{x}_d + \lambda\dot{e} \qquad \text{A.3}$$

Inserting the value of u from equation (4.19) into equation (A.3), we find:

$$\dot{s} = -ksgn(s) \qquad \text{A.4}$$

Taking the time derivative of equation (A.1) along the system trajectories and using the result from equation (A.4), we obtain:

$$\begin{aligned} \dot{V} &= -ksgn(s) \\ &= -k|s| \\ &\leq 0 \end{aligned} \qquad \text{A.5}$$

APPENDIX B: HARDWARE REALIZATION OF THE B&P SYSTEM

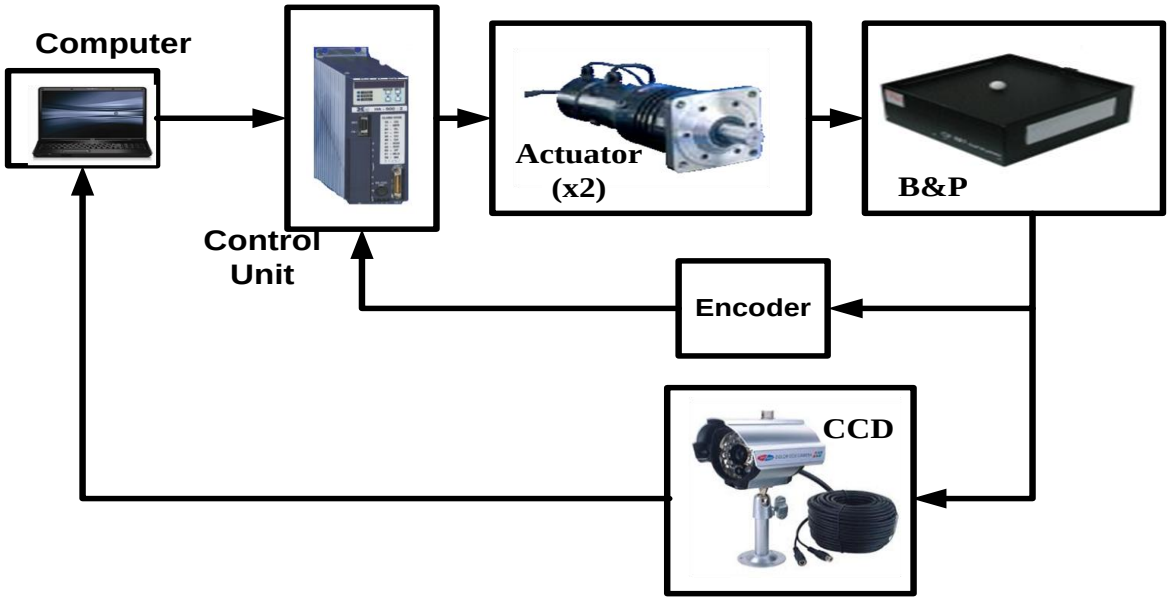


Figure B.1: Hardware Realization of the B&P System

APPENDIX C: DETAILED SIMULINK REPRESENTAION OF THE SYSTEM.

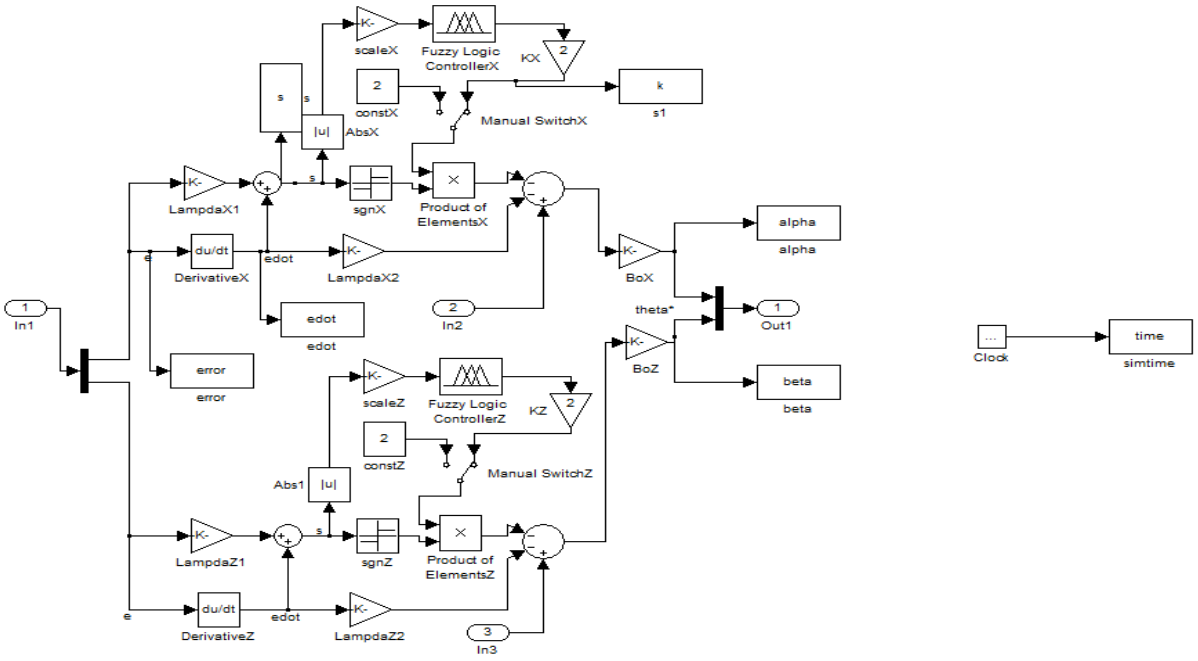


Figure C.1: Realization of FSMC

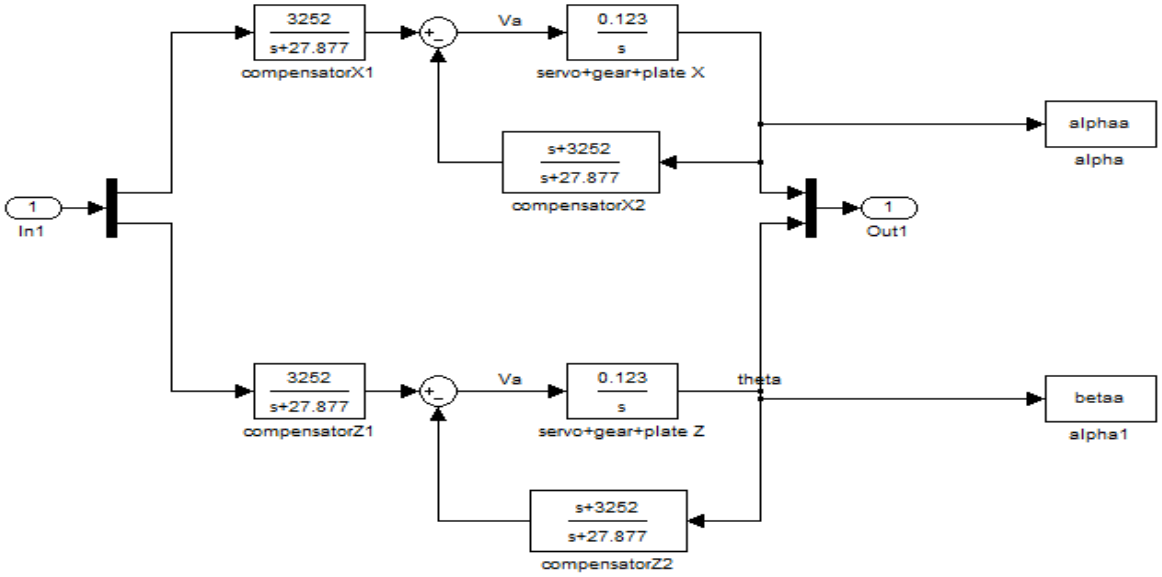


Figure C.2: Realization of Actuator and two-parameter controller

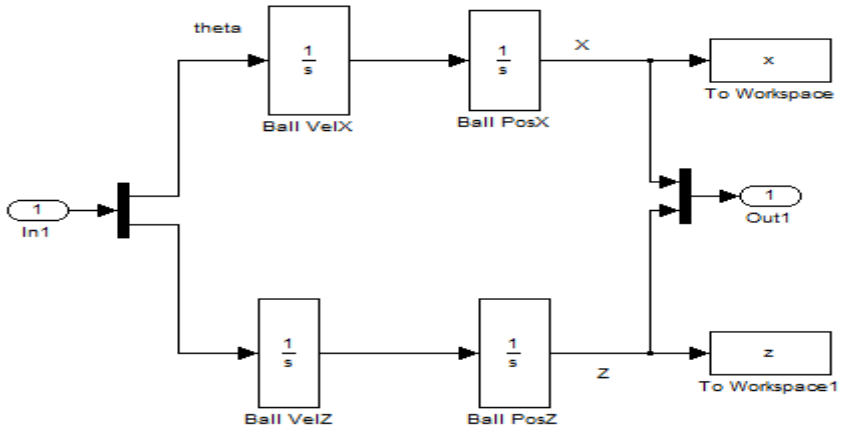


Figure C.3: Realization of the Ball Dynamics

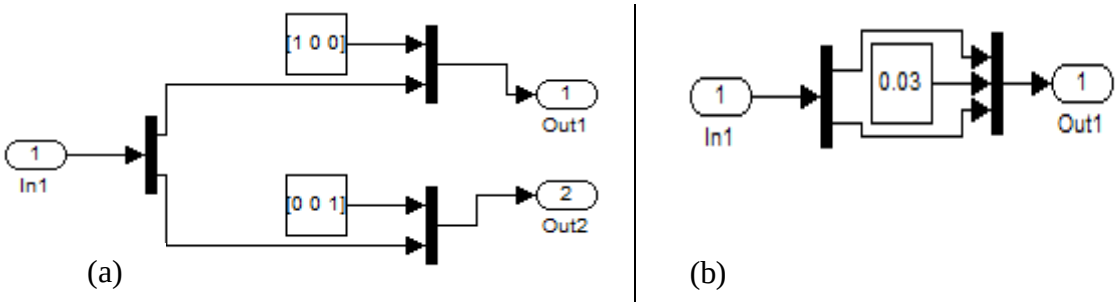


Figure C.4: Realization of 2D to 3D (a) angle converter (b) position converter

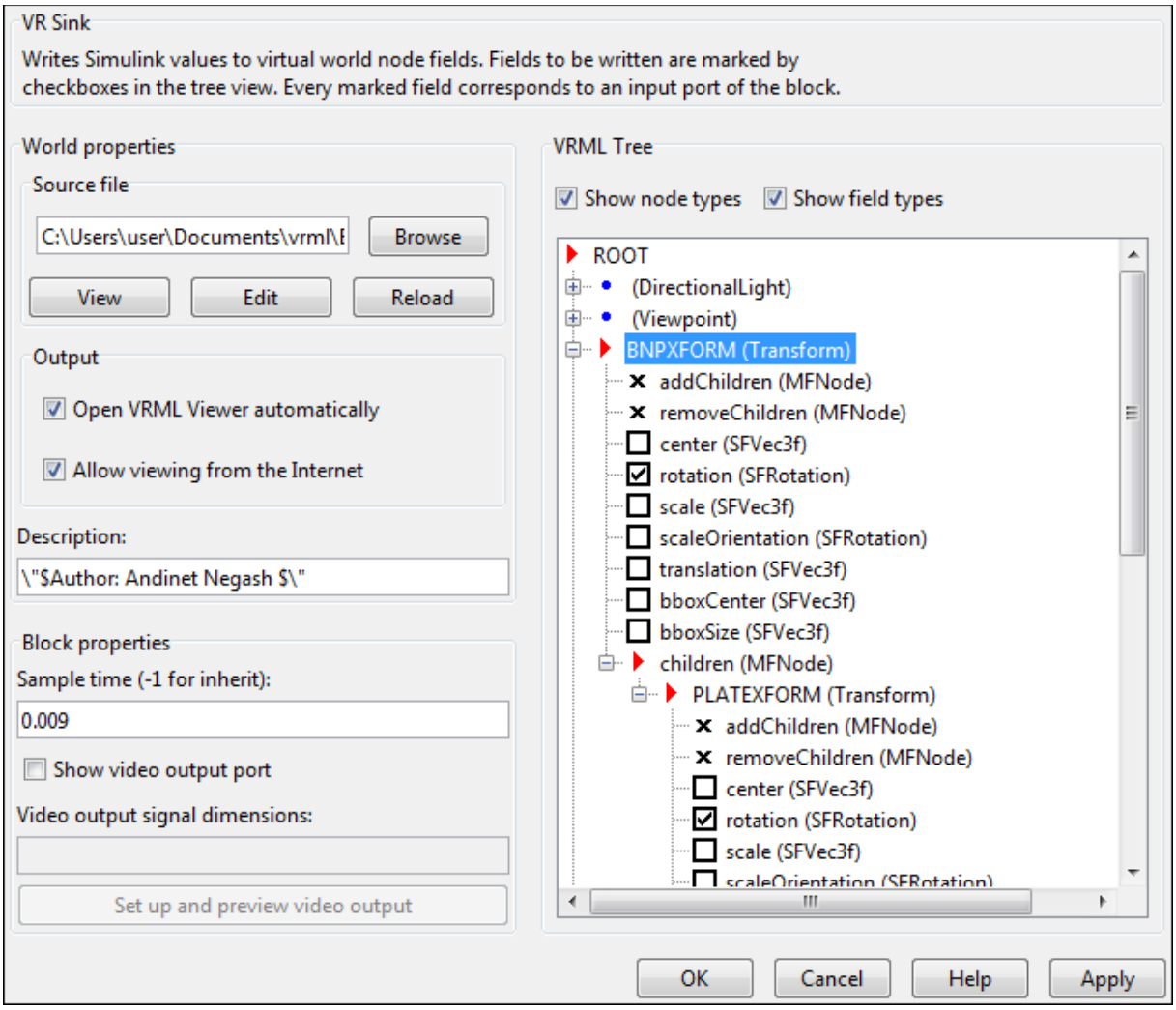


Figure C.5: Realization of Virtual Reality (VR) interface

APPENDIX D: SOURCE CODE OF GENETIC ALGORITHM AND FUZZY LOGIC

```
%%%%
%     Genetic algorithm and Fuzzy Logic Code Development for the Control of
%     the B&P system
%
%     © Andinet Negash Hunde, 2011
%
%     This code generates the plot of optimal membership function 'shapes'
%     of Fuzzy sets using Genetic Algorithm
%
%%% initialize parameter
popsize=100;
g=rand(popsize,1);   %%% random nos. b/n 0 and 1 for 100 populations
p=floor(10^52.*g);   %%% scale and find integer value
m=dec2bin(p,72);     %%% at least 72 bits each
m(:,37:72)=m(:,1:36);%%% to avoid all-zero trailing bits

StrArray=java_array('java.lang.String',popsize);
for p=1:popsize
    StrArray(p)=java.lang.String(m(p,1:72));
end
%%%% convert to MATLAB® string
cellArray=char(StrArray);
%%%% initialize fitness values
n=zeros(popsize);
fit=n(1,:);
score=n(1,:);
for iter=1:1
for p=1:popsize %%% until all population are evaluated do
    pop=cellArray(p,:);
    sum=0;
    scale=[0 0 0 0 0 0 0 0 0 0 0 0];
    for j=1:6 %%% scale the random bits into acceptable range
        if rem(j,2)==0
            scale(j)=0.0+(bin2dec(pop(6*j-5:6*j)))*(0.1/63); %%% input
            scale,sigma
            scale(j+6)=0.0+(bin2dec(pop(6*j+31:6*j+36)))*(-0.9/63);%%%
            output scale,sigma
        else
            scale(j)=(bin2dec(pop(6*j-5:6*j)))*(0.4/63); %%% input
            scale,m(center)
            scale(j+6)=(bin2dec(pop(6*j+31:6*j+36)))*(2/63);%% output
            scale, m(center)
        end
    end
    %%%% apply fuzzy rule
    muf=[0 0 0];
    for k=0.4:-0.01:0 %%% for this input data
        for j=1:3     %%% calculate its belongigness in each fuzzy set
            muf(j)=exp(-0.5*(k-scale(2*j-1))^2/(scale(2*j)^2));
        end
        [mu,mf]=max(muf); %%% apply maxaximum fuzzy inference rule
```

```
            x=scale(2*mf+5)-(scale(2*mf+6))*sqrt(-2*log(mu)); %%% defuzzify
        sum=sum+(x+k*1.25)*(x+k*1.25); %%% find the squared error
    end
    fit(p)=1000000-sum;
end
%%% find the next generation
averPop=mean(fit);
for p=1:popsize
    score(p)=fit(p)/averPop;
end

%%% reproduction
[fitVal,Index]=sort(score,'descend');
Nextgen=cellArray;
for p=1:(0.15*popsize)
    Nextgen(p,:)=cellArray(Index(p),:);
end

%%% crossover
p=0.15*popsize+1;
while(p<=(0.8+0.15)*popsize)
    k=floor(1+71*rand);  %%% indexing number of where to interchange bits
    pop1=floor(1+(popsize-1)*rand);
    pop2=floor(1+(popsize-1)*rand);
    mem1=cellArray(pop1,:); %%% select two populations at random
    mem2=cellArray(pop2,:);
    mem3=mem1(k:72);
    Nextgen(p,:)=strcat(mem1(1:k-1),mem2(k:72));
    Nextgen(p+1,:)=strcat(mem2(1:k-1),mem3);
    p=p+1;
end
%%% mutation
p=(0.8+0.15)*popsize+1;
while (p<=popsize)
    k=floor(1+71*rand);
    pop1=floor(1+(popsize-1)*rand);
    if cellArray(pop1,k)==1 %%% if the 'pop1'th population has a '1' at its
      'k'th position, then it is changed to a '0'
        cellArray(pop1,k)=0;
    else
        cellArray(pop1,k)=1;
    end
    Nextgen(p,:)=cellArray(pop1,:);
    p=p+1;
end
end
%%% Now generate the plot
pop=Nextgen(1,:)
scale=[0 0 0 0 0 0 0 0 0 0 0 0];
    for j=1:6 %%% scale the random bits into acceptable range
        if rem(j,2)==0
            scale(j)=0.1+(bin2dec(pop(6*j-5:6*j)))*(0.3/63);
scale(j+6)=0.1+(bin2dec(pop(6*j+31:6*j+36)))*(0.3/63);
        else
            scale(j)=(bin2dec(pop(6*j-5:6*j)))*(0.4/63);
scale(j+6)=(bin2dec(pop(6*j+31:6*j+36)))*(2/63);
```

```
    end
end
s=0:0.01:0.4;
scale
for j=1:3
    plot(s,exp(-0.5*(s-scale(2*j-1)).^2/(scale(2*j).^2)),'r');
    hold on;
end
s=0:0.01:2;
for j=1:3
    plot(s,exp(-0.5*(s-scale(2*j+5)).^2/(scale(2*j+6).^2)),'b');
    hold on;
end
```

DECLARATION

I, the undersigned, declare that this thesis work is my original work, has not been presented for a degree in this or any other universities, and all sources of materials used for the thesis work have been fully acknowledged.

Andinet Negash Hunde _____

Name Signature

Place: Addis Ababa Institute of Technology, Addis Ababa University, Addis Ababa

Date of Submission: July 22, 2011

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

Professor N.P. Singh _____

Advisor's Name Signature