MODELLING AND OPTIMIZATION OF PID CONTROLLER's PARAMETERS FOR DEEP SPACE ANTENNA POSITIONING SYSTEM USING GENETIC ALGORITHM

By

EMMANUEL ENEMI IDENYI

DEPARTMENT OF MECHANICAL ENGINEERING,
AHMADU BELLO UNIVERSITY, ZARIA,
NIGERIA.

JUNE 2016

MODELLING AND OPTIMIZATION OF PID CONTROLLER's PARAMETERS FOR DEEP SPACE ANTENNA POSITIONING SYSTEM USING GENETIC ALGORITHM

By

EMMANUEL ENEMI IDENYI

(B.Eng. Mechanical Engineering, University of Agriculture, Makurdi. 2010)

P13EGME8064

A DISSERTATION SUBMITTED TO THE SCHOOL OF POSTGRADUATE STUDIES, AHMADU BELLO UNIVERSITY, ZARIA

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF A MASTERS OF SCIENCE (M.Sc) IN MECHANICAL ENGINEERING

DEPARTMENT OF MECHANICAL ENGINEERING,

FACULTY OF ENGINEERING

AHMADU BELLO UNIVERSITY, ZARIA

JUNE, 2016

DECLARATION

I declare that the Dissertation titled "**Modelling and Optimization of PID Controller's Parameters for Deep Space Antenna Positioning System using Genetic Algorithm**" was carried out by me in the Department of Mechanical (Production Option) Engineering, under the supervision of Engr. Dr I. M. Dagwa and Engr. Dr M. O. Afolayan, Faculty of Engineering, Ahmadu Bello University, Zaria, Nigeria.

The information derived from the literature is duly acknowledged and a list of references provided. No part of this dissertation was previously presented for the award of a degree at this or any other Institution.

Idenyi, E. E

_____	_____	_____
Name	Signature	Date

CERTIFICATION

This Dissertation titled "**Modelling and Optimization of PID Controller's Parameters for Deep Space Antenna Positioning System using Genetic Algorithm"** by Emmanuel Enemi IDENYI (P13EGME8064) meets the regulations governing the award of the degree of Master of Science in Mechanical Engineering of the Ahmadu Bello University, and is approved for its contribution to knowledge and literary presentation.

Engr. Dr. I. M. Dagwa

Chairman, Supervisory Committee	Signature	Date

Engr. Dr. M. O. Afolayan

Member, Supervisory Committee	Signature	Date

Dr. M. Dauda

Head of Department	Signature	Date

Prof. K. Bala

Dean, School of Postgraduate Studies	Signature	Date

DEDICATION

This dissertation is dedicated to the Most High God, the giver and custodian of life and the one who bestows knowledge. It is also dedicated to those who in one way or the other are involved in space technology.

ACKNOWLEDGEMENTS

My profound gratitude goes to my supervisor, Engr. Dr I. M. Dagwa, for his constant motivation and support throughout my postgraduate studies. I am indebted to him for reshaping the problem and providing insights towards solving them. I truly appreciate and value his esteemed guidance and encouragement from the beginning to the end of this Dissertation. My sincere thanks also go to my co-supervisor, Engr. Dr M. O. Afolayan, for his unending tolerance, patience, constant encouragement, guidance and expertise as a control expert; without which this work would not have been this special and technical in terms of content and richness. I will never forget the long hours we spent, despite his tight schedule, discussing various ideas and methods. Their respective constructive criticisms ensured the nitty-gritty of the work.

My unending gratitude goes to my parents, Mr and Mrs K. A. Idenyi, for giving me life through birth and in education. I also thank the family of Mr and Mrs Musa James for their constant encouragement and motivation for me to have a postgraduate education. I sincerely thank the entire management staff of National Space Research and Development Agency (NASRDA), Abuja, of which am privileged to be a member of staff, for the opportunity given to me to have a postgraduate education and to undergo a research in one of the challenges faced by the Agency. My heartfelt appreciation goes to Engr. S. Hussein of NASRDA for being one of the originators of the problem leading to this research.

My sincere gratitude goes to Prof S. Y. Aku for the privilege to work with him on this research earlier and also to the Head of Department, Dr M. Dauda, for the enabling environment given to me to undertake this research. My sincere thanks also go to Prof D. S. Yawas and Dr F. O. Anafi for their constructive criticisms during my internal defence which gave this work a meaning. Not to be left out are my colleagues, Mr Malomo and Mr Salawudeen for their great contributions to the work especially in the optimization algorithm (GA); and also to Engr.

Valentine of the Federal Ministry of Science and Technology, Abuja for tutoring me on the Bond-Graph model.

You all made this work a reality and may the Almighty God reward all of you abundantly. Amen

Abstract

Proportional-Integral-Derivative (PID) controllers have been widely used in process industry for decades, from small industry to high technology industry, but they still remain poorly tuned by use of conventional tuning methods like the Zeigler-Nichols method. In this work, PID controller's parameters for deep space antenna positioning system were optimized using Genetic Algorithm (GA). Using genetic algorithm, the tuning of the controller results in the optimum controller being evaluated for the system every time. The system was modelled using Bond-Graph in 20Sim environment and the PID Controller was optimized using GA in Matlab/Simulink environment in order to get the optimum value for its parameters. Simulation result shows that the performance of the optimized PID Controller using Genetic Algorithm (GA) for deep space antenna positioning system at response values of 2.2412sec rise time, 2.9861sec settling time and 0% overshoot and undershoot is better than the conventionally, Zeigler-Nichols method, tuned Controller at response values of 0.8568sec rise time, 9.2289sec settling time, 66.3812% overshoot and 23.1264% undershoot; thereby comparing the work at an amplifier gain value of 100. Results for different amplifier gain values also show that the system response at an amplifier gain of 250 produced the best response in terms of rise time, settling time and overshoot but has a problem of peaking in its transient state characteristics.

TABLE OF CONTENTS

LIST OF PLATE

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

0	Zero junction
1	One junction
BG	Bond Graph
C	Compliance
Const	Constant
e	Effort
f	Flow
GA	Genetic algorithm
GY	Gyrator
I	Inertia
LQR	Linear Quadratic Regulator
MGY	Modulated Gyrator
MR	Modulated Resistance
MSe	Modulated Source effort
MSf	Modulated Source flow
MTF	Modulated Transformer
p.e	port of effort
p.f	port of flow
PID	Proportional Integral and Derivative

R	Resistance
Se	Source effort
Sf	Source flow
TF	Transformer

DEFINITIONS

Rise time, T_r: The time required for the waveform to go from 0.1 of the final value to 0.9 of the final value.

Peak time, T_p: The time required to reach the first, or maximum, peak.

Percent overshoot, %OS: The amount that the waveform overshoots the steady state, or final, value at the peak time, expressed as a percentage of the steady-state value.

Settling time, T_s: The time required for the transient's damped oscillations to reach and stay within ±2% of the steady-state value.

LIST OF APPENDIX

CHAPTER ONE

INTRODUCTION

1.1 Background

Antennas are electrical devices which convert electric power into radio waves, and vice versa (Gawronski, 2008). They communicate with spacecraft by sending commands (uplink) and receiving information (downlink) from it (Gawronski, 2008). An antenna tracking (the act or process of following the trail) a satellite must keep the satellite well within its beam-width in order not to lose track (Nise, 2006). Therefore, in order to ensure this due to Earth's rotation, the antenna as shown in Plate I is continuously positioned with the aid of a controller and a drive mechanism. This implies that suitable and efficient positioning of antenna structure will enhance signal clarity, wider coverage area and satisfactory reception of radiated signal (Agubor *et al*., 2013). However, for frequencies in the Ka-band, there is need for a very efficient tracking (Gawronski, 2008). This requirement is a driver for the upgrade of control systems of antennas.

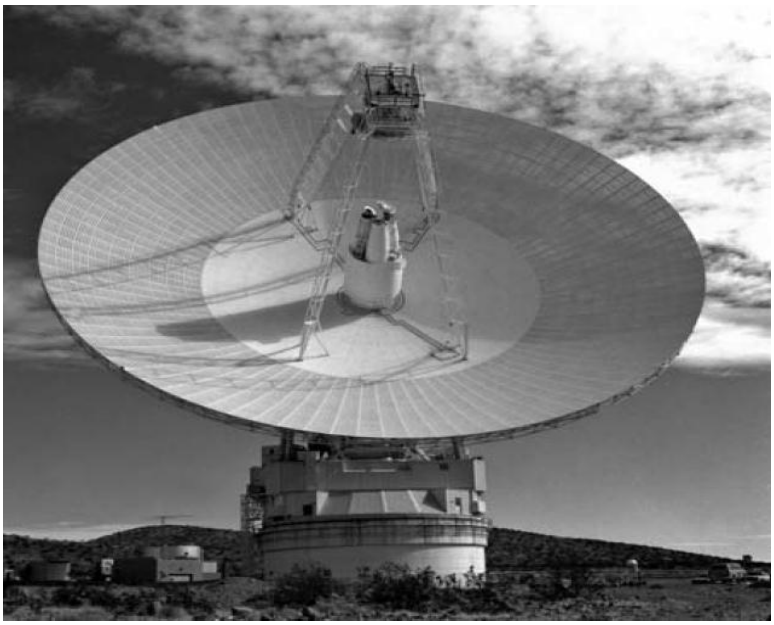


Plate I: A deep space antenna (Source: Gawronski, 2008)

The antenna dish rotates with respect to the horizontal axis while the whole structure rotates on a circular track with respect to the vertical axis. The position of antenna is controlled by using gears and feedback potentiometer. Antenna positioning is also controlled by using some controllers (Chishti *et al.*, 2014). A controller aims at minimizing the error between a measured process variable of the controlled system and a reference, by calculating the error and generating a correction signal to the system from the error (Pillai *et al*., 2013). For getting better response for antenna positioning system, several controllers like Proportional-Integral-Derivative (PID) Controller, Linear Quadratic Regulatory (LQR) Controller, Fuzzy Logic Controller (FLC) etc have been proposed and used.

The Proportional-Integral-Derivative (PID) Controller is widely used in most industrial processes due to their simplicity of operation, ease of design, inexpensive maintenance, low cost, and effectiveness for most linear systems but they still remain poorly tuned. Conventional technique like Zeigler-Nichols method does not give an optimized value for PID controller's parameters (Pillai *et al*., 2013).

In this work, the PID controller parameters for antenna positioning system were optimized using Genetic Algorithm (GA). GA is a stochastic global search method that replicates the process of evolution (Pillai *et al*., 2013). The advantage of GA over other popular and efficient optimization algorithm such as Artificial Neural Networks and Fuzzy Logic is its high convergence speed of execution (Zhang *et al*., 2009).

The overall system was modelled using bond graphs. Bond graphs are a domain-independent graphical technique that exploits the fundamental laws of energy to model a system very close to reality i.e. nonlinear system (Broenink, 1999). They have certain advantages over network models. In particular, bond graph elements exist which allow multiport elements to be modelled explicitly, whereas network models, even of simple two-port elements, are awkward to draw and manipulate (Wellstead, 1979). From a practical viewpoint the relatively compact nature of bond graphs commends the technique as the basis for computer-aided modelling. This, coupled

with the development of low-cost, high-speed interactive computer systems, distinguishes the bond graph approach as potentially the most useful of the systematic modelling techniques (Wellstead, 1979)

1.2 Statement of the Problem

The conventional optimization method of Proportional-Integral-Derivative (PID) Controller parameters cannot meet the requirements of control performance in positioning deep space antennas control system. The problems usually encountered in deep space antenna research are:

1. Due to earth's rotation, there is non-continuous tracking of signal by the deep space antenna. As a result of this, the deep space antennas are continuously positioned with the aid of a drive mechanism and a controller (Gawronski, 2008).
2. Proportional-Integral-Derivative (PID) controllers have been widely used in process industry for decades, from small industry to high technology industry, but they still remain poorly tuned by use of conventional tuning methods like the Zeigler-Nichols method (Pillai *et al*., 2013).
3. The popular and efficient optimization algorithm such as Artificial Neural Networks and Fuzzy Logic have some disadvantages, such as premature convergence and low convergence speed of execution (Zhang *et al*., 2009). This implies that GA has a higher speed of execution as compared to them.

1.3 Aim and Objectives

The aim of this research is to optimize the Proportional-Integral-Derivative (PID) Controller parameters for optimum performance and positioning of deep space antenna control system.

The specific objectives are:

1. To develop a nonlinear mathematical model of deep space antenna system using Bond-Graph in 20Sim environment.

2. To develop a Genetic Algorithm (GA) to tune the Proportional-Integral-Derivative controller in order to achieve specifications and stability parameters in Matlab.
3. To analyse the system and simulate it using Matlab.
4. To validate the work by comparing it with previous and latest work done in this research field.

1.4 Significance of Research/Justification

There are many areas in Nigeria and globally where antennas and its optimization prove to be very useful. Some of these are:

1. The application of antennas and satellite with its optimum performance and upgrade prove to be very useful as part of the space policy and activities of the Federal Republic of Nigeria, implemented through National Space Research and Development Agency (NASRDA) with a mission to pursue the research and development of space science and technology which ensures that Nigeria vigorously pursues the attainment of space capabilities as an essential tool for its socio-economic development and the enhancement of the quality of life of its people.
2. There is much need for constant and efficient communication using optimized antennas since the deployment of communication infrastructure in Nigeria in recent times have been massive. This is as a result of the ever increasing demand of wireless mobile services.
3. There is also more demand for the use of antennas and it's upgrade for efficient broadcasting of information due to noticeable increase in the establishment of radio and television stations since this sector was deregulated.

1.5 Scope of Work

The scope of this research is limited to:

1. Open-loop model design of the system.

2. Closed-loop model design of the system with a PID Controller.
3. Using Genetic Algorithm built into Matlab to determine the optimized parameters of the PID controller.
4. Using Matlab/Simulink to simulate the deep space antenna control system.

1.6 Present Research

This present research efforts focuses on optimizing the parameters of the PID controller and the performance is evaluated using some of the following criteria:

1. Settling time of a small step response.
2. Overshoot of a small step response.
3. Overshoot of a large step response.
4. Steady-state error.
5. Amplitude and settling time of a disturbance step.
6. Magnitude of the disturbance transfer function.
7. Phase and gain stability margins.

CHAPTER TWO

LITERATURE REVIEW

2.1 Overview of Fundamental Concepts

In this section, concepts fundamental to the research such as Deep Space Antenna Control System, PID Controller, Genetic Algorithm and Bond Graphs are presented.

2.1.1 Deep space antenna position control system

A position control system converts a position input command to a position output response (Chishti *et al*., 2014).This consists of a position error sensing module and a position error correcting module (Sheth and Gonsai, 2012). A deep space antenna position control system is shown below with a more detailed layout, schematic and a functional block diagram (Nise, 2006).

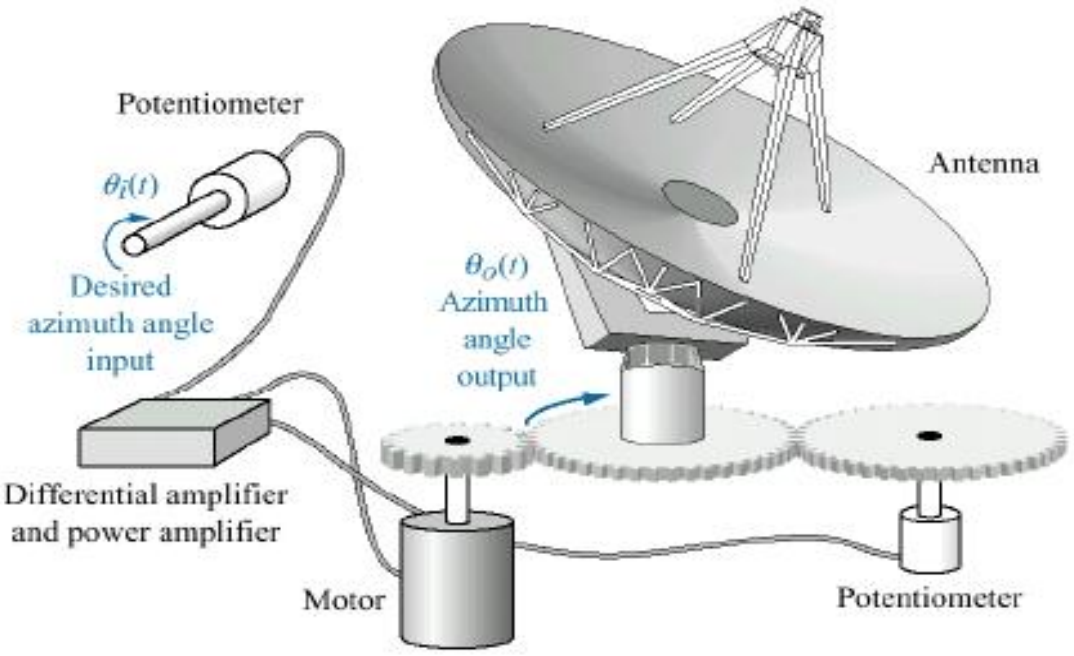


Figure 2.1: Antenna azimuth position control system detailed layout (Source: Nise, 2006)

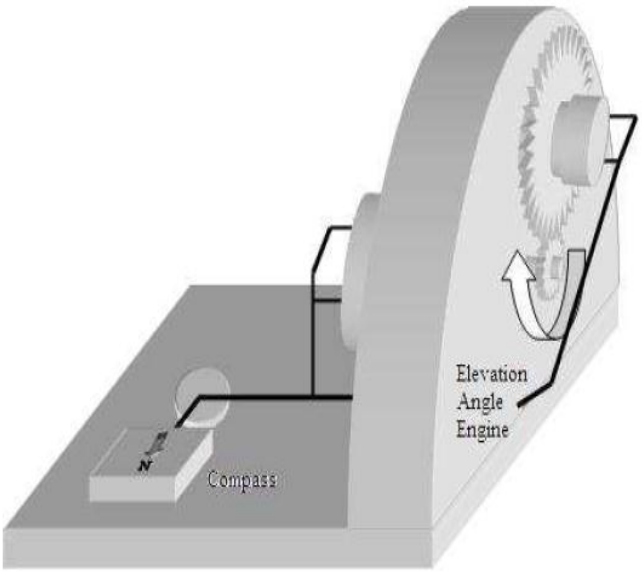


Figure 2.2: Antenna dish elevation position control system detailed layout (Source: Nise, 2006)

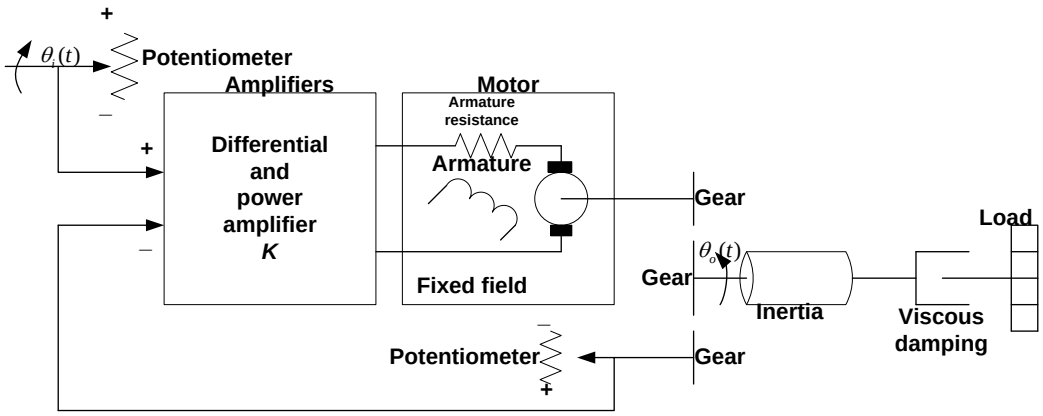


Figure 2.3: Antenna azimuth position control system schematic (Source: Nise 2006)

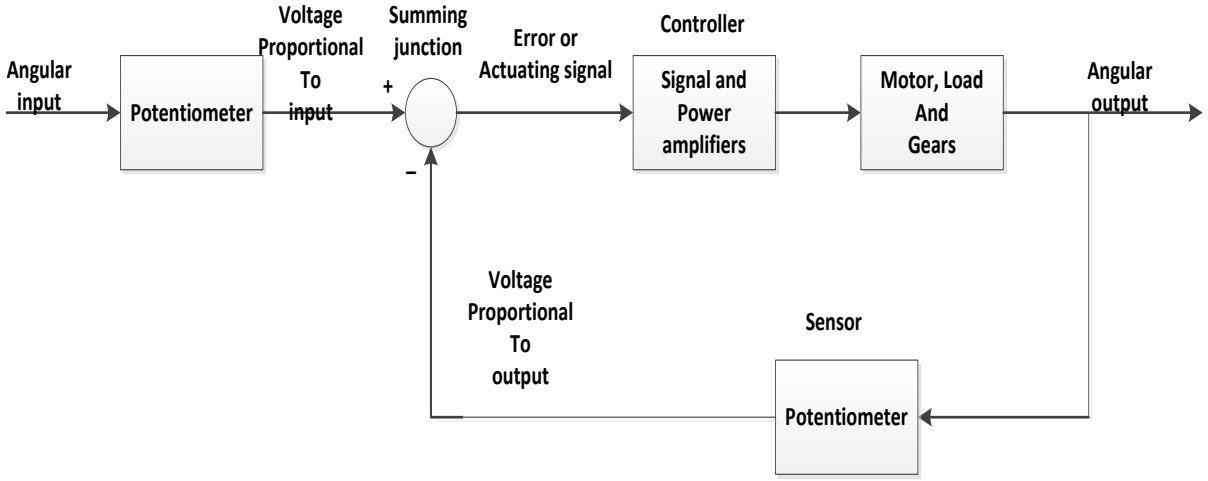


Figure 2.4: Antenna azimuth position control system block diagram (Source: Nise, 2006)

The quality of this antenna drives (hardware), the control algorithm (software), and the physical structure of the antenna (in terms of thermal deformations, gravity distortions, encoder mounting, and wind gusts) all influence pointing precision, and create the challenging task of remaining within the required pointing-error budget (Sheth and Gonsai, 2012).

The purpose of this system is to have the azimuth angle output of the antenna, $\Theta_o(t)$, following the input angle of the potentiometer, $\Theta_i(t)$.

The block diagram of the system as shown in Figure 2.4 describes how the system works. The input command is an angular displacement. The potentiometer converts the angular displacement into a voltage. Similarly, the output angular displacement is converted to a voltage by the potentiometer in the feedback path. After that, differential amplifier check how much the obtained signal is different from the given signal and also find the error. The signal and power amplifiers boost the difference between the input and output voltages. This amplified actuating signal drives the system (Nise, 2006). The system normally operates to drive the error to zero. When the input and output match, the error will be zero, and the motor will not turn. Thus, the motor is driven only when the output and the input do not match. The greater the difference between the input and the output, the larger the motor input voltage, and

the faster the motor will turn (Nise, 2006). This system uses a fixed field DC servo motor (Okumus *et al*., 2012).

There are 5 subsystems of the overall antenna position system, each with its associated transfer function as shown below (Liu, 2009). These are described both in the schematic (Figure 2.3) and the given block diagram (Figure 2.4) (Nise, 2006). Table 2.1 define all the variables used in the antenna position system (Liu, 2009).

Table 2.1: Variables used in the antenna position system (Liu, 2009)

Schematic Parameters			
Parameter	**Definition**	**Configuration 1**	**Configuration 2**
V	Voltage across Potentiometer [Volts]	10	10
N	Turns of potentiometer	10	1
K	Preamplifier gain	-	1
K_1	Power Amplifier Gain	100	100
a	Power Amplifier pole	100	100
R_a	Motor Resistance [ohms]	8	5
J_a	Motor Inertial constant [kg-m2]	0.02	0.05
D_a	Motor Dampening constant [N-m s/rad]	0.01	0.01
K_b	Back EMF constant [V-s/rad]	0.5	1
K_t	Motor Torque constant [N-m/A]	0.5	1
N_1	Gear teeth	25	50
N_2	Gear teeth	250	250
N_3	Gear teeth	250	250
J_L	Load inertial constant [kg-m2]	1	5
D_L	Load dampening constant [N-m s/rad]	1	3
Block Diagram Parameters			
Parameter	**Definition**		
K_{pot}	Potentiometer gain	0.318	3.18
K	Preamplifier gain	-	1
K_1	Power Amplifier gain	100	100
a	Power Amplifier pole	100	100
K_m	Motor and load gain	2.083	0.8
a_m	Motor and load pole	1.71	1.32
K_g	Gear ratio	0.1	0.2

Configurations 1 and 2 are the values of the parameters used in modelling an antenna positioning system.

Subsystem 1 and 5.

The input and feedback potentiometer each have an associated transfer function, in the form of a gain. The potentiometer changes the input angle $\theta_i(s)$, to a voltage, $V_i(s)$. This ratio is described by the value K_{pot}. as given in Table 2.1 above. This value is computed by equation 2.1. The value of this gain is determined by the voltage applied to the potentiometer and the number of turns the potentiometer is built for, both of these values are given in Table 2.1 (Liu, 2009).

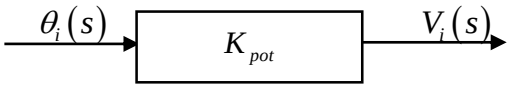
$$\frac{V_i(s)}{\theta_i(s)} = K_{pot_i} = \frac{V}{N\pi} = \frac{10}{1*\pi} = 3.18 \qquad (2.1)$$

Where $V_i(s)$ is voltage from the input potentiometer [V]

$\theta_i(s)$ is the input angle [rad]

K_{pot_i} is the potentiometer gain

N is the turns of the potentiometer



Potentiometer

Figure 2.5: Block diagram of a potentiometer

Subsystem 2.

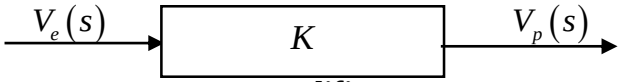
The purpose of the preamplifier is to amplify the input signal voltage for input into the power amplifier. The Preamplifier is also modelled by a gain that can be specified by the design engineer to achieve a desired output. The preamplifier is a system in which the input voltage is amplified by some gain K and output as a voltage. The resulting equation therefore is quite simple, as shown in equation 2.2.

$$\frac{V_p(s)}{V_e(s)} = K \qquad (2.2)$$

Where $V_p(s)$ is the output voltage of the preamplifier [V]

$V_e(s)$ is the voltage after the summation [V]

K is the preamplifier gain



Preamplifier

Figure 2.6: Block diagram of a preamplifier

Subsystem 3.

The third subsystem is a power amplifier which takes the output voltage from the preamplifier and converts it into a voltage that is useable by the motor. The power amplifier type is given in the design schematic and the given block diagram. Also the value of $\mathbf{K_1}$ and **a** in the transfer function are given in the configuration data shown in Table 2.1 above.

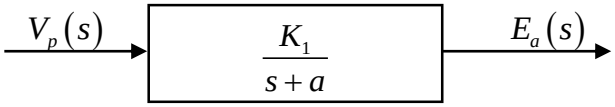
$$\frac{E_a(s)}{V_p(s)} = \frac{K_1}{s+a} \tag{2.3}$$

Where $E_a(s)$ is the output voltage of the power amplifier [V]

$V_p(s)$ is the input voltage of the power amplifier from the preamplifier [V]

K_1 is the power amplifier gain

a is the power amplifier pole



Power Amplifier

Figure 2.7: Block diagram of a power amplifier

Subsystem 4.

After the Power Amplifier is the motor, attached to the gears and load, which in this case is an antenna. All of these items must be considered when computing the transfer function of the resulting mechanical system. It is assumed that the motor is an armature controlled DC servo motor. This is determined by the fact that the motor has a fixed field, which also simplifies the control of the motor.

To derive the transfer function of the subsystem one must find the Kirchoff's Voltage Law (KVL) equation relating the input voltage to the motor to the output position of the armature (Liu, 2009). Figure 2.5 shows the general circuit of a motor.

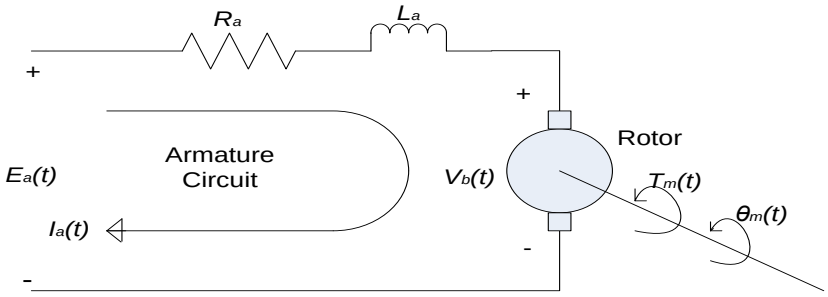


Figure 2.8: Circuit of a motor (Source: Nise, 2006)

The resulting KVL equation is shown in equation 2.4. We are only concerned with the input voltage, and we have no information regarding the input current, thus it would be useful to replace the current term I_a with its equivalent Torque term, shown in equation 2.5. Definition of parameters are given in Table 2.1 above.

$$R_a I_a(s) + L_a s I_a(s) + V_b(s) = E_a(s) \tag{2.4}$$

$$T_m(s) = K_t I_a(s) \therefore I_a(s) = \frac{T_m(s)}{K_t} \tag{2.5}$$

Where R_a is the motor resistance [ohms]

I_a is the circuit current [A]

L_a is the motor inductance [H]

$V_b(s)$ is the voltage across the rotor (back EMF) [V]

$E_a(s)$ is the voltage across the motor [V]

$T_m(s)$ is the motor toque [N.m]

K_t is the motor torque constant [N-m/A]

This results in an equation with no current term, but instead a motor torque. This torque term can be replaced by a term relating torque to motor speed, position, inertia and dampening. One can replace the back EMF term, V_b, with a term that relates back EMF to the derivative of

speed, which is position, a term we will need to create the transfer function. These are both shown in equations 2.6 and 2.7.

$$V_b(s) = K_b s\theta_m(s) \tag{2.6}$$

$$T_m(s) = (Js^2 + D_m s)\theta_m(s) \tag{2.7}$$

K_b is the back EMF constant [V-s/rad]

$\theta_m(s)$ is the angle of rotation of the rotor [rad]

J is the inertial constant [Kg-m2]

D_m is the dampening constant [N-m s/rad]

Thus replacing the corresponding variables with their equivalents into equation 2.4, and simplifying creates equation 2.8.

$$\frac{(Js^2 + D_m s)(R_a + L_a s)\theta_m(s)}{K_t} + K_b s\theta_m(s) = E_a(s) \tag{2.8}$$

This is making use of the assumption that this is a fixed field motor, which makes K_b and K_t equal. Both of these values are given; for configuration two in Table 2.1 above, they are 1. Then pulling $\theta_m(s)$ to the outside yields equation 2.9.

$$\left[\frac{(Js^2 + D_m s)(R_a + L_a s) + K_b K_t s}{K_t}\right]\theta_m(s) = E_a(s) \tag{2.9}$$

Definition of terms are given in Table 2.1

Assuming that $R_a >> L_a$ one can further simplify to equation 2.10.

$$\frac{\theta_m(s)}{E_a(s)} = \frac{\frac{K_t}{JR_a}}{s\left(s + \frac{D_m R_a + K_b K_t}{JR_a}\right)} \tag{2.10}$$

The damping and inertial components of the system are connected to the motor through a set of gears. This changes their effective values as seen by the motor and must be compensated for. This can be done using equation 2.12 and 2.13. The gear ratios is shown in equation 2.11.

$$K_g = \frac{N_1}{N_2} = 0.2 \tag{2.11}$$

$$J = J_a + J_L(K_g)^2 = 0.25 \tag{2.12}$$

$$D_m = D_a + D_L(K_g)^2 = 0.13 \tag{2.13}$$

K_g is gear ratio

N is gear teeth

J_a is the motor inertial constant [kg-m2]

J_L is the load inertial constant [kg-m2]

D_a is motor dampening constant [N-m s/rad]

D_L is load dampening constant [N-m s/rad]

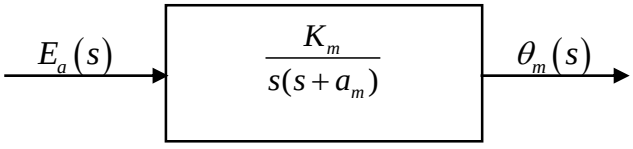
Equation 2.10 is in the same form as the given transfer function for the motor and load, thus we can relate the variables K_m, and a_m. This, along with their computed values is shown in equations 2.14 and 2.15.

$$K_m = \frac{K_t}{JR_a} = \frac{1}{0.25*5} = 0.8 \tag{2.14}$$

$$a_m = \frac{D_m R_a + K_b K_a}{JR_a} = \frac{0.13*5+1*1}{0.25*5} = 1.32 \tag{2.15}$$

Where K_m is motor and load gain

a_m is motor and load pole



Motor, Load and Gears

Figure 2.9: Block diagram of motor, load and gears

2.1.1.1 Block diagram reduction for the antenna azimuth position control system

Figure 2.10 shows how to reduce the subsystems of the antenna azimuth position control system to a single, closed-loop transfer function in order to analyse and design the transient response characteristics. This is done using block diagram reduction (Nise, 2006).

The steps taken to reduce the block diagram to a single, closed-loop transfer function relating the output angular displacement to the input angular displacement are shown in Figure 2.10 (a-d). In Figure 2.10(b), the input potentiometer was pushed to the right past the summing junction, creating a unity feedback system. In Figure 2.10(c), all the blocks of the forward transfer function are multiplied together, forming the equivalent forward transfer function. Finally, the feedback formula is applied, yielding the closed-loop transfer function in Figure 2.10(d).

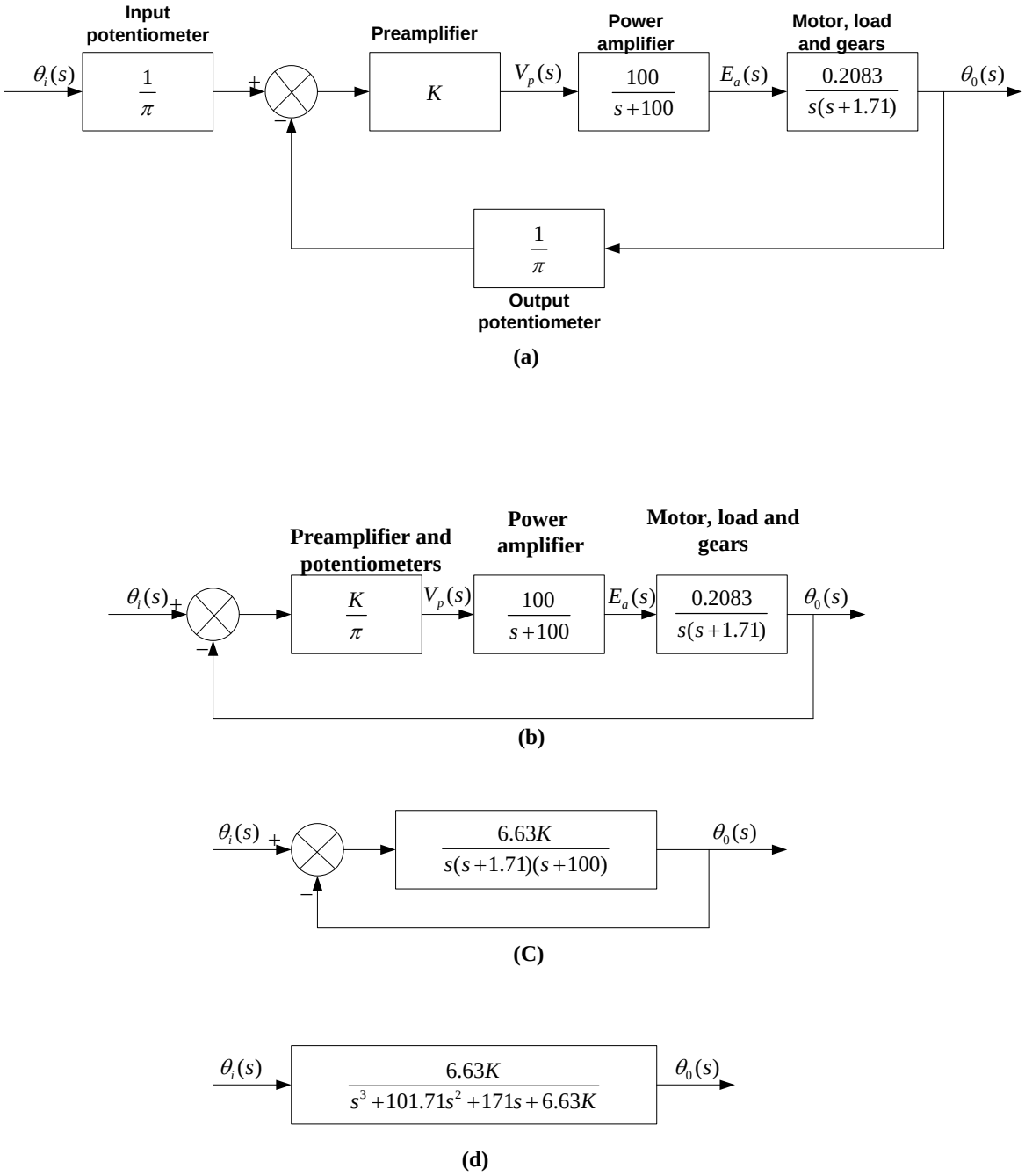


Figure 2.10: a. original; b. pushing input potentiometer to the right past the summing junction; c. equivalent forward transfer function; d. final closed-loop transfer function (Source: Nise, 2006)

Gain value "K" represents the preamplifier block. The value of preamplifier gain "K" can be found out for stable system by utilizing the Routh-Herwitz criterion. According to this criterion, system will give stable response the value of gain "K" is in the range 0-262.3. If the gain of the signal amplifier, K, is increased, then for a given actuating signal, the motor will be driven harder. However, the motor will still stop when the actuating signal reaches zero, that is, when

the output matches the input. The difference in the response, however, will be in the transients. Since the motor is driven harder, it turns faster toward its final position. Also, because of the increased speed, increased momentum could cause the motor to overshoot the final value and be forced by the system to return to the commanded position. Thus, the possibility exists for a transient response that consists of *damped oscillations* (that is, a sinusoidal response whose amplitude diminishes with time) about the steady-state value if the gain is high. The responses for low gain and high gain are shown in Figure 2.11 (Ogata, 2007).

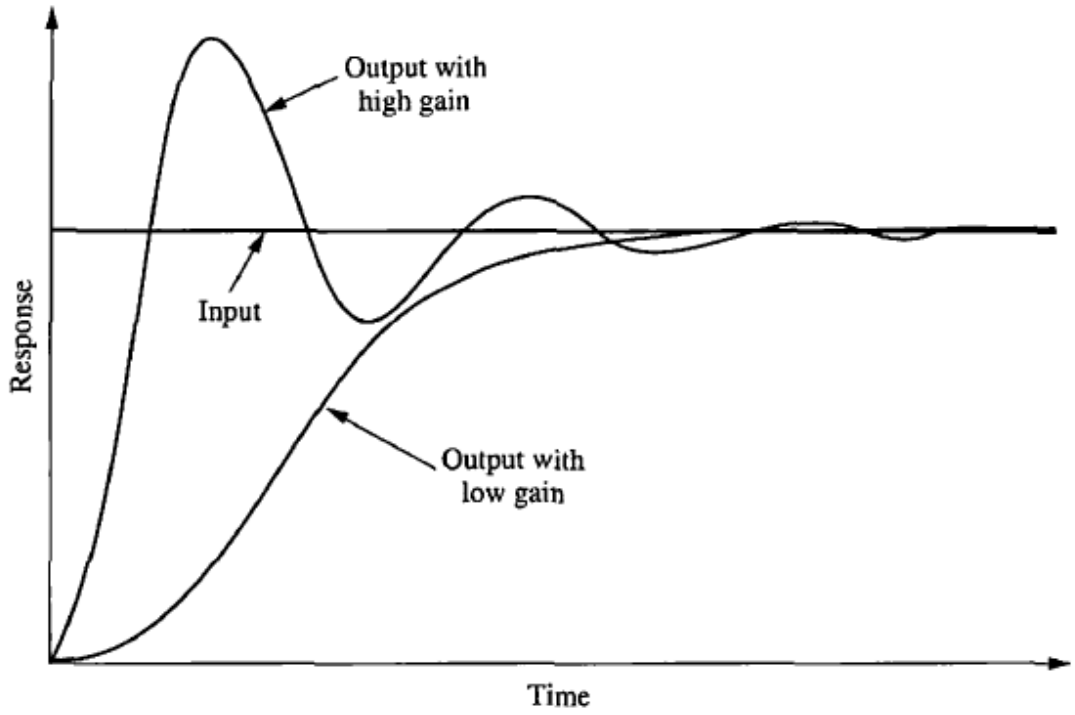


Figure 2.11: Response of a position control system, showing effect of high and low controller gain on the output response (Source: Ogata, 2007)

Steady-state error is the difference between the input and the output after the transients have effectively disappeared. The definition holds equally well for step, ramp, and other types of inputs. Typically, the steady-state error decreases with an increase in gain and increases with a decrease in gain. Figure 2.11 shows zero error in the steady-state response; that is, after the transients have disappeared, the output position equals the commanded input position. In some systems, the steady-state error will not be zero; for these systems, a simple gain adjustment to regulate the transient response is either not effective or leads to a trade-off between the desired transient response and the desired steady-state accuracy (Nise, 2006).

To solve this problem, a controller with a dynamic response, such as an electrical filter, is used along with an amplifier. With this type of controller, it is possible to design both the required transient response and the required steady-state accuracy without the trade-off required by a simple setting of gain. However, the controller is now more complex. The filter in this case is called a compensator (Ogata, 2007).

In summary, the design objectives and the system's performance revolve around the transient response, the steady-state error, and stability. Gain adjustments can affect performance and sometimes lead to trade-offs between the performances criteria. Controllers can often be designed to achieve performance specifications without the need for trade-offs (Nise, 2006).

2.1.2 PID controller

A controller aims at minimizing the error between a measured process variable of the controlled system and a reference, by calculating the error and generating a correction signal to the system from the error (Pillai *et al*., 2013). An automatic controller compares the actual value of the plant output with the reference input (desired value), determines the deviation and produces a control signal that will reduce the deviation to zero or to a small value. The manner in which the automatic controller produces the control signal is called the *control action* (Ogata, 2007).

The Proportional Integral Derivative (PID) controller has several important functions: it provides feed-back, it has the ability to eliminate steady state offsets through integral action and it can anticipate the future through derivative action (Astrom *et al*., 1995). It is widely used in most industrial processes due to their simplicity of operation, ease of design, inexpensive maintenance, low cost, and effectiveness for most linear systems (Franklin *et. al*., 2002). It also have the advantages of simple structure, good stability, and high reliability (Pillai *et al*., 2013). Figure 2.12 explains an automatic controller.

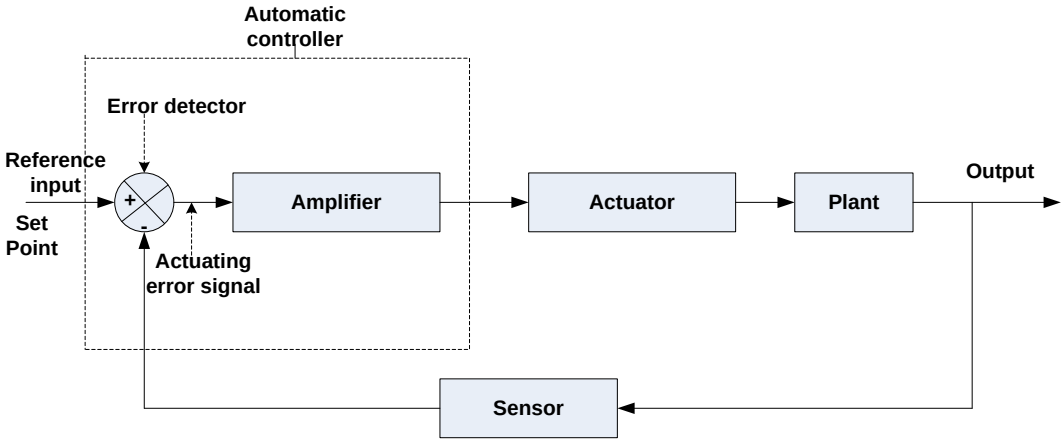


Figure 2.12: A block diagram of an industrial control system, which consists of an automatic controller, an actuator, a plant and a sensor (measuring element) (Source: Ogata, 2007).

The controller detects the actuating error signal, which is usually at a very low power level and amplifies it to a sufficiently high level. The output of an automatic controller is fed to an actuator such as an electric motor. The actuator is a power device that produces the input to the plant according to the control signal so that the output signal will approach the reference input signal. The sensor or measuring element is a device that converts the output variable into another suitable variable such as a displacement, pressure or voltage that can be used to compare the output to the reference input signal. This element is in the feedback path of the closed loop system. The set point of the controller must be converted to a reference input with the same units as the feedback signal from the sensor or measuring element (Pillai *et al*., 2013).

The combination of proportional control action, integral control action and derivative control action is termed proportional-plus-integral-plus-derivative control action.

2.1.2.1 Proportional control action

For a controller with proportional control action, the relationship between the output of the controller *u(t)* and the actuating error signal *e(t)* is

$$u(t) = K_p e(t) \qquad 2.16$$

or in Laplace transformed quantities

$$\frac{U(s)}{E(s)} = K_p \tag{2.17}$$

Where K_p is termed the proportional gain.

The proportional controller is essentially an amplifier with an adjustable gain (Ogata, 2007).

2.1.2.2 Integral control action

In a controller with integral control action, the value of the controller output *u(t)* is changed at a rate proportional to the actuating error signal *e(t)*. That is,

$$\frac{du(t)}{dt} = K_i e(t) \tag{2.19}$$

$$u(t) = K_i \int_0^t e(t)dt \tag{2.20}$$

Where K_i is the integral gain, an adjustable constant. The transfer function of the integral controller is

$$\frac{U(s)}{E(s)} = \frac{K_i}{s} \tag{2.21}$$

If the value of *e(t)* is doubled, then the value of *u(t)* varies twice as fast. In the proportional control of a plant whose transfer function does not possess an integrator 1/s, there is a steady state error or offset in the response to a step input. Such an offset can be eliminated if the integral control action is included in the controller (Ogata, 2007).

2.1.2.3 Derivative control action

The derivative control action is given by

$$u(t) = K_d \frac{de(t)}{dt} \tag{2.22}$$

and the transfer function is

$$\frac{U(s)}{E(s)} = T_d s \qquad 2.23$$

Where K_d is a constant called the *derivative gain*. When derivative control action is added to a proportional controller, it provides a means of obtaining a controller with high sensitivity. An advantage of using derivative control action is that it responds to the rate of change of the actuating error and can produce a significant correction before the magnitude of the actuating error becomes too large. Derivative control thus anticipates the actuating error, initiates an early corrective action and tends to increase the stability of the system (Pillai *et al*., 2013).

The combined action has the advantages of each of the three individual control actions. The equation of a controller with this combined action is given by

$$u(t) = K_p e(t) + \frac{K_p}{T_i}\int_0^t e(t)dt + K_p T_d \frac{de(t)}{dt} \qquad 2.24$$

Where T_i = integration time

T_d = derivative time

$u(t)$ = controller output

$K_p e(t)$ = proportional control action

$\frac{K_p}{T_i}\int_0^t e(t)dt$ = integral control action

$K_p T_d \frac{de(t)}{dt}$ = derivative control action

Or the transfer function is

$$\frac{U(s)}{E(s)} = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) \qquad 2.25$$

Where K_p is the proportional gain, T_i is the integral time and T_d is the derivative time (Ogata, 2007).

The block diagram of a controller is shown in Figure 2.13.

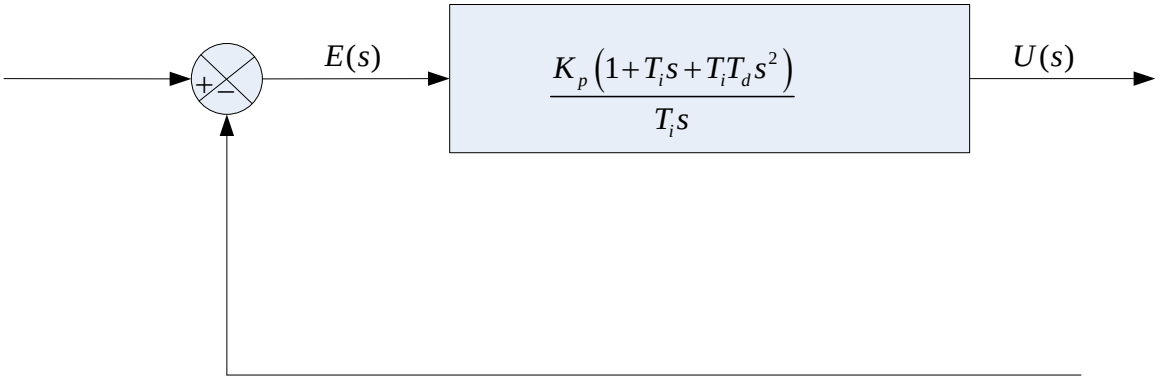


Figure 2.13: Block diagram of a PID controller (Source: Ogata, 2007)

2.1.2.4 Design procedures

The design problem is improving system performance by insertion of a controller. Setting (tuning) the gain is the first step in adjusting the system for satisfactory performance. In the trial and error approach to system design, we set up a mathematical model of the control system and adjust the parameters of a controller. The most time consuming part of such work is the checking of the system performance by analysis with each adjustment of the parameters (Pillai *et al*., 2013).

Once a satisfactory mathematical model has been obtained, the designer must construct a prototype and test the open-loop system. If absolute stability of the closed loop is assured, the designer closes the loop and tests the performance of the resulting closed-loop system. Because of the neglected loading effects among the components, nonlinearities, distributed parameters and so on which were not taken into consideration in the original design work, the actual performance of the prototype system will probably differ from the theoretical predictions. Thus the first design may not satisfy all the requirements on performance. By trial and error, the designer must make changes in the prototype until the system meets the specifications. In doing this, he or she must analyse each trial, and the results of the analysis must be incorporated into the next trial. The designer must see that final system meets the performance specifications and at the same time, is reliable and economical (Ogata, 2007).

2.1.2.5 Tuning PID controllers

Figure 2.14 shows a PID control of a plant. If a mathematical model of the plant can be derived, then it is possible to apply various design techniques for determining parameters of the controller that will meet the transient and steady-state specifications of the closed-loop system (Pillai *et al*., 2013).

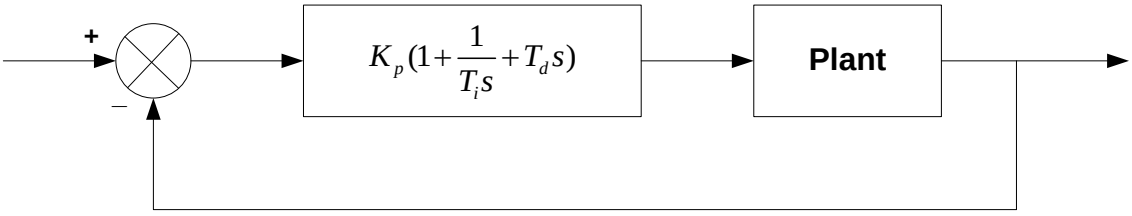


Figure 2.14: PID control of a plant (Source: Pillai *et al*., 2013).

The process of selecting the controller parameters (K_p, K_i and K_d) to meet given performance specifications is known as *controller tuning*. The individual effects of these terms on the closed-loop performance are summarized in Table 2.2.

Table 2.2: Effects of Independent P, I and D Tuning (Ang et.al, 2005)

Closed loop response	**Rise Time**	**Over-shoot**	**Settling Time**	**Steady State Error**	**Stability**
Increase K_p	Decrease	Increase	Small Increase	Decrease	Degrade
Increase K_i	Small Decrease	Increase	Increase	Large Decrease	Degrade
Increase K_d	Small Decrease	Decrease	Decrease	Minor change	Improve

Ziegler and Nichols (1942) suggested rules for tuning PID controllers (meaning to set values K_p, T_i and T_d) based on experimental step responses or based on the value of K_p that results in marginal stability when only the proportional control action is used (Ogata, 2007).

Ziegler and Nicholes (1942) proposed rules for determining values of the proportional gain K_p, integral time T_i and derivative time T_d based on the transient response characteristics of a given plant. Such determination of the parameters of PID controllers or tuning of PID controllers can be made by engineers on site by experiments on the plant. There are two methods called

Ziegler-Nichols tuning rules. In both methods, they aimed at obtaining 25% maximum overshoot in step response shown in Figure 2.15 (Ogata, 2007).

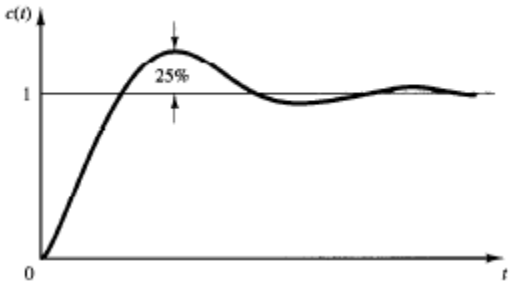


Figure 2.15: Unit-step response curve showing 25% maximum overshoot (Source: Ogata, 2007).

First method: In the first method, the response of the plant to a unit-step input is obtained experimentally shown in Figure 2.16.

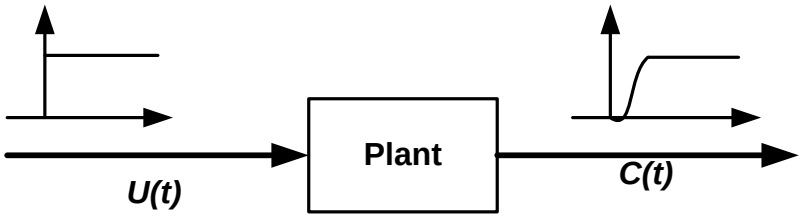


Figure 2.16: Unit-step response of a plant (Source: Ogata, 2007)

If the plant involves neither integrator(s) nor dominant complex-conjugate poles, then such a unit-step response curve may look like an S-shaped curve as shown in Figure 2.17. (If the response does not exhibit an S-shaped curve, this method does not apply) (Ogata, 2007).

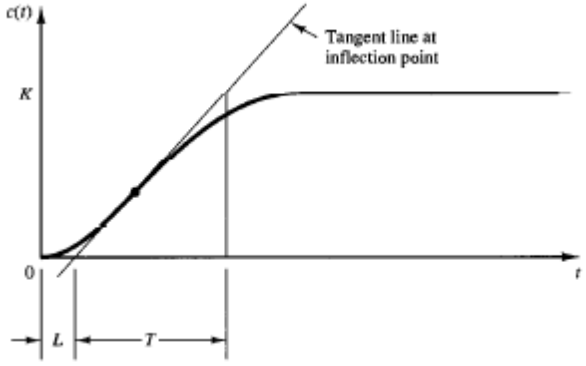


Figure 2.17: S-shaped response curve (Source: Ogata, 2007)

The S-shaped curve may be characterised by two constants, delay time L and time constant T.

Ziegler and Nichols (1942) suggested to set the values of K_p, T_i and T_d according to the formula shown in Table 2.3.

Table 2.3: Ziegler-Nichols Tuning Rule Based on Step Response of Plant (First Method) (Ogata, 2007)

Type of controller	K_p	T_i	T_d
PID	$1.2\frac{T}{L}$	$2L$	$0.5L$

$$G_c(s) = \mathrm{K}_p\left(1+\frac{1}{T_i s}+T_d s\right)$$

$$G_c(s) = 1.2\frac{T}{L}\left(1+\frac{1}{2Ls}+0.5Ls\right) \qquad 2.26$$

$$G_c(s) = 0.6T\frac{\left(s+\frac{1}{L}\right)^2}{s}$$

Thus, the PID controller has a pole at the origin and double zeros at **s = -1/L**

Second method: The second method uses the proportional control action only (Ogata, 2007).

Although conventional technique like Zeigler-Nichols method does not give an optimized value for PID controller parameters (Pillai *et al.*, 2013).

2.1.3 Genetic algorithm (GA)

The key issue for PID controllers is the accurate and efficient tuning of parameters. In practice, controlled systems usually have some features, such as nonlinearity, time-variability, and time delay, which make controller parameter tuning more complex. Moreover, in some cases, system parameters and even system structure can vary with time and environment. As a result, the conventional PID parameter tuning methods such as Zeigler-Nichols method are not suitable for these difficult calculations. Therefore, with the aid of Genetic Algorithms (GAs), Artificial Neural Networks and Fuzzy Logic, many researchers have recently proposed various alternative, intelligent PID controllers (Zhang *et al*., 2009).

Genetic Algorithm is a stochastic search and optimization method that mimics the process of natural evolution (Pillai *et al*., 2013). The steps involved in creating and implementing a genetic algorithm are:

1. Generate an initial, random population of individuals for a fixed size.
2. Evaluate their fitness.
3. Select the fittest members of the population.
4. Reproduce using a probabilistic method (e.g., roulette wheel).
5. Implement crossover operation on the reproduced chromosomes (choosing probabilistically both the crossover site and the mates).
6. Execute mutation operation with low probability.
7. Repeat step 2 until a predefined convergence criterion is met.

The convergence criterion of a genetic algorithm is a user-specified condition for example the maximum number of generations or when the string fitness value exceeds a certain threshold (Ibrahim, 2005).

1. Initialization: The Genetic Algorithm has to be initialized before the algorithm can proceed. The first and the most crucial step is to encoding the problem into suitable GA chromosomes

or individuals and then construct the population. Some works recommend 20 to 100 chromosomes in one population. The more the chromosomes number, the better the chance to get the optimal results (Mirzal *et al.,* 2012). However, considering the execution time, 80 or 100 chromosomes is used in each generation (Mirzal *et al.,* 2012).

Encoding is done in real number rather than binary encoding because the latter discards the parameters value if it exceeds its precision capability. Each chromosome comprises of three parameters, K_d, K_p, K_i, with value bounds varied depend on the delay and objective functions used (Mirzal *et al.,* 2012).

2. Evaluation: The most crucial step in applying GA is to choose the objective functions that are used to evaluate fitness of each chromosome. Some works use *performance indices* as the objective functions. Others use Mean of the Squared Error (MSE), Integral of Time multiplied by Absolute Error (ITAE), Integral of Absolute Magnitude of the Error (IAE), and Integral of the Squared Error (ISE) (Mirzal *et al.,* 2012). The performance of each individual is measured and assessed by the objective or evaluation function. The objective function for this research is to find a PID controller that gives the smallest overshoot, fastest rise time and quickest settling time. When the chromosome enters the evaluation function, it is split up into its three terms (the K_p, K_i and K_d term). The objective function assigns each individual a corresponding number called its fitness. The fitness of each chromosome is assessed and a survival of the fittest strategy is applied. If the termination criteria are not met with the current population then new individuals are created with genetic operators (Ibrahim, 2005).

3. Selection: During the ***reproduction*** phase, the fitness value of each chromosome is assessed. This value is used in the selection process to provide bias towards fitter individuals. Just like in natural evolution, a fit chromosome has a higher probability of being selected for reproduction. This continues until the selection criterion has been met. The probability of an individual being selected is thus related to its fitness, ensuring that fitter individuals are more

likely to leave offspring. Multiple copies of the same string may be selected for reproduction and the fitter strings should begin to dominate (Ibrahim, 2005).

4. Crossover: Once the selection process is complete, the ***crossover*** algorithm is initiated.

The crossover operations swap certain parts of the two selected strings in a bid to capture the good parts of old chromosomes and create better new ones.

Genetic operators manipulate the characters of a chromosome directly, using the assumption that certain individual gene codes, on average, produce fitter individuals. The crossover probability indicates how often crossover is performed. A probability of 0% means that the offspring will be exact replicas of their parents and a probability of 100% means that each generation will be composed of entirely new offspring (Ibrahim, 2005). Using selection and crossover on their own will generate a large amount of different strings. However, there are two main problems with this:

1. Depending on the initial population chosen, there may not be enough diversity in the initial strings to ensure the GA searches the entire problem space.
2. The GA may converge on sub-optimum strings due to a bad choice of initial population.

These problems may be overcomed by the introduction of a mutation operator into the GA (Ibrahim, 2005).

5. Mutation: This is the occasional random alteration of a value of a string position. It is considered a background operator in the genetic algorithm.

The probability of mutation is normally low because a high mutation rate would destroy fit strings and degenerate the genetic algorithm into a random search. Mutation probability values of around 0.1% or 0.2% are common. These values represent the probability that a certain string will be selected for mutation for an example for a probability of 0.1%; one string in one thousand will be selected for mutation. Once a string is selected for mutation, a randomly chosen element of the string is changed or mutated (Ibrahim, 2005).

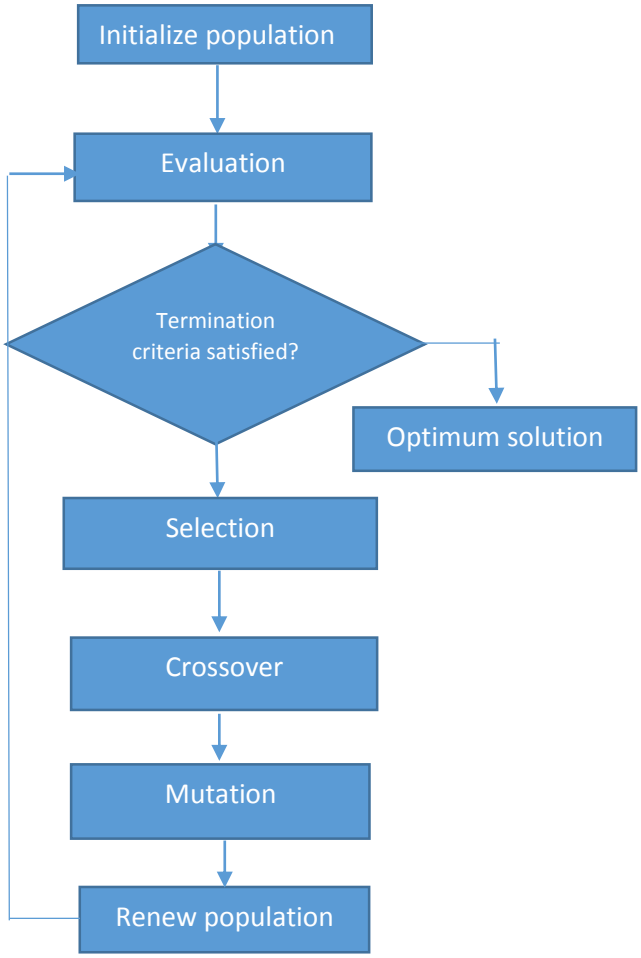


Figure 2.18: Work flow of genetic algorithm (Pillai *et al*., 2013)

2.1.4 Bond graph

Bond graphs are a domain-independent graphical technique that exploits the fundamental laws of energy to model a system very close to reality i.e. nonlinear system (Broenink, 1999). With

bond graph method, the basic law principles proved by Kirtchoff 's current flow in circuit elements Resistor, Inductor and Capacitor (RLC) are used to deduce algebraic equations representing the modelled system. These characteristics of the RLC elements in electrical are compared directly to analogues elements in other fields of physical science to produce equivalent nonlinear models based on energy share concept. The key graphical model is realised via bonds indicating effort (e) and flows (f) and the indications of the various junctions with 1, 0, Transformer (TF), Gyrator (GY), Signals and Modulated GY, Modulated TF (Samantaray and Bouamama, 2008).

In bond graph theory, the four major modelling techniques for system derivation; algorithmic, algebraic, mathematical and symbolic methods are aided by a graphical and visual implementation. The compositional elements of a system or subsystems interact through energy exchange and information. This form a synergy which completely defines the dynamic system setup. Hence, bond graph depicts a pictorial display of flow of energy and information. Some of the analogous categories in other field are potential variables (e.g. pressure, electrical potential, temperature, chemical potential, force, etc.) called generalized effort (e) and a current variable (e.g. volume flow, current, entropy flow, velocity, molar flow, etc.) referred to as generalized flow (f). The positive sense of power flow is represented by the half arrow on the bond. The energy is exchanged according to the sense represented by a power direction (half arrow head) only when the power variables have the same sign, i.e. either both are positive or both are negative; otherwise, the energy transfer is in the opposite direction to the assigned power direction (Samantaray and Bouamama, 2008).

The causality (cause and effect relationship) is the most important concept in bond graph theory. Indeed, the determination of causes and effects in the system is directly achieved through graphical representation. In a bond graph model, the causality is denoted by a cross-

stroke at the end of a bond from which force is injected, or energy evolved thereby depicting either integral causality or differential causality. (Samantaray and Bouamama, 2008).

Moreover, bond graphs have certain advantages over network models. In particular, bond graph elements exist which allow multiport elements to be modelled explicitly, whereas network models, even of simple two-port elements, are awkward to draw and manipulate. From a practical viewpoint the relatively compact nature of bond graphs commends the technique as the basis for computer-aided modelling. This, coupled with the development of low-cost, high-speed interactive computer systems, distinguishes the bond graph approach as potentially the most useful of the systematic modelling techniques (Wellstead, 1979). In support of this, Samantaray and Bouamama (2008) also outlined the advantages of bond graph modelling as follows:

1. It is a unique language for all physical domains.
2. It clearly shows the cause and effect relations in the model.
3. It allows further possible development and evolution of the model.
4. It is also a tool for analysing the system's structural properties.

2.1.4.1 Characteristic of bond graph elements

A. Inertia element:

Where e is effort variable and f is the flow variable such that (Samantaray and Bouamama ,2008)

$$\begin{aligned} & v = L\frac{di}{dt} \\ & i.e \\ & i = \frac{1}{L}\int_{-\infty}^{now} v\,dt \\ & effort = [\,function\,or\,cons\,]x\int_{-\infty}^{now} flow\,dt \end{aligned} \qquad 2.27$$

B Compliant element:

$$
\begin{aligned}
&i = C\frac{dv}{dt} \\
&i.e \\
&v = \frac{1}{C}\int_{-\infty}^{now} i\,dt \\
&effort = [\,function\,or\,const\,]x\int_{-\infty}^{now} flow\,dt
\end{aligned}
\qquad 2.28
$$

C Resistive element:

$$
\begin{aligned}
&i = \frac{1}{R}v \\
&i.e \\
&v = Ri \\
&effort = [\,function\,or\,const\,]\,x\;flow
\end{aligned}
\qquad 2.29
$$

2.1.4.2 Systematic procedure to derive a bond–graph model

To generate a bond–graph model starting from an *ideal–physical model*, a *systematic method* exist (Broenink, 1999**)**. This procedure consists roughly of the identification of the domains and basic elements, the generation of the connection structure (called the *junction structure*), the placement of the elements, and possibly simplifying the graph. The procedure is different for the mechanical domain compared to the other domains. These differences are indicated between parentheses. The reason is that elements need to be connected to *difference variables* or *across variables*. The efforts in the non–mechanical domains and the velocities (flows) in the mechanical domains are the across variables needed.

Step 1 and 2 concern the identification of the domains and elements.

1 Determine which physical domains exist in the system and identify all basic elements like C, I, R, Se, Sf, TF and GY.

Give every element a unique name to distinguish them from each other.

2 Indicate in the ideal–physical model per domain a reference effort (reference velocity with positive direction for the mechanical domains).

Note that only the references in the mechanical domains have a direction.

Steps 3 through 6 describe the generation of the connection structure (called the *junction structure*).

3 Identify all *other* efforts (mechanical domains: velocities) and give them unique names.

4 Draw these efforts (mechanical: velocities), and *not* the references, graphically by 0–junctions (mechanical: 1–junctions). Keep if possible, the same layout as the Ideal Pysical Model (IPM).

5 Identify all effort differences (mechanical: velocity (= flow) differences) needed to connect the ports of all elements enumerated in step 1 to the junction structure.

Give these differences a unique name, preferably showing the difference nature. The difference between e_1 and e_2 can be indicated by e_{12}.

6 Construct the effort differences using a 1–junction (mechanical: flow differences with a 0-junction) according to Figure 2.19, and draw them as such in the graph.

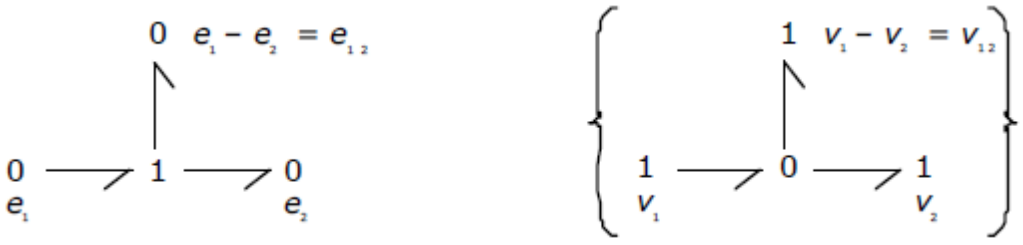


Figure 2.19: Construction of effort differences (velocity differences) (Broenink, 1999**)**.

The junction structure is now ready and the elements can be connected.

7 Connect the port of all elements found at step 1 with the 0–junctions of the corresponding efforts or effort differences (mechanical: 1–junctions of the corresponding flows or flow differences).

8 Simplify the resulting graph by applying the following simplification rules shown in Figure 2.20:

- A junction between two bonds can be left out, if the bonds have a ‘through’ power direction (one bond incoming, the other outgoing).
- A bond between two the same junctions can be left out, and the junctions can join into one junction.
- Two separately constructed identical effort or flow differences can join into one effort or flow difference.

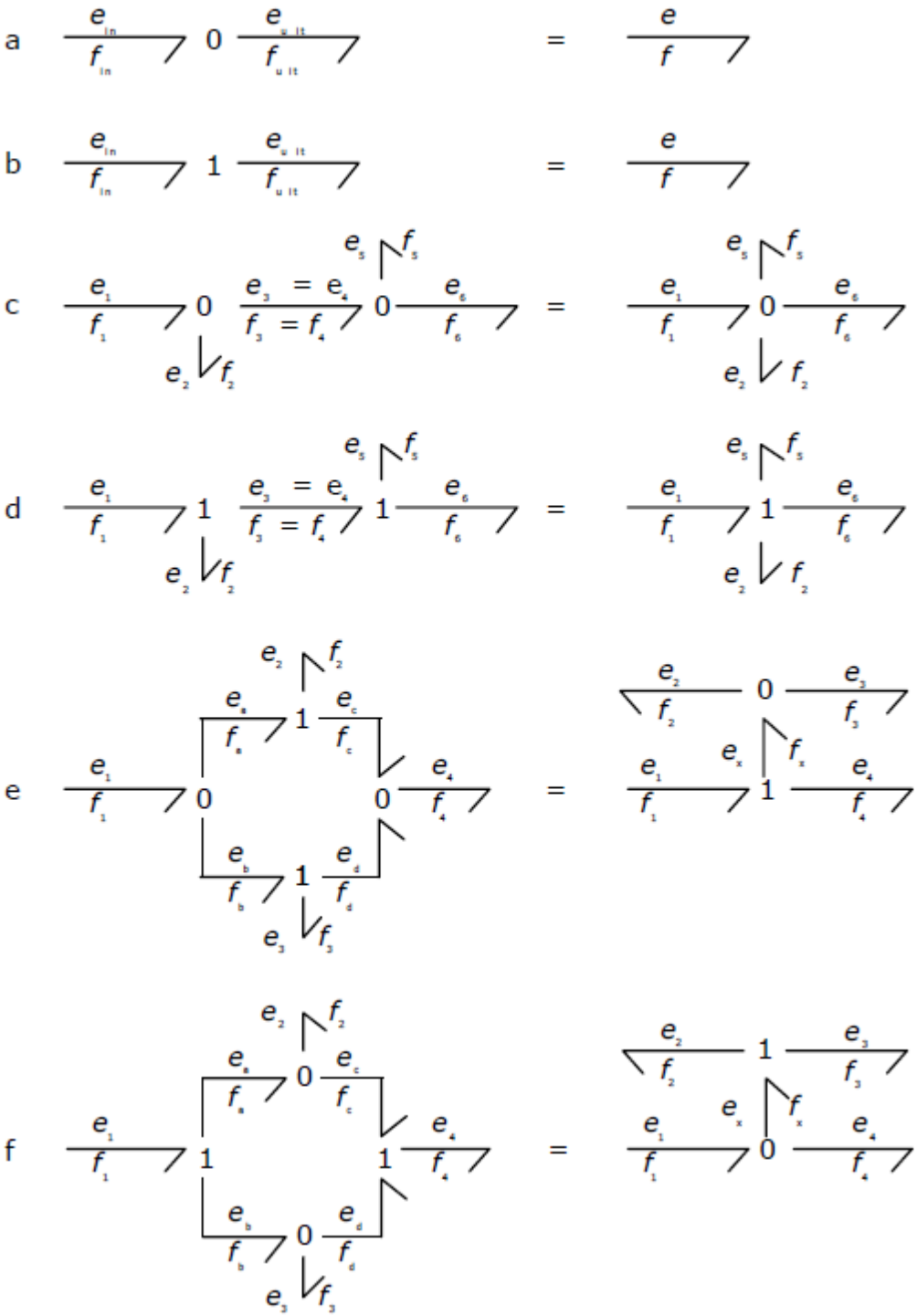


Figure 2.20: Simplification rules for the junction structure. (a, b) Elimination of a junction between bonds. (c, d) Contraction of two the same junctions. (e, f) Two separately constructed identical differences fuse to one difference (Broenink, 1999**)**..

2.2 Review of Related Past Works

Several control methods have been proposed and used to control antenna position system but know optimization algorithm, especially GA, has been used to control this system. However, it is difficult using the traditional PID controller to obtain excellent control performance due to its limitations such as avoidable overshoots and large settling time (Chishti *et al.*, 2014). In order to achieve good control performance for the deep space antenna system, several modern control strategies have been proposed. The latest being:

Okumus *et al.* (2012) controlled an antenna azimuth position system by using a Proportional-Integral-Derivative (PID) controller and a Fuzzy Logic Controller (FLC) designed in Matlab/Simulink environment. In order to obtain the best system response with FLC, different types of fuzzy rules and membership functions were tested. System responses with proposed controller and PID controller were compared to each other and commented according to the results. Simulation results showed that the performance of proposed controller was better than the other controllers in terms of the settling time and overshoot. However, no optimization algorithm was used.

Saini (2014) controlled an antenna azimuth position system by using three controllers: PID, Fuzzy and Fuzzy-PID. The performance of different controllers in terms of various performance specifications such as rise time, settling time, peak overshoot, integral square error (ISE), integral absolute error (IAE) and integral time absolute error (ITAE) has been compared. Simulation results reveals that fuzzy controller outperforms PID controller in terms of rise time, settling time and peak overshoot while fuzzy-PID controller outperforms fuzzy controller in terms of settling time and rise time. However, no optimization algorithm was used for its parameters.

Okumus *et al.* (2014) controlled an antenna azimuth position system by using a classical proportional-integral controller (PI), fuzzy logic controller (FLC) and a self-tuning fuzzy logic controller (STFLC). The proposed self-tuning fuzzy logic controller was designed via

Matlab/SIMULINK environment in order to tune the scaling factors G_1 and G_2, the fuzzy gains of the controller inputs error and its change, on-line. Simulation results showed that the performance of proposed controller was better than the other controllers in terms of the settling time and overshoot. Also no optimization algorithm was used to optimize the system.

Chishti *et al*. (2014) implemented the design and control of antenna position system using Matlab/Simulink. In this work, the response of the system was checked without a controller and the response of the system was analysed. For getting a better response, a PID Controller was used and it was shown that the response was better than without a controller. Further, LQR Controller was used and the response was better than PID Controller. Also, no optimization algorithm was used to tune its parameters. In conclusion, it was shown that the response of the system using LQR controller was better than with PID controller as regards to quicker settling time and lesser percentage overshoot.

From the above review of related past works, it can be seen that no optimization algorithm has been used to optimise the PID Controller parameters for deep space antenna positioning system. Hence, this research is undertaken to fill this gap.

CHAPTER THREE

MATERIALS AND METHODS

3.1 Introduction

In this chapter, the detailed procedures for the modelling and optimization of the nonlinear system and subsystems are carried out. The interfaces for the 20sim software used for the bond graph modelling and the Matlab used for optimization of the system are attached in Appendix A and B respectively. Also, abbreviations and their meanings used here are given in the preliminary page.

3.2 Materials/Equipment

1. 20sim software
2. Matlab
3. Genetic algorithm codes

3.3 Derivation of Bond Graph Models of the System

The procedures for the derivation of the bond graph model of the system were:

Step 1: The physical domains which exist in the system were determined and all the basic elements like the Capacitor, Inductor, Resistor, Source effort, Source flow, Transformer and Gyrator were identified. Every element was given a unique name of C, I, R, Se, Sf, TF and GY respectively to distinguish it from others.

Step 2: An effort in the ideal–physical model per domain was indicated like the electric mains (source) for an electrical domain and gyrator for a mechanical domain.

Steps 3 through 6 describe the generation of the connection structure (called the junction structure).

Step 3: All other efforts were identified and given unique names such as Transformer as TF.

Step 4: These efforts were drawn, and the positions were noted, graphically by 0–junctions or 1–junctions. 1-junction signifies same flow through the elements like current for electrical

domain and velocity for mechanical domain while 0-junction signifies different flow through the elements.

Step 5: All effort differences were identified and connected to the ports of all elements enumerated in step 1 to the junction structure. These differences were given a unique name, preferably e_1, e_2 and so on showing the difference in nature. The difference between e_1 and e_2 can be indicated by e_{12}.

Step 6: The effort differences were constructed using a 1–junction and a 0-junction as shown in Figure 3.2 through Figure 3.7.

3.4 Open-Loop Model of the System

The open loop systems model depicts the system prior to application of a controller and feedback. The various sub models are shown below and the procedures for modelling it using bond graphs is same as given in section 3.3.

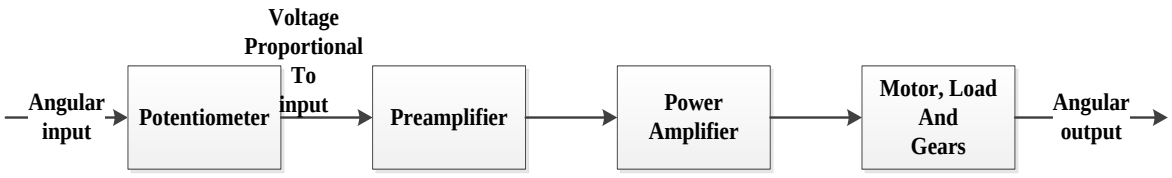


Figure 3.1: Open-loop model of the system

3.4.1 The Potentiometer:

The potentiometer type is a sensor with the characteristic such that there is a direct correlation between the electrical domain and the mechanical domain.

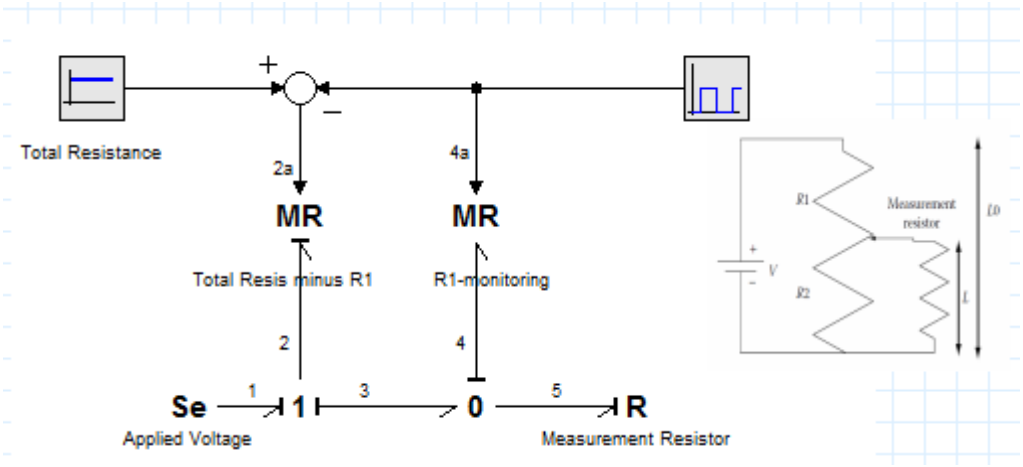


Figure 3.2: Bond Graph model of Potentiometer

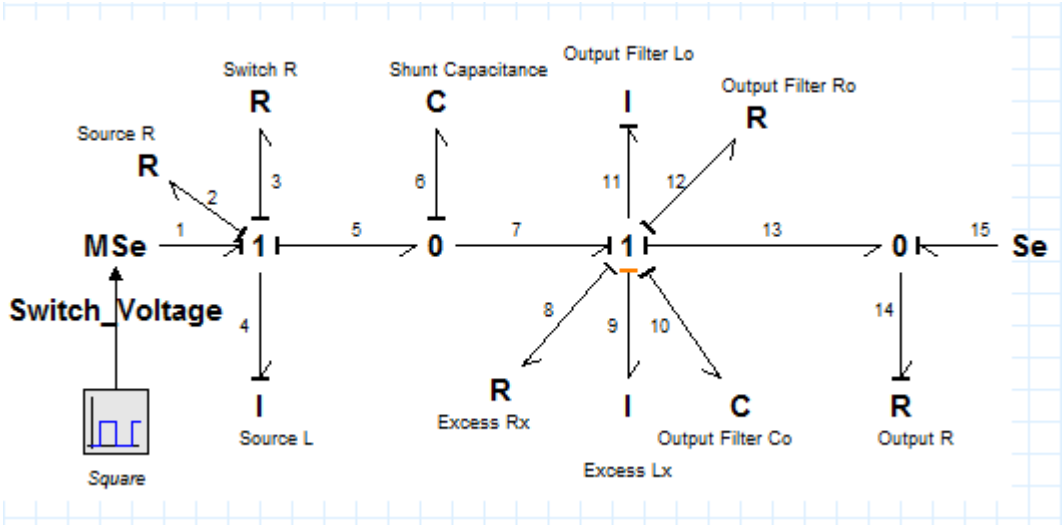


Figure 3.3: Bond Graph Model of Class E Pre/Power Amplifier

From Figure 3.3, the dynamic model equations are derived as follows. Definition of parameters are given in the preliminary page:

$$e_6 = \frac{q_6}{C_6};\ f_{10} = \frac{P_{11}}{L_{11}};\ f_4 = \frac{P_4}{L_4};\ e_{10} = \frac{q_{10}}{C_{10}};\ e_3 = R_3 \frac{P_4}{L_4}; \tag{3.1}$$

$$\dot{q}_{10} = C_{10} \frac{P_9}{L_9};\ e_8 = R_8 \frac{q_{10}}{C_{10}};\ e_{12} = R_{12} \frac{q_{10}}{C_{10}}; \tag{3.2}$$

$$e_9 = \left(\left(\left(R_{12} \frac{q_{10}}{C_{10}} + \frac{q_{10}}{C_{10}} \right) + R_8 \frac{q_{10}}{C_{10}} \right) + E \right) / \left(1.0 + \frac{P_9}{L_9} \frac{L_{11}}{q_{11}} \right) \tag{3.3}$$

$$f_6 = \frac{P_4}{L_4} - \frac{q_{10}}{C_{10}}; e_2 = R_2 \frac{P_4}{L_4}; \tag{3.4}$$

$$e4 = \left(\left(E - \frac{q_8}{C_8} \right) - R_2 \frac{P_4}{L_4} \right) - R_3 \frac{P_4}{L_4} \tag{3.5}$$

$$e_{11} = \frac{q_8}{C_8} - \left(\left(\left(\left(R_{12} \frac{q_{10}}{C_{10}} + \frac{q_{10}}{C_{10}} \right) + R_8 \frac{q_{10}}{C_{10}} \right) + E \right) + \frac{\dot{q}_9}{C_9} \right) \tag{3.6}$$

$$f_{14} = R_{14} \frac{P_{11}}{L_{11}} - \frac{P_{11}}{L_{11}} \tag{3.7}$$

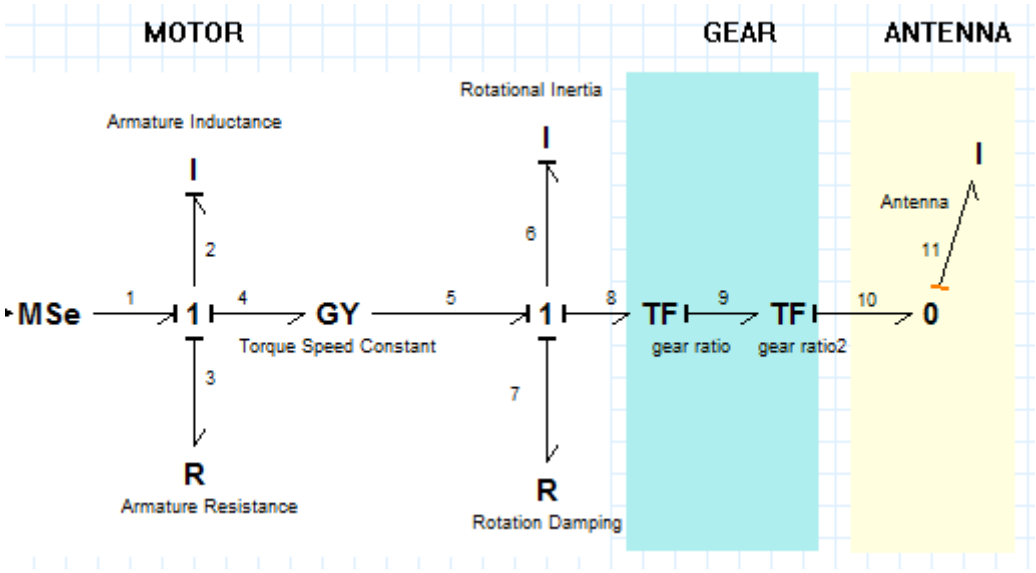


Figure 3.4: Bond Graph Model of Motor and Load

3.4.2 Mathematical modelling of Nonlinear Motor System

From Figure 3.4, the state variables are P_2 and P_6

The contribution of each element to the system are derived as follows:

$$e_{_1} = E \tag{3.8}$$

$$F_2 = \frac{P_2}{La}, F_6 = \frac{P_6}{I_1}, F_{11} = \frac{P_{11}}{I_L} \tag{3.9}$$

$$e_3 = R_a f_3 = R_a f_2 = R_a \frac{P_2}{L_a} \tag{3.10}$$

$$e_7 = R_1 f_7 = R_1 \frac{P_6}{I_1} \tag{3.11}$$

$$e_4 = k\varphi f_5 = k\varphi f_6 = k\varphi \frac{P_6}{I_1} \tag{3.12}$$

$$e_5 = k\varphi f_4 = k\varphi \frac{P_2}{L_a} \tag{3.13}$$

$$e_8 = \mu_1 e_9 = \mu_1\mu_2 e_{10} = \mu_1\mu_2 e_{11} = \mu_1\mu_2 \int \frac{\dot{P}_{11}}{I_L} \tag{3.14}$$

$$f_{10} = \mu_2 f_9 = \mu_1\mu_2 f_8 = \mu_1\mu_2 f_6 = \mu_1\mu_2 \frac{P_6}{I_1} \tag{3.15}$$

The storage elements will receive from the system

$$\begin{aligned} I_2 &: e_2 = e_1 - e_3 - e_4 = \dot{P}_2 \\ &: \dot{P}_2 = E - R_a \frac{P_2}{L_a} - k\varphi \frac{P_6}{I_1} \end{aligned} \tag{3.16}$$

$$\begin{aligned} I_6 &: e_6 = P_6 = e_5 - e_7 - e_8 \\ &: P_6 = k\varphi \frac{P_2}{L_a} - R_1 \frac{P_6}{I_1} - \mu_1\mu_2 \int \frac{\dot{P}_{11}}{I_L} \end{aligned} \tag{3.17}$$

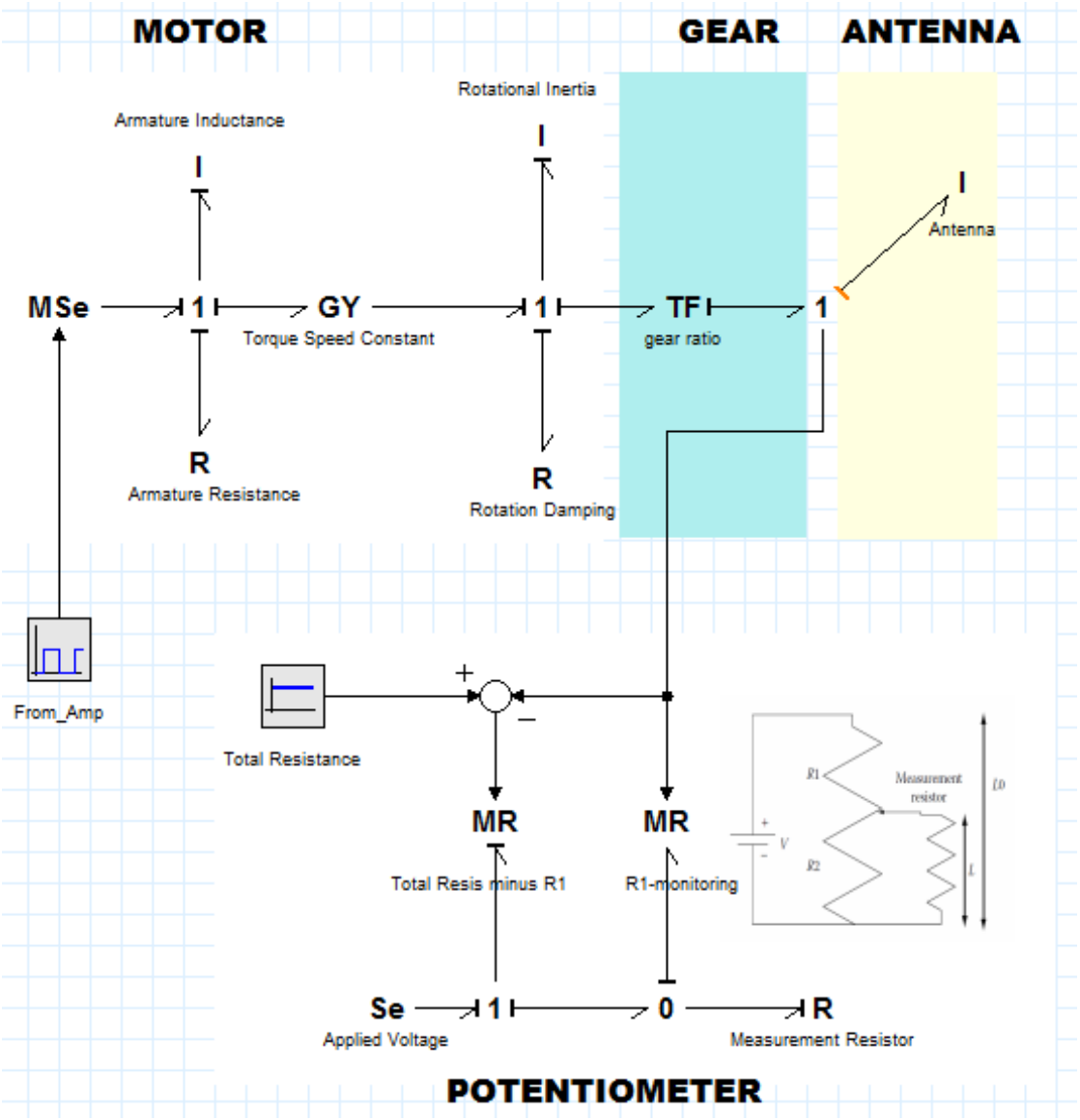


Figure 3.5: Open loop sub-system blocks

3.5 Closed-Loop Model of the System

The purpose of this system is to have the azimuth angle output of the antenna, $\Theta_o(t)$, following the input angle of the potentiometer, $\Theta_i(t)$. The block diagram representation of the closed-loop system is as shown in Figure 3.6. Moreover, the design visualization of the closed-loop system with bond graph is as shown in Figure 3.7 which comprises the various subsystems and the PID controller. The procedures for its modelling is same as given in section 3.3 of this work.

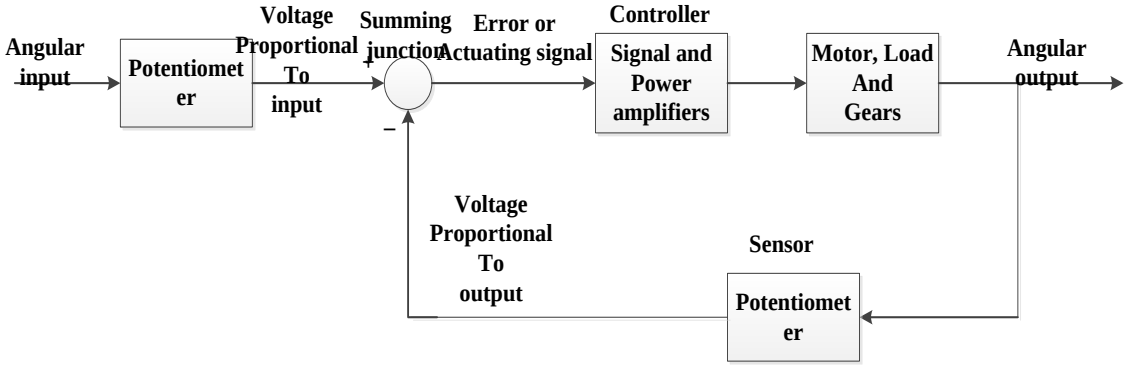


Figure 3.6: Closed-loop model of the system

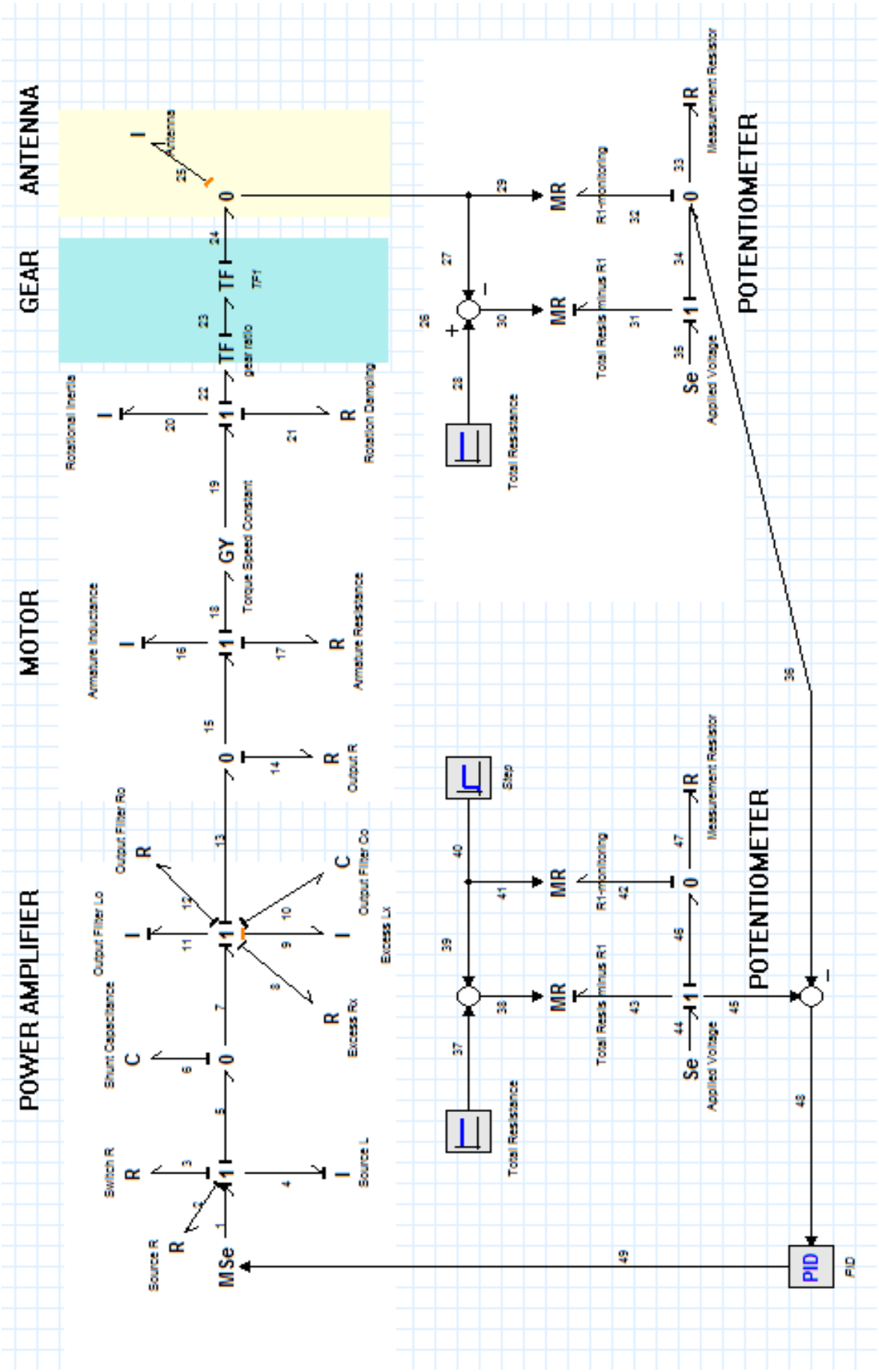


Figure 3.7 Modelling of the Closed Loop System using Bond Graph

3.6 PID Tuning using GA

In this research, Matlab Genetic Algorithm Toolbox was used to optimize and simulate the system. The codes for the implementation of the GA for this research based on the Genetic Algorithm Optimization Toolbox (GAOT) is given in Appendix C.

The work flow for the GA implementation on this work is as shown below.

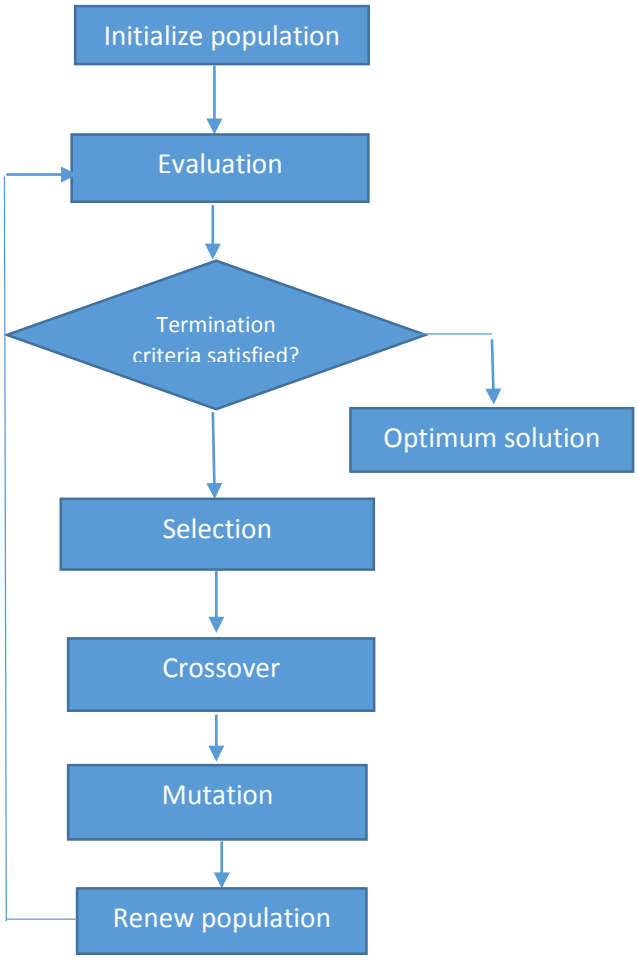


Figure 3.8: Work flow of the Genetic Algorithm (GA)

3.5 Simulation of the System

Simulation of the system was done using Matlab. The procedures were:

1. The GA optimized PID Controller was initialized with a population size of 80 chromosomes and the responses were analysed for different preamplifier's gain, K.
2. Gain value "K" represents the preamplifier block. The value of preamplifier gain "K" can be found out for stable system by utilizing the Routh-Herwitz criterion. According to this criterion, system will give stable response if we take the value of gain "K" in the range 0-262 according to the method of Chishti *et al*.,(2014). Utilizing this criterion, different amplifier gain values at interval of 50 was selected for this work.
3. The objective function for this research was to find a PID controller that gives the smallest overshoot, fastest rise time and quickest settling time.
4. A probability of 70% (0.7) cross over operation was selected for this work as this gives the best result. Reason being that a probability of 0% means that the offspring will be exact replicas of their parents and a probability of 100% means that each generation will be composed of entirely new offspring as reported by Ibrahim (2005).
5. A mutation probability of 0.2% was selected for this work. Iteration was done 100 times reason being that optimal performance was obtained at this value.

RESULTS AND DISCUSSION

4.1 Results

Simulation result for the deep space antenna control system using Matlab/Simulink is shown below. The GA parameters, plant parameters and the system response values and curves are also shown for the different amplifier's gain values. By utilizing the Routh-Herwitz criterion as discussed in chapter three, different amplifier gains "K" in the range 0-262 were selected for this work at interval of 50 and also the GA parameters were selected as discussed in chapter three. The system response values were extracted from the simulation curves.

(1) System Response at Amplifier Gain of 50

Table 4.1: (a) GA parameters, (b) Plant parameter and (c) System response parameter

S/N	Parameter	Value
1	Population	80
2	Iteration	100
3	Crossover	0.7
4	Mutation	0.2

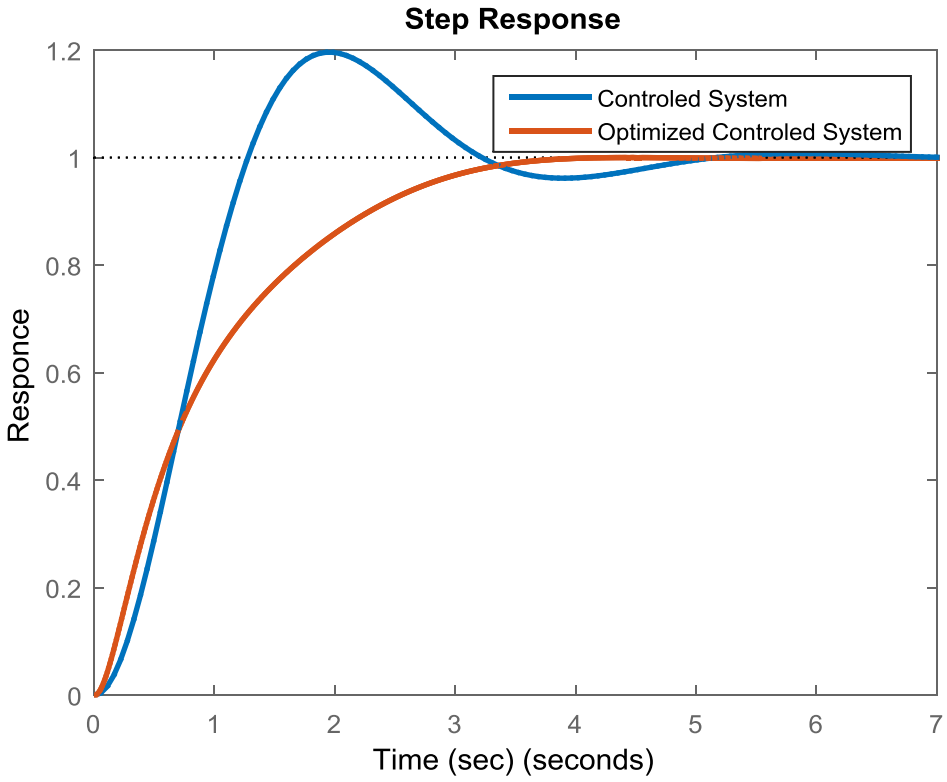
(a)

S/N	Parameter	Value
1	Amplifier gain	50

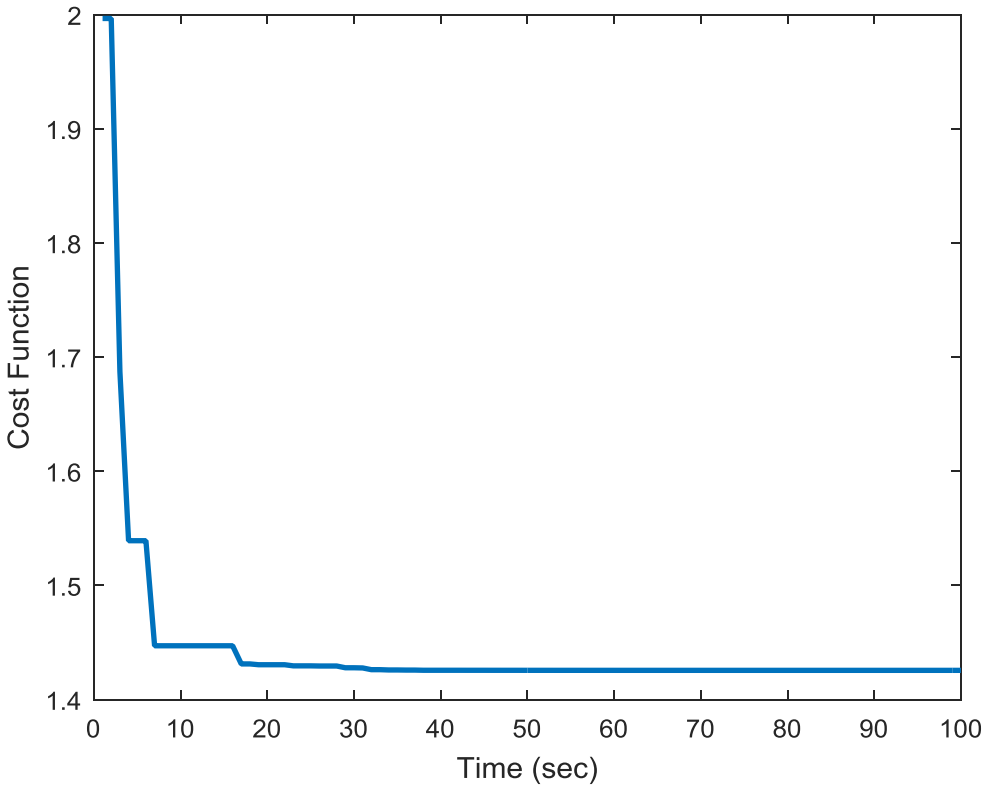
(b)

S/N	Parameter	Value
1	PID Controller gain (K_P, K_I & K_D)	0.2862, 1 & 0.3063
2	Best cost	1.4255
3	Rise time	2.1064sec
4	Settling time	3.2339sec
5	Overshoot	0%
6	Undershoot	0%
7	Peak	1.0000sec
8	Peak Time	4.4407sec

(c)



(a)



(b)

Figure 4.1: (a) System response for gain value of 50 (b) GA cost function minimization for gain value of 50

From Table 4.1 and Figure 4.1 above, it can be seen that the optimised system response at amplifier gain value of 50 produced a fast and stable response as there are no overshoots and undershoots which is in agreement with Routh-Herwitz criterion for stability as reported by Chishti *et al*, (2014) and Okumus *et al*.(2014) Also it can be noticed that there are no problem of peaking in its transient state characteristics. This problem can be referred to as a non-uniform movement in its transient state, a reversal kind of movement at some point.

(2) System Response at Amplifier Gain of 100

Table 4.2: (a) GA parameters**,** (b) Plant parameter and (c) System response parameters

S/N	Parameter	Value
1	Population	80
2	Iteration	100
3	Crossover	0.7
4	Mutation	0.2

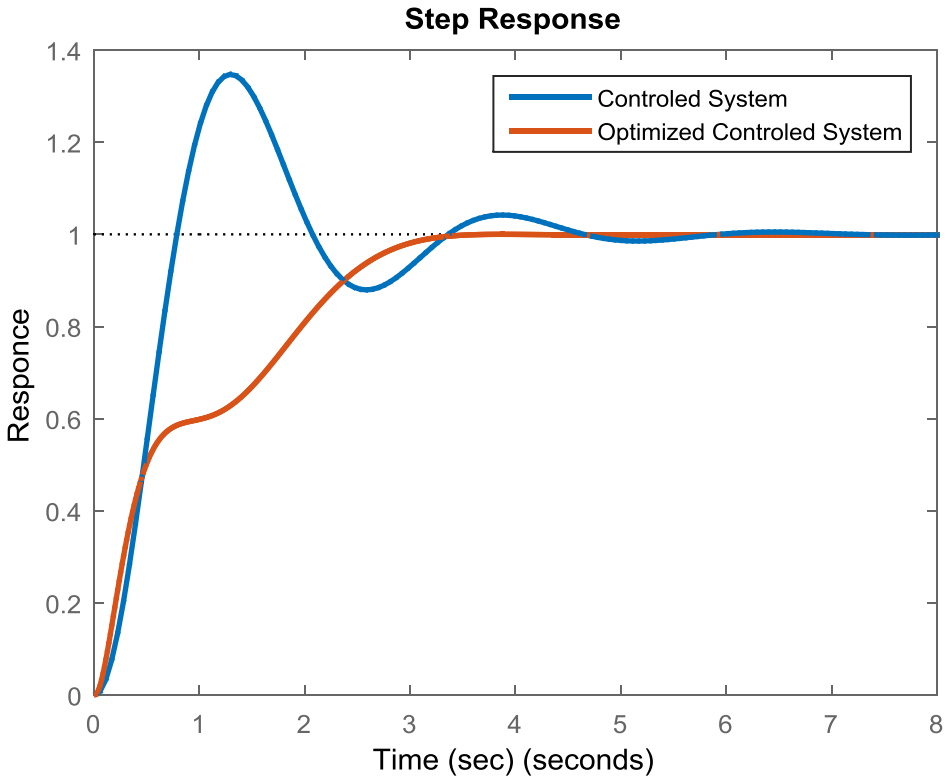
(a)

S/N	Parameter	Value
1	Amplifier gain	100

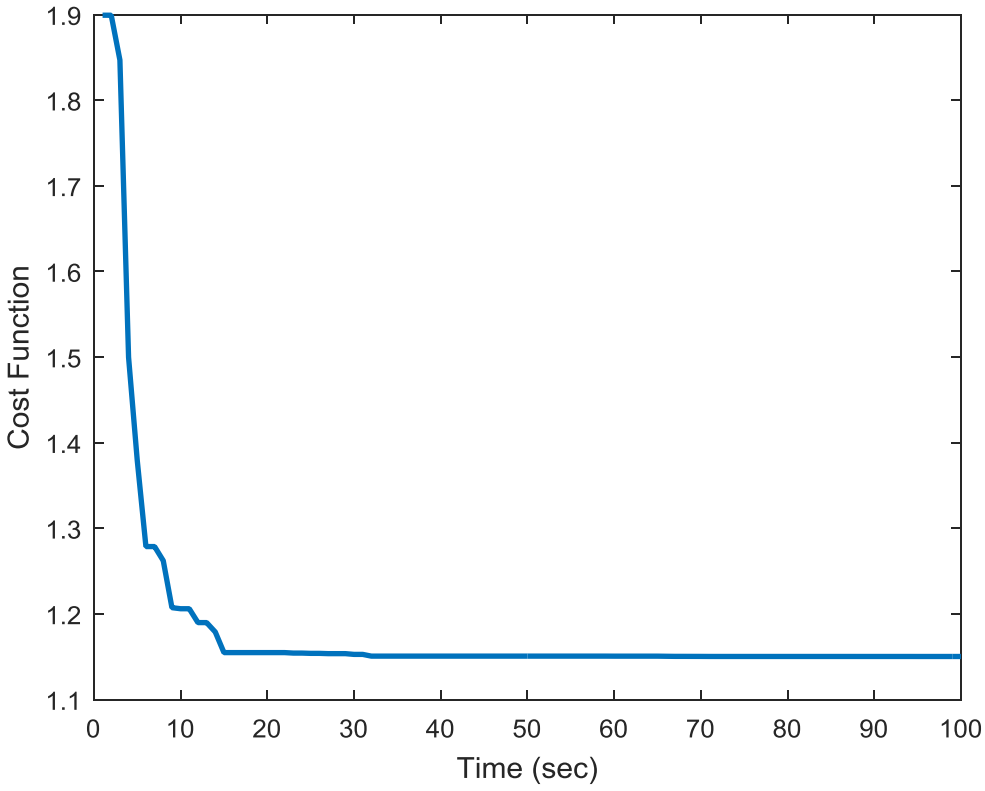
(b)

S/N	Parameter	Value
1	PID Controller gain (K_P, K_I & K_D)	0.0930, 1.0000 & 0.2905
2	Best cost	1.1503
3	Rise time	2.2412sec
4	Settling time	2.9861sec
5	Overshoot	0%
6	Undershoot	0%
7	Peak	1.0000sec
8	Peak Time	3.8740sec

(c)



(a)



(b)

Figure 4.2: (a) System response for gain value of 100 and (b) GA cost function minimization for gain value of 100

From Table 4.2 and Figure 4.2 shown above, the optimized system response at amplifier gain of 100 rises slower and settles faster as compared to that at amplifier gain of 50 producing 0% overshoots and undershoots. This is also in agreement with Routh-Herwitz criterion for stability as observed by Saini (2014) and Chishti *et al,* (2014). However, problem of peaking starts to build up in its transient state.

(3) System Response at Amplifier Gain of 150

Table 4.3: (a) GA parameters, (b) Plant parameter and (c) System response parameter

S/N	Parameter	Value
1	Population	80
2	Iteration	100
3	Crossover	0.7
4	Mutation	0.2

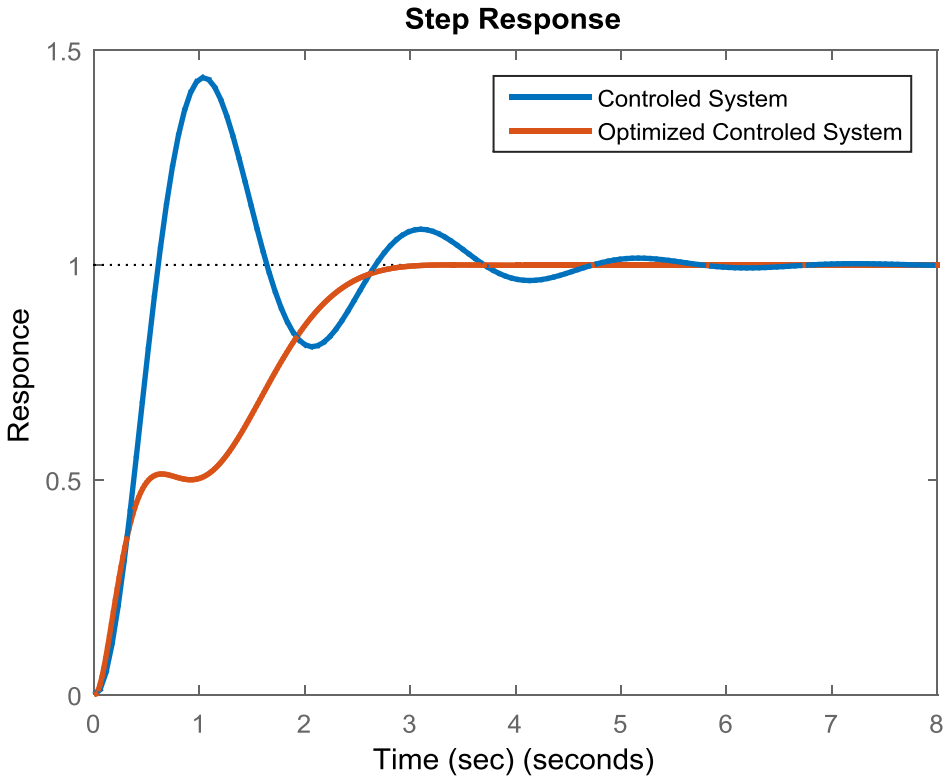
(a)

S/N	Parameter	Value
1	Amplifier gain	150

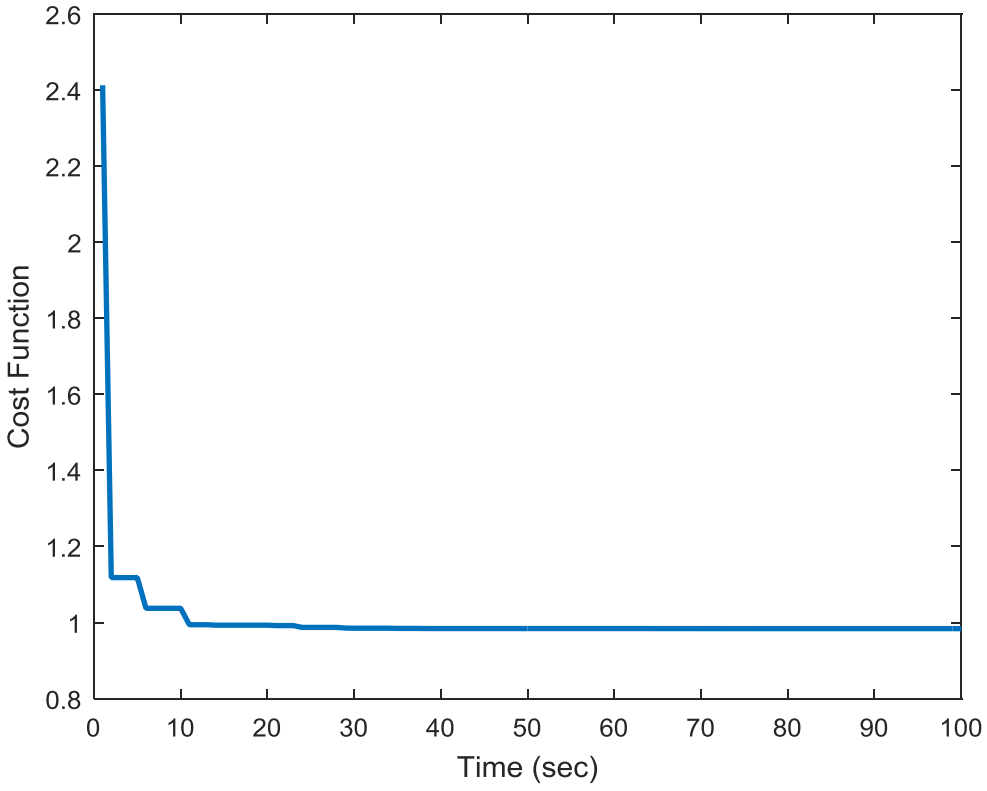
(b)

S/N	Parameter	Value
1	PID Controller gain (K_P, K_I & K_D)	0.0211, 1.0000 & 0.2735
2	Best cost	0.98395
3	Rise time	2.0182sec
4	Settling time	2.6281sec
5	Overshoot	0%
6	Undershoot	0%
7	Peak	1.0000sec
8	Peak Time	3.3936sec

(c)



(a)



(b)

Figure 4.3: (a) System response at gain value of 150 and (b) GA cost function minimization for gain value of 150

From Table 4.3 and Figure 4.3 shown above, the optimized system response at amplifier gain of 150 rises faster and settles faster as compared to that at amplifier gain of 100 with no overshoots and undershoots. This is also in agreement with Routh-Herwitz criterion for stability as reported by Saini (2014). The problem of peaking in its transient state is seen to have increased.

(4) System Response with Amplifier Gain of 200

Table 4.4: (a) GA parameters**,** (b) Plant parameter and (c) System response parameters

S/N	Parameter	Value
1	Population	80
2	Iteration	100
3	Crossover	0.7
4	Mutation	0.2

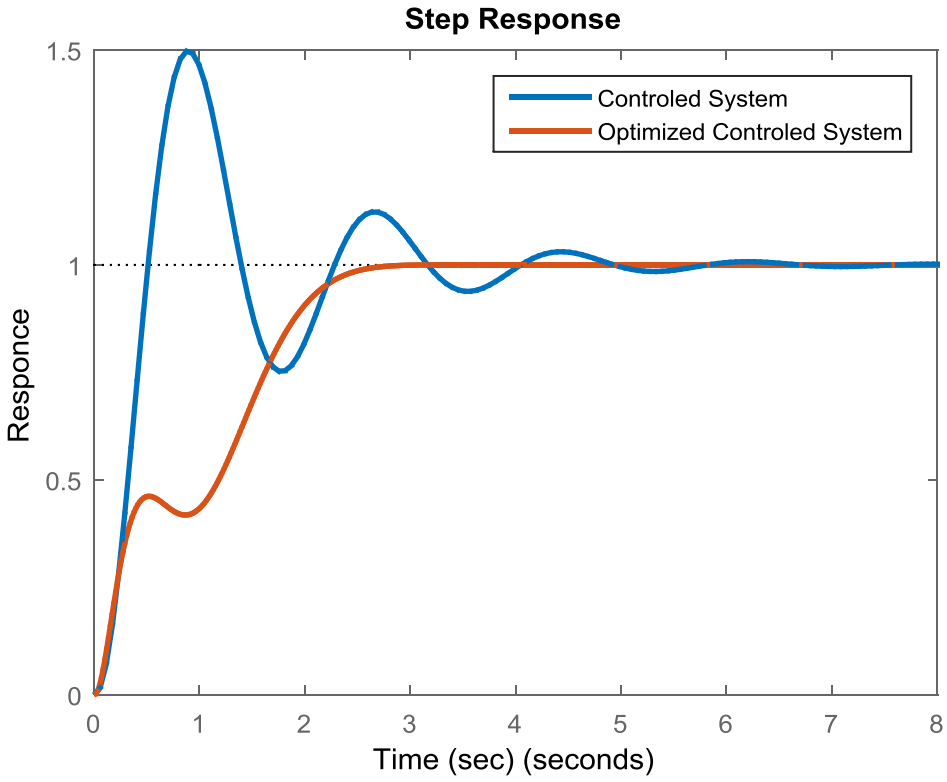
(a)

S/N	Parameter	Value
1	Amplifier gain	200

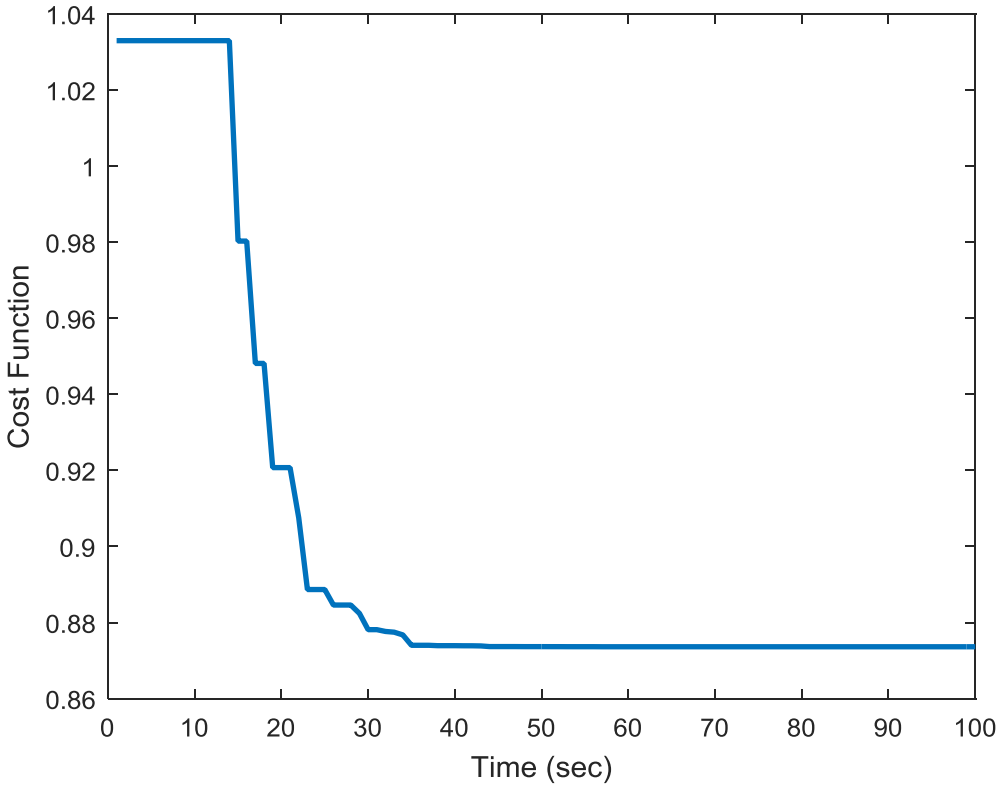
(b)

S/N	Parameter	Value
1	PID Controller gain (K_P, K_I & K_D)	0.0374, 1.0000 & 0.2073
2	Best cost	0.87362
3	Rise time	1.8619sec
4	Settling time	2.4181sec
5	Overshoot	0%
6	Undershoot	0%
7	Peak	1.0000sec
8	Peak Time	3.2930sec

(c)



(a)



(b)

Figure 4.4: (a) System response at gain value of 200, (b) GA cost function minimization for gain value of 200

From Table 4.4 and Figure 4.4 shown above, the optimized system response at an amplifier gain of 200 rises faster and settles faster as compared to that at 150 with no overshoots and undershoots. This is also in agreement with Routh-Herwitz criterion for stability as observed by Chishti *et al*, (2014) and Nise (2006). However the problem of peaking increases.

(5) System Response with Amplifier Gain of 250

Table 4.5: (a) GA parameters, (b) Plant parameter and (c) System response parameters

S/N	Parameter	Value
1	Population	80
2	Iteration	100
3	Crossover	0.7
4	Mutation	0.2

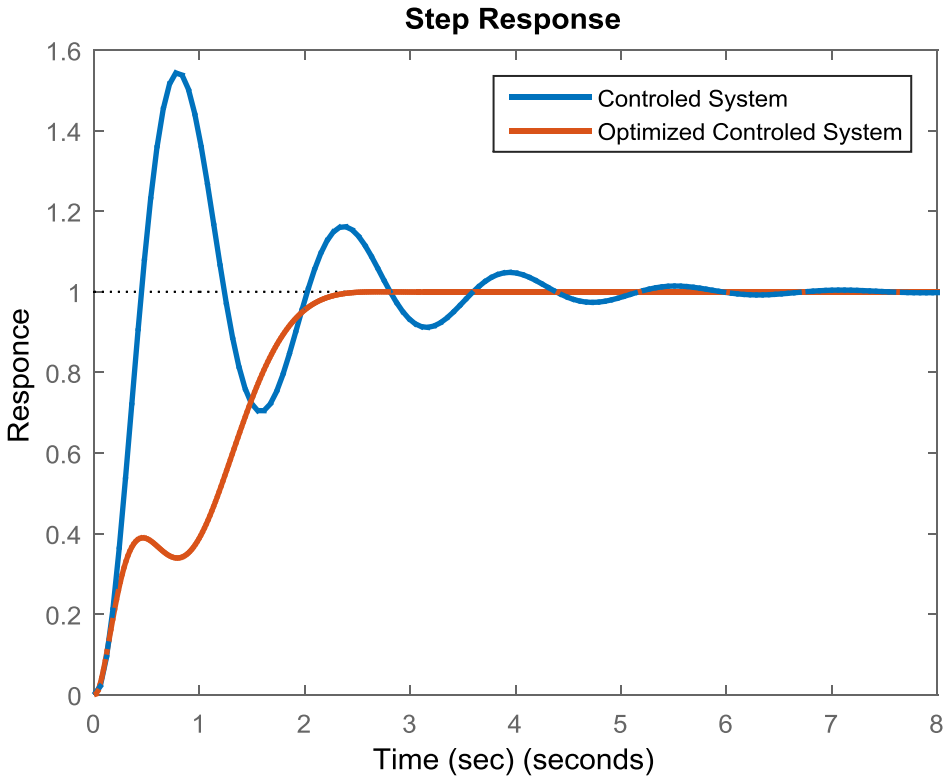
(a)

S/N	Parameter	Value
1	Amplifier gain	250

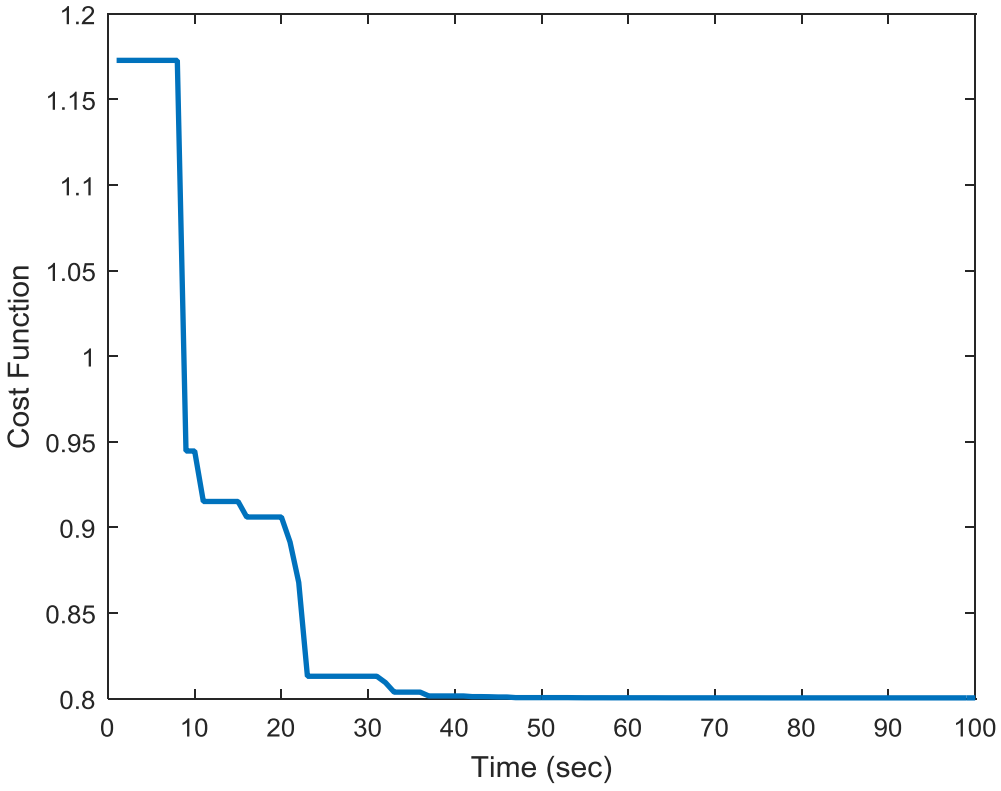
(b)

S/N	Parameter	Value
1	PID Controller gain (K_P, K_I & K_D)	-0.1679, 1.0000 & 0.0750
2	Best cost	0.80041
3	Rise time	1.6919sec
4	Settling time	2.1563sec
5	Overshoot	0%
6	Undershoot	0%
7	Peak	1.0000sec
8	Peak Time	2.7042sec

(c)



(a)



(b)

Figure 4.5: (a) System response at gain value of 250 and (b) GA cost function minimization for gain value of 250

From Table 4.5 and Figure 4.5 shown above, the optimised system response at amplifier gain of 250 produced the fastest rise and settling time at no overshoots and undershoots. This is also in agreement with Routh-Herwitz criterion for stability as observed by Nice (2006) and Ogata (2007). However, the problem of peaking is biggest at this value.

Table 4.6 below gives the summary of the optimised system responses at the different amplifier gain values.

Table 4.6: System response values at different amplifier gains

Amplifier Gain	**Rise Time (sec)**	**Settling Time (sec)**	**Overshoot (%)**	**Undershoot (%)**
50	2.1064	3.2339	0	0
100	2.2412	2.9861	0	0
150	2.0182	2.6281	0	0
200	1.8619	2.4181	0	0
250	1.6919	2.1563	0	0

4.2 System Response Values at Different Amplifier Gains

From Table 4.6 shown above, it clearly shows that the response of the system at the amplifier gain value of 50 gives the best transient characteristics ie there is no problem of peaking. However at amplifier gain of 50, it produces the longest settling time. The system response at amplifier gain of 100 gives a faster settling time and a slower rise time as compared to that at 50. This implies that the system at 100 settles faster and rises slower. However, there is a negligible problem of peaking in its transient response characteristics. At amplifier gain of 150, the system settles faster than that at gains 50 and 100 and rises faster than that at gain of 100 and slower than that at gain of 50. This implies that it's settling time is smaller and rise time is smaller than that at 100 and higher than that at 50. However, there is a significant problem of

peaking in its transient response characteristics. The response of the system at gain value of 200 settles faster and rises faster than that at 50, 100 and 150. This implies that it has the smallest settling time and rise time as compared to that at gain value of 50, 100 and 150. However, the problem of peaking is seen to have increased. Finally, the response of the system at gain value of 250 settles and rises fastest as compared to the previous gain values. This implies that it has the least settling time and rise time. However, it can be seen from the response graph that it has the biggest problem of peaking. This problem is associated with the increase in the amplifier gain values. The higher the gain value, the bigger the problem. The overall responses have zero overshoots and undershoots.

Conclusively, the system responds faster with increase in amplifier gain values which is in agreement with the work done by Nise (2006) and Chishti *et al,* (2014).

4.3 Comparison of Work

Comparison of this work is done by comparing the system response of the GA tuned PID controller with the work done by Chishti *et al*. (2014) at an amplifier gain value of 100.

Table 4.7 below gives the summary of the system response values at amplifier gain of 100 for the Zeigler-Nichols method tuned PID Controller and Genetic Algorithm tuned PID Controller.

Table 4.7: System response values at different tuning technique

Tuning Technique	**PID Controller Gain (K_p, K_i & K_d)**	**Rise Time (sec)**	**Settling Time (sec)**	**Overshoot (%)**	**Undershoot (%)**
Zeigler-Nichols tuned PID Controller	16, 2 & 5	0.8568	9.2289	66.3812	23.1264
Genetic Algorithm tuned PID Controller	0.0930, 1.0000 & 0.2905	2.2412	2.9861	0	0

From Table 4.7 shown above, it can be concluded that Zeigler-Nichols tuned PID Controller system rises faster but settles much slower with large percentage of overshoot and undershoot. While the genetic algorithm tuned PID Controller system rises slower but settles much faster with 0% overshoot and undershoot

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 Summary

Genetic Algorithm applied in tuning PID controller improves the response for Deep Space Antenna Positioning System in producing the required transient response, eliminating steady state error and achieving stability. In this work, two specific contributions have been added to knowledge: the system was successfully modelled using Bond-Graph in 20Sim environment and the PID Controller was optimized using GA in Matlab environment.

5.2 Conclusion

From the results and discussion of the study, the following conclusions can be made:

A nonlinear mathematical model of deep space antenna positioning system was developed using Bond-Graph. The open loop and closed loop model of the system shows the versatility and corresponding coherence with the system as modelled in other papers; however, via bond graph, a graphical visualization made it easily validated.

A Genetic Algorithm was developed to tune the Proportional-Integral-Derivative Controller in which its parameters were optimised and stability of the system was achieved.

From the simulation results, the response of the system with Genetic Algorithm tuned PID controller is better than the system response with conventionally tuned PID Controller ie Zeigler-Nichols method tuned PID Controller in terms of the transient response, steady state response and stability.

Conclusively, the best response is obtained at an amplifier gain of 100 in terms of transient state and steady state characteristics of the system response as it rises and settles faster than that at 50 and has a negligible problem of peaking as compared to that at 150, 200 and 250. Therefore, from the simulation results shown, the PID Controller is best designed at gain

parameter values (K_p, K_i and K_d) of 0.0930, 1.0000 and 0.2905 respectively for optimum performance.

5.3 Recommendation

For different amplifier gain values, the system response at an amplifier gain of 250 has the fastest rise time and settling time but has the biggest problem of peaking in its transient state characteristics. This problem is due to increase in gain values as it increases with increase in amplifier gain values. However, for further research:

1. The transient state response at an amplifier gain of 250 can be improved upon since this value produces the fastest response time.
2. Other optimization algorithm such as Artificial Neural Networks and Fuzzy Logic can be used to optimize the deep space antenna positioning system and results compared with that optimized using GA.

References

Agubor, C., Ndinechi, M., and Opara, R. (2013). A Practical Guide to Site Selection for Communication Antennas and thier Suport Structures in Nigeria. *Academic Research International*, 4(1): 391.

Ang, K. H., Chong, G. and Li, Y. (2005). PID control system analysis, design and technology. *IEEE Transactions on Control Systems Technology*, vol 13, no 4: pp. 555-576.

Astrom, K. J., Hagglund, T. (1995). *PID Controllers: Theory, Design and Tuning*. 2nd Edition. Society of America, Research Triangle Park.

Broenink, J. F. (1999). *Introduction to Physical Systems Modelling with Bond Graphs.* University of Twente, Dept EE, Control Laboratory. PO Box 217, NL-7500 AE Enschede Netherlands.

Chishti, A. R., Bukhari, S. F. U. R., Khaliq, H. S., Khan, M. H., and Bukhari, S. Z. H. (2014). Radio Telescope Antenna Azimuth Position Control System Design and Analysis IN MATLAB/SIMULINK using PID and LQR Controller. *Universitatea Tehnică „Gheorghe Asachi" din Iaşi*, Tomul LX (LXIV) Fasc: 3-4, Secţia

Franklin, G., Powell, F. and Emami-Naeini, A. (2002). *Feedback Control of Dynamic Systems*. 4th Ed., Addison-Wesley Publishing Company.

Gawronski, W. (2008). *Modeling and Control of Antennas and Telescopes.* Springer Science & Business Media.

Ibrahim, S. M. (2005). *The PID Controller Design using Genetic Algorithm*. University of Southern Queensland.

Liu, X. (2009). Antenna Azimuth position control system analysis and controller implementation. *International Journal of Advanced Research in Computer and Communication Engineering*, vol 4: 67-71.

Mirzal, A., Yoshii, S. and Furukawa, M. (2012). *PID Parameter Optimization by using Genetic Algorithm.* Graduate School of Information Science and Technology. Hokkaido University. Sapporo, Japan.

Nise, N.S (2006). *Textbook on Control System Engineering*. 6th Edition, John Wiley & Sons.

Ogata, K. (2007). *Modern Control Engineering*. 3rd Edition. Pearson Education, Inc., publishing as Prentice Hall, One Lake Street, Upper Saddle River, New Jersey 07458.

Okumus, H.I., Sahin, E. and Akyazi, O. (2012). Antenna Azimuth Position Control with Classical PID and Fuzzy Logic Controllers. *IEEE Transaction on Education,* 2(3): 7

Okumus, H.I., Sahin, E. and Akyazi, O. (2014). Antenna Azimuth Position Control with fuzzy logic controllers and self-tuning fuzzy logic controllers. *IEEE Transaction on Education,* 5(4): 8

Pillai, R. P., Jadhav, S. P., and Patil, M. D. (2013). Tuning of PID controllers using advanced genetic algorithm. *International Journal of Advanced Computer Science and Applications (IJACSA),* 1(1): 6.

Saini, R. (2014). *A Comparative Study of various Controllers for Azimuth Position Control of Reduced Order Antenna System.* Department of Electrical and Instrumentation Engineering, Thapar University, Patiala.

Samantaray, A. K., and Bouamama, B. O. (2008). *Model-based process supervision: A bond graph approach.* Springer Science & Business Media.

Sheth, S. S. and Gonsai, S. K. (2012). Antenna Position Control Systems, Review and New Perception. *Journal of Information, Knowledge and Research in Electronics and Communication Engineering*, (0975-6779): 2.

Wellstead, P. E. (1979). *Introduction to physical system modelling*. Academic Press London.

Xiaoping, W., and Liming, C. (2002). *Genetic algorithm-theory, application and software implementation [M]*. Xi'an Jiao tong University Press, Xi'an.

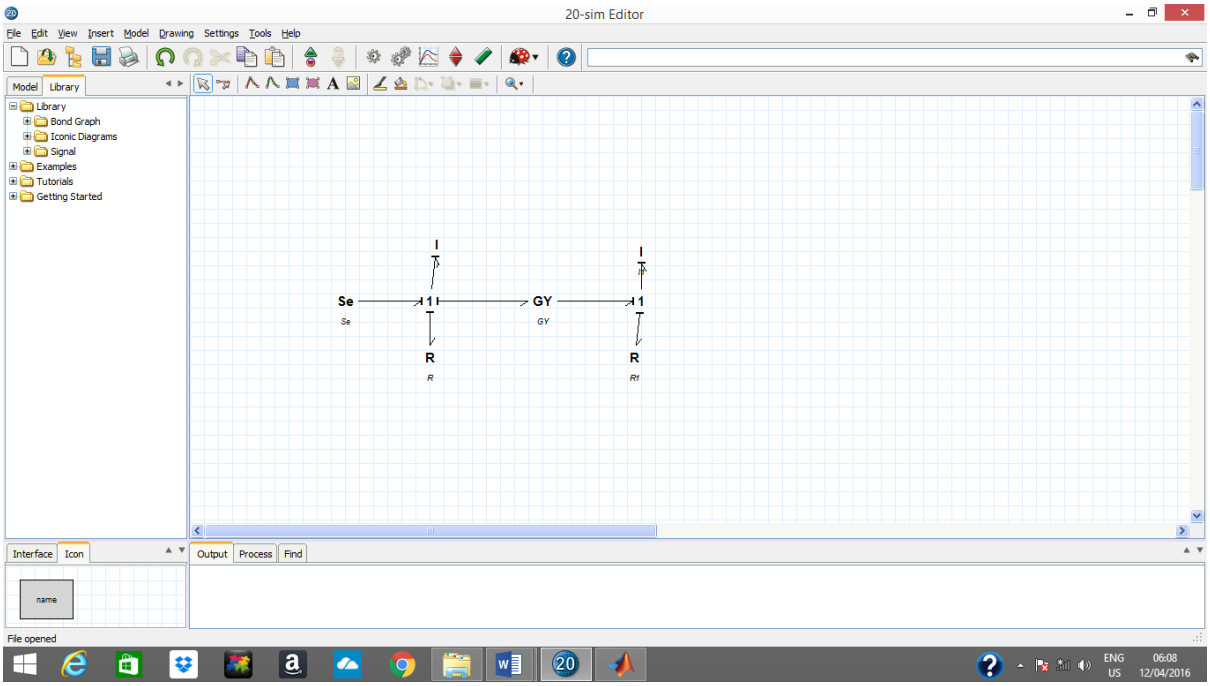
Xuan, L., Estrada J., and Digiacomandrea J (2009). Antenna Azimuth Position Control System Analysis and Controller Implementation. Term Project. *International*

Journal of Advanced Research in Computer and Communication Engineering, 1(1): 15

Zhang, J., Zhuang, J., and Du, H. (2009). Self-organizing genetic algorithm based tuning of PID controllers. *Information Sciences*, 179(7), 1007-1018.

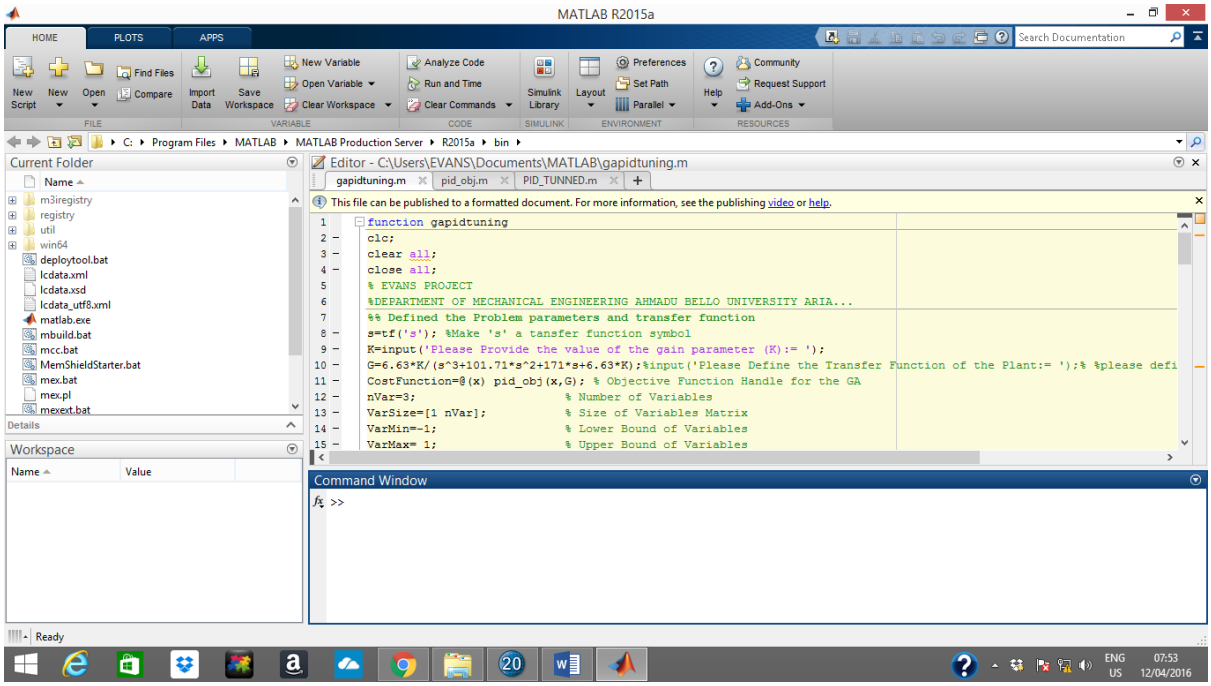
APPENDIX A

20SIM SOFTWARE INTERFACE



APPENDIX B

MATLAB INTERFACE



APPENDIX C

CODES FOR GENETIC ALGORITHM (GA) IMPLEMENTATION

```
function gapidtuning
clc;
clear all;
close all;
% EVANS PROJECT
%DEPARTMENT OF MECHANICAL ENGINEERING AHMADU BELLO UNIVERSITY
ARIA...
%% Defined the Problem parameters and transfer function
s=tf('s'); %Make 's' a tansfer function symbol
K=input('Please Provide the value of the gain parameter (K):= ');
G=6.63*K/(s^3+101.71*s^2+171*s+6.63*K);%input('Please Define the Transfer Function of
the Plant:= ');% %please defined the plant transfer function
CostFunction=@(x) pid_obj(x,G); % Objective Function Handle for the GA
nVar=3;                 % Number of Variables
VarSize=[1 nVar];       % Size of Variables Matrix
VarMin=-1;              % Lower Bound of Variables
VarMax= 1;              % Upper Bound of Variables
VarRange=[VarMin VarMax];   % Variation Range of Variables
%% GA Parameters
MaxIt=100;% Maximum Number of Iterations
nPop=input('Please Provide the population Size of Genes:= ');%50;% Population Size
pCrossover=input('Please provide the Percentage of Cross Over Operator:= ');%0.7;%
Crossover Percentage
nCrossover=round(pCrossover*nPop/2)*2;  % Number of Parents (Offsprings)
pMutation=input('Please Provide the Mutant percentage:= ');%0.2;% Mutation Percentage
nMutation=round(pMutation*nPop);% Number of Mutants
%% Initialization
% Empty Structure to Hold Individuals Data
ini_ind.Position=[];
ini_ind.Cost=[];
ini_ind.Out=[];
% Create Population Matrix
pop=repmat(ini_ind,nPop,1);
% Initialize Positions
for i=1:nPop
   pop(i).Position=unifrnd(VarMin,VarMax,VarSize);
   [pop(i).Cost, pop(i).Out]=CostFunction(pop(i).Position);
end
% Sort Population
pop=SortPop(pop);
% Store Best Solution
BestSol=pop(1);
% Vector to Hold Best Cost Values
BestCost=zeros(MaxIt,1);
%% GA Main Loop
for it=1:MaxIt
   % Crossover
   popc=repmat(ini_ind,nCrossover/2,2);
   for k=1:nCrossover/2
```

```
        i1=randi([1 nPop]);
        i2=randi([1 nPop]);
        p1=pop(i1);
        p2=pop(i2);
        [popc(k,1).Position, popc(k,2).Position]=Crossover(p1.Position,p2.Position,VarRange);
        [popc(k,1).Cost, popc(k,1).Out]=CostFunction(popc(k,1).Position);
        [popc(k,2).Cost, popc(k,2).Out]=CostFunction(popc(k,2).Position);
    end
    popc=popc(:);
    % Mutation
    popm=repmat(ini_ind,nMutation,1);
    for k=1:nMutation
        i=randi([1 nPop]);
        p=pop(i);
        popm(k).Position=Mutate(p.Position,VarRange);
        [popm(k).Cost, popm(k).Out]=CostFunction(popm(k).Position);
    end
    % Merge Population
    pop=[pop
         popc
         popm];
    % Sort Population
    pop=SortPop(pop);
    % Delete Extra Individuals
    pop=pop(1:nPop);
    % Update Best Solution
    BestSol=pop(1);
    % Store Best Cost
    BestCost(it)=BestSol.Cost;
    % Show Iteration Information
    disp(['TJ_Itr ' num2str(it) ': Best Cost = ' num2str(BestCost(it))]);

end
    % Plot Step Response
    figure(1);
    step(G);
    hold on
    step(BestSol.Out.T);
    legend('Controled System','Optimized Controled System')
    xlabel('Time (sec)')
    ylabel('Responce')
    stepinfo(BestSol.Out.T)
%% Plots
figure;
plot(BestCost);
xlabel('Time (sec)')
ylabel('Cost Function')
```

APPENDIX D

SIMULATION RESULTS