Design of Genetic Algorithm Based Fractional Order Controller for Temperature Control of Electric Furnace for Glass Tempering Process

Eyosiyas Ayele Gebre

A Thesis Paper Submitted to

The Department of Electrical Power and Control Engineering
School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of Master's in Electrical Power and Control Engineering (Control Engineering)

Office of Graduate studies
Adama Science and Technology University

July, 2021
Adama, Ethiopia

Design of Genetic Algorithm Based Fractional Order Controller for Temperature Control of Electric Furnace for Glass Tempering Process

Eyosiyas Ayele Gebre

Advisor: Dr. Tefera Terefe Yitayew

A Thesis Paper Submitted to
The Department of Electrical Power and Control Engineering
School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of Master's in Electrical Power and Control Engineering (Control Engineering)

Office of Graduate studies
Adama Science and Technology University

July, 2021
Adama, Ethiopia

APPROVAL PAGE

I, the advisors of the thesis entitled “Design of Genetic Algorithm Based Fractional Order Controller for Temperature Control of Electric Furnace for Glass Tempering Process” and developed by Eyosiyas Ayele, hereby certify that the recommendation and suggestions made by the board of examiners are appropriately incorporated into the final version of the thesis.

__________________________	____________________	____________________
Advisor	Signature	Date

We, the undersigned, members of the Board of Examiners of the thesis by Eyosiyas Ayele have read and evaluated the thesis entitled “Design of Genetic Algorithm Based Fractional Order Controller for Temperature Control of Electric Furnace for Glass Tempering Process” and examined the candidate during the open defense. This is, therefore, to certify that the thesis is accepted for partial fulfillment of the requirement of the degree of Master of Science in Electrical Power and Control Engineering (Control Engineering).

Chair Person	Signature	Date
Internal Examiner	Signature	Date
Externa Examiner	Signature	Date

Finally, approval and acceptance of the thesis are contingent upon submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and School Graduate Committee (SGC).

Head of Department	Signature	Date
School Dean	Signature	Date
Postgraduate Dean	Signature	Date

DECLARATION

I hereby declare that this Master Thesis entitled "Design of Genetic Algorithm Based Fractional Order Controller for Temperature Control of Electric Furnace for Glass Tempering Process" is my original work. That is, it has not been submitted for the award of any academic degree, diploma, or certificate in any other university. All sources of materials that are used for this thesis have been duly acknowledged through citation.

________________________	____________________	____________________
Name of Student	Signature	Date

RECOMMENDATION

I, the advisor of this thesis, hereby certify that I have read the revised version of the thesis entitled "Design of Genetic Algorithm Based Fractional Order Controller for Temperature Control of Electric Furnace for Glass Tempering Process" prepared under my guidance by Eyosiyas Ayele submitted in partial fulfillment of the requirements for the degree of Mater's of Science in Electrical Power and Control Engineering (Control Engineering). Therefore, I recommend the submission of a revised version of the thesis to the department following the applicable procedures.

____________________________	______________________	______________________
Advisor	Signature	Date

ACKNOWLEDGEMENTS

First of all, I would like to thank the almighty of **GOD**, who gave me strength, patience, and courage to complete the thesis successfully.

I would like to express my deepest gratitude to my advisor **Dr. Tefera Terefe** for his support of my thesis, for his patience, motivation, interest, and most importantly his guidance helped me in all times of writing, modeling, designing, and simulating the thesis document.

And also I would like to express my deepest gratitude to **Dr. Milkias Berhanu** for his patience, helpful attitude, important ideas, guidance, and unwavering support throughout the document writing and designing of the thesis.

Furthermore, I would also like to express thanks to the EPCE department head **Mr. Mesifin Megra (MSc.),** and laboratory staff members for encouraging, providing access to computer labs for discussion when I have a meeting with my advisor.

Finally, I would like to thank my whole family, and words that cannot express how grateful I am to my parents and friends for their love, encouragement, and all of those who paid sacrifices that they have made on my behalf.

TABLE OF CONTENTS

APPROVAL PAGE i

DECLARATION ii

RECOMMENDATION iii

ACKNOWLEDGEMENTS iv

LIST OF TABLES viii

LIST OF FIGURES ix

LIST OF ACRONYMS xi

ABSTRACT xii

CHAPTER ONE 1

INTRODUCTION 1

1.1. Background 1

1.2. The Problem Statement 2

1.3. Objective of the Thesis 2

1.3.1. General Objective of the Thesis 2

1.3.2. Specific Objectives of the Thesis 2

1.4. Motivation of the Study 3

1.5. Scope of the Study 3

1.6. Limitation of the Study 3

1.7. Significance of the Study 3

CHAPTER TWO 4

LITERATURE REVIEW 4

2.1. Introduction of Glass Tempering Process 4

2.1.1. History of Tempered Glass 4

2.1.2. Tempering Process 4

2.1.3. Glass Properties 5

2.2. Resistance Heating and Melting Furnace 7

2.2.1. Direct Resistance Heating Furnace 8

2.2.2. Indirect Resistance Heating Furnace 8

2.3 Related Works 8

CHAPTER THREE 11

METHODOLOGIES AND MATERIALS 11

3.1. Materials 11

3.2. Methods 11

3.3. Model of the Glass Tempering Process..12
3.4. Temperature Control Systems of Resistance Furnace..13
3.4.1. PID Controller..13
3.4.2. Algorithm of PID Controller..13
3.4.3. The PID Controller Limitation..15
3.5. Genetic Algorithm..16
3.6. Fractional Order Calculus..16
3.6.1 Fractional Order PID Controller..17
3.7. Model of Electric Furnace..18
3.8. Model of the Delay Time..19
3.9. Mathematical Model and Analysis..20
3.10. Model of the Temperature Sensor..23
3.11. Design of PID Controller..25
3.11.1. Loop Tuning Mechanism..25
3.11.2. Manual Tuning..25
3.11.3. Conventional PID Tunings Method..26
3.12. Digital controller..32
3.13. MATLAB/Simulink Model of the System..35
3.13.1. MATLAB/Simulink Model of Open-Loop System..35
3.13.2. MATLAB/Simulink Model of PID Controller..35
3.13.3. GA Tuned of PID Controller..36
3.13.4. MATLAB/Simulink Model of Fractional Order Controller..37
3.13.5. GA Tuned of FOPID Controller..38
3.14. Design of Model Reference Adaptive Control..39
3.14.1. MIT Rule..42
3.14.2. Lyapunov Stability Method..43
3.14.3. RHP Zeros..45
3.14.4. Design of Feed-Forward Control..46
3.15. MATLAB/Simulink Model of MRAC with Feedforward PI Controller..46
3.15.1. MATLAB/ Simulink Model of Open-Loop Reference Model..46
3.15.2. MATLAB/Simulink Model of MIT Rule with FFPI Controller..47
3.15.3. MATLAB/Simulink Model of Lyapunov Rule with FFPI controller..47
CHAPTER FOUR..48
RESULTS AND DISCUSSIONS..48
4.1. Open-Loop Response of Electric Furnace with Time Delay Consideration..48

4.2. The Result of Open Loop system Response of the Model Reference 49
4.3. The Result of GA Based PID Controller 49
4.4. The Result of MIT Rule without Feed-Forward Controller 50
4.5. The Result of MIT Rule with Feed-Forward PI Controller 51
4.6. The Result of Lyapunov Rule without Feed-Forward Controller 51
4.7. The Result of Lyapunov Rule with Feed-Forward PI Controller 52
4.8. The Result Comparison of MIT Rule with FFPI and Lyapunov Rule with FFPI 53
4.9. The Result of GA Based Fractional Order Controller 54
4.10 Controller Results for Parameter Variation 54
4.10.1 The Result MIT Rule with Feed-Forward Control for Parameter Variation 54
4.10.2. Lyapunov Rule with Feed-Forward Controller for Parameter Variation 55
4.10.3. The Result of GA Based PID Controller for Parameter Variation 56
4.10.4. The Result of GA Based FOPID Controller for Parameter Variation 56
4.11. The Result of System with Impulse Disturbance 57
4.11.1. The Result of PID Controller with Impulse Disturbance 57
4.11.2. The Result of Fractional Order Controller with Impulse Disturbance 58
4.11.3. Comparison of GA-FOPID and GA-PID Controller with Impulse Disturbance .. 58
4.12. The Response of Control Signal 59
4.13. Comparison of the Results of GA Based FOPID and GA Based PID 60
4.14. Comparison of GA Based FOPID, GA Based PID, and MRAC Controllers 60
5. CONCLUSIONS AND RECOMMENDATIONS 62
5.1 Conclusions 62
5.2 Recommendations 63
6. REFERENCES 65
7. APPENDIXIES 68
7.1. APPENDIX A 68
7.2. APPENDIX B 70

LIST OF TABLES

Table 3.1 Effects of independent P, I, and D tuning on closed-loop response 26
Table 3.2 ZN PID controller parameter for first order plus dead time model 27
Table 3.3 Cohen Coon PID controller parameter for first order plus dead time model 28
Table 3.4 ITAE PID controller parameter for first order plus dead time model 29
Table 3.5 The specifications of ZN, Cohen coon, setpoint, and load point change 31
Table 3.6 Parameter settings of the genetic algorithm optimization 36
Table 3.7 The GA optimized parameters of the PID controller...37
Table 3.8 Parameter settings of the genetic algorithm optimization 38
Table 3.9 The GA optimized parameters of the FOPID controller...39
Table 4.1 Performance of MIT rule and Lyapunov rule with FFPI...53
Table 4.2 Performance specifications for parameter variations of controller..........................57
Table 4.3 Performance of GA based FOPID and PID with impulse disturbance....................59
Table 4.4 The Time-domain specifications of FOPID-GA and PID-GA controllers..............60
Table 4.5 Performance of MIT rule, Lyapunov rule, GA based FOPID, and PID 61

LIST OF FIGURES

Figure 2.1 Block diagram of the thermal process of glass tempering 5
Figure 2.2 viscousness of Glass from 500^0C to 730^0C 6
Figure 3.1: Block diagram of the research methodology...12
Figure 3.2 General block diagram of temperature control of furnace for glass tempering 13
Figure 3.3 The PID controller block diagram...15
Figure 3.4 The flow chart for implementation of Genetic Algorithm 16
Figure 3.5 schematic diagram of FOPID controller..18
Figure 3.6 Step response for choosing L and T parameters utilizing the ZN method 27
Figure 3.7 System response using PID controller of ZN and Cohen-Coon tuning.................28
Figure 3.8 System step response using ZN, Cohen Coon, and ITAE load setting tuning 30
Figure 3.9 Responses of ZN, Cohen Coon, setpoint change, and load point change 31
Figure 3.10 Bode diagram for first-order Pade approximation...33
Figure 3.11 Bode diagram for third-order Pade approximation 34
Figure 3.12 Step response of the original and Pade Approximated transfer function 34
Figure3.13 The Simulink model of the plant without controller 35
Figure 3.14 Simulink model of the electric furnace with PID controller 36
Figure 3.15 Fitness values vs generation for object function 37
Figure 3.16 Simulink model of the electric furnace with FOPID controller 38
Figure 3.17 Fitness values vs generation for object function of FOPID 39
Figure 3.18 Model Reference Adaptive Control structure..40
Figure 3.19 The Model of MRAC using MIT rule 43
Figure 3.20 The Block diagram of MRAC using Lyapunov rule 44
Figure 3.21 combination of feedback and feed-forward control design.................................46
Figure 3.22 The Simulink model of open-loop SISO Reference Model 46
Figure 3.23 Simulink model of MIT rule with Feed-Forward controller 47
Figure 3.24 Simulink model of Lyapunov rule with Feed-Forward controller 47
Figure 4.1 The open-loop step response with time delay consideration.................................48
Figure 4.2 Step response of the open-loop system of the model reference 49
Figure 4.3 The Response of GA-based PID temperature Controller of the system.................50
Figure 4.4 The response of MIT rule without FFPI Controller 50
Figure 4.5 The system response for MIT Rule with FFPI Controller.....................................51
Figure 4.6 The system response of Lyapunov rule without FFPI controller 52
Figure 4.7 The Lyapunov rule with FFPI controller system response....................................52

Figure 4.8 Comparison of MIT rule and Lyapunov rule of the system responses....................53
Figure 4.9 The result of GA based FOPID Controller..54
Figure 4.10 The MIT rule with feed-forward PI controller for parameter variation 55
Figure 4.11 The Lyapunov rule with feed-forward PI controller for parameter variation 55
Figure 4.12 PID with GA Tuned controller for parameter variation.......................................56
Figure 4.13 GA based FOPID for parameters variations...57
Figure 4.14 The result of GA based PID with impulse disturbance..58
Figure 4.15 The result of GA based FOPID with impulse disturbance...................................58
Figure 4.16 The result of GA based FOPID, and PID with impulse disturbance....................59
Figure 4.17 The Control signal of the PID controller..59
Figure 4.18 Comparison system responses of GA-FOPID and GA-PID.................................60
Figure 4.19 Comparison of FOPID-GA, PID-GA, Lyapunov rule, and MIT rule 61

LIST OF ACRONYMS

FFMRAC	Feed Forward Model Reference Adaptive Controller
FFPI	Feed Forward Proportional Integration
FLC	Fuzzy Logic Control
FOPID	Fractional Order Proportional Integration Derivation
GA	Genetic Algorithm
MRAC	Model Reference Adaptive Controller
ODE	Ordinary Differential Equation
PI	Proportional Integral
PID	Proportional Integration Derivation
SISO	Single Input Single Output
SMC	Sliding Mode Controller
ZN	Ziegler Nichols
ITAE	Integral Time Absolute Error

ABSTRACT

Safety glass or Tempered glass is made through the process by chemical action or thermal method to increase its strength compared through the normal or annealed glass. Therefore, in the thermal tempering industrial process, the temperature control of the electric furnace is the main problem. This thesis is proposed as a Genetic Algorithm based Fractional Order method designed to control the temperature of the electric furnace for the tempering process of glass. The simulation results show that the Genetic Algorithm-based Fractional Order controller can provide improved closed-loop system performances. The electric furnace is nonlinear and time-varying processes, using manual tuning PID controller is difficult to meet the control requirement. Because it is not robust, has not good dynamic response, and has a large percentage overshoot. Hence Genetic Algorithm based Fractional Order PID (GA-FOPID) is used to improve and optimize the system response, achieve steady-state and transient requirements.

Therefore, in this thesis, a model of the glass tempering electric furnace is prepared using MATLAB/Simulink, and then FOPID controller with optimizing algorithm GA has been designed and simulated and its performance has been compared with Model Reference Adaptive Control (MRAC), and parameter variations of each controller. The performance specifications of each controller are compared. It is observed that from the system response and performance specifications, the GA-based FOPID controller is used to improved and optimized the system output and transient requirements to control the temperature of the electric furnace for the tempering process of glass.

Keywords: *Electric Furnace, Fractional Order, Genetic Algorithm, Glass Tempering, Model Reference Adaptive Control, PID, Temperature Control.*

CHAPTER ONE

INTRODUCTION

1.1. Background

Tempering glass could be a strong glass processed by thermal procedure or chemical action to extend its strength associated with the standard annealed glass. Generally, tempered glass is about four times stronger than non-tempered (standard) glass. This tempered glass property causes the broken glass to crumble into little granular chunks instead of break into rough shards. For example, tempered glass is working in environments wherever safety is a critical issue. The tempered glass applications are the on windows of cars, entry of doors, shower, different home furniture, microwave furnaces, and skylights, etc. [1], [2], [3].

Purposely tempered glass was, to begin with, licensed in July 1874 by Frenchman Francois Royer de la Beastie. But not progress until the early 1900s and the primary patent for the method to deliver tempered glass was created by Rudolf Seiden, an Austrian chemist who moved to the United States in 1935. Before this time the as it was recorded utilize of purposely tempered glass was the generation of Sovereign Rupert Drops which were novelty things [4].

The tempered glass is made by the thermal process of tempering method. The thermal process of the glass tempering method uses an electrical furnace for heating the glass. An additional method for creating glass tempering is the chemical process method. The thermal tempering process is the easiest method, this is done by heating the glass overhead the transition temperature, preserve the glass optically flat, form it a shape, then consistently cooling it [5].

To prepare glass for the tempering process, it must first be cut to the desired size (strength reductions or product failure can occur if any fabrication operations, such as etching or edging, take place after heat treatment.) The glass is then examined for imperfections that could cause breakage at any step during tempering. Subsequently, the glass starts a heat conduct method within which it travels through a tempering furnace, either in an exceedingly batch or continuously. The glass heated by the furnace during the tempering process in the temperature range is between 610 degrees Celsius and 640 degrees Celsius. After it passes through this high temperature, the glass undergoes a powerful cooling process known as a termination. Throughout this method, which lasts simply seconds, forceful cold air blasts the surface of the glass from the associate in a nursing array of nozzles in variable positions. The outer surface of the glass cools more quickly than the middle surface of the glass because when the center of the glass cooled, the outer surfaces pulled back. As consequence, the middle remains in tension, and also the outer surfaces rise in compression, which delivers the tempered glass its strength. Since the electric furnace is a nonlinear and time-varying process, using PID controllers is difficult to meet the control requirement. Because they are not robust, have not good dynamic response, and have large overshoot. Hence in this thesis, the GA-based PID, and GA-based Fractional Order controllers are used to improve and achieve steady-state and transient requirements.

1.2. The Problem Statement

Temperature control of the electric furnace is the main part and has a significant impact on the quality of products in the thermal tempering glass production process. Thus, the temperature must be controlled very well in the glass tempering furnace to produce tempered glass with high quality. The temperature change among different chunks of the glass surface should be very small, and the glass surface should be heated.

The controller that is widely used to control the temperature in glass tempering industries is the conventional PID controller which has not good dynamic response, long rise time, and large overshot is used to control the temperature. This indicates the system is not optimized and tuned. Generally, many authors used different techniques to control temperature like, PID, Fuzzy, Fuzzy-PID, and SMC controllers. These controllers should be tuned and optimized. So, an efficient controller that can work on an extensive variety of working situations and is adaptable to the variation of system procedure is required. Therefore, this thesis proposed GA based FOPID controller to solve the drawback (longer rise time and large percentage overshot) of the existing controller and improve system performance.

1.3. Objective of the Thesis

1.3.1. General Objective of the Thesis

The main objective of this thesis is to design and simulate the temperature control of an electric furnace that is used for the glass tempering process.

1.3.2. Specific Objectives of the Thesis

This thesis work has the following specific objectives:

- ✓ To develop the model of temperature control of the electric furnace for glass tempering.
- ✓ To decrease the temperature difference during the glass tempering process.
- ✓ To design the temperature controller of the furnace for the tempering process of glass.
- ✓ To optimize, performance measure, and tuning FOPID controller for glass tempering.

1.4. Motivation of the Study

Due to the futurity of glass factory technology, it is very interesting to research in this area. At present glass is widely needed for different purposes like windows, doors, buildings, furniture, vehicles, etc. Thus, the glass must be high quality and good strength but the quality of glass is affected by temperature because of an immediate increase or decrease in temperature during the production process. The temperature controlling method must be optimized and tuned. Tempered glass is strong and used as safety glass which is widely used in the world. So, to get these glasses, temperature control plays a great role.

1.5. Scope of the Study

In this thesis work, to have a good understanding of the Genetic Algorithm based on Fractional Order PID techniques to design a controller of the temperature of the electric furnace for the tempering process of glass. The thesis work will focus on MATLAB/Simulation of the MRAC, PID, and Fractional Order controllers are utilized as the benchmark to illustrate the performance of the system, and GA is used to tune gains of Fractional Order and PID gains to get the best response of the system.

1.6. Limitation of the Study

The thesis is limited to the simulation of the system using MATLAB/Simulink by showing how the different apparatuses of the system are arranged with each other. This is because the implementation of the prototype is difficult. Because the materials of the system components are expensive and some components are not found in the country.

1.7. Significance of the Study

The proposed thesis of temperature control for the electric furnace of tempering glass process can contribute the following advantages.

- ✓ It minimizes the overheating or cooling of the glass.
- ✓ It protects the materials from damage.
- ✓ It increases the reliability or accuracy of the system.
- ✓ It improves the efficiency of the system.

CHAPTER TWO

LITERATURE REVIEW

2.1. Introduction of Glass Tempering Process

2.1.1. History of Tempered Glass

Purposely tempered glass was, to begin with, licensed in July 1874 by Frenchman Francois Royer de la Beastie. But not progress until the early 1900s and the primary patent for the method to deliver tempered glass was created by Rudolf Seiden, an Austrian chemist who moved to the United States in 1935. Before this time the as it was recorded utilize of purposely tempered glass was the generation of Sovereign Rupert Drops which were novelty things [1]. Back within the early 1600s, a German in Mecklenburg found that in case he dropped liquid glass into a bucket of cold water, he delivered tadpole-shaped pieces of glass that were unbreakable until the tail was broken. When the tail was broken the tadpole of glass at that point vanished, as if by enchantment, throughout a cloud of dirt [1].

The tempered glass contains a more essential mechanical quality and warm resistance than warm invigorated glass. It is around four to five times more strength than typical or non-tempered glass. Within the occasion of breakage, tempered glass will break with a tall vitality discharge of little fine circular pieces. Within the 1940s, Henry Passage started utilizing tempered glass for the side and raise windows of his automobiles and its utilize in vehicles got to be commonplace within the 1950s [1], [3].

2.1.2. Tempering Process

Tempered glasses can be made from the reinforced glass through a warm treating handle. Within the warm hardening process, utilizing an electric heater for heating the glass. Another approach for making tempered glass is chemical hardening. The thermal tempering process is very simple, just heat the glass very well above the transition temperature, maintain the glass optically flat, or form it to a shape if required, then uniformly cool it. So that the temperatures of the top and bottom surfaces are equal, and lower than the temperature at the center plane of the glass, as it cools to below the strain temperature, then cool it down to ambient temperature [4], [5], [6].

Whether hardening huge structural boards, car glasses, or little exceedingly focused mechanical glasses, the essential hardening process is the same heat the glass to a temperature over its Move Point, where it gets to be plastic, keep it optically level at that point cool it at a controlled rate to a temperature below the Strain Point. So, there's a temperature qualification between the surfaces and the central plane of the glass since it gets to be solid once more. The differential compressions between the surface and the central plane will show the extreme extend of the glass since it cools down to the encompassing temperature [1], [7].

To get ready glass for the treating handle, it must start with being cut to the desired evaluation. Glass strength reductions or product failure can occur if any fabrication operations, such as etching or edging, take place after heat treatment. The glass is at that point assessed for imperfections which will cause breakage at any step after treatment. A grinding such as sandpaper takes sharp edges off the glass, which is along these lines washed [3], [4].

Another, the glass begins a warm treatment handle in which it voyages through a solidifying heater, either in a bunch or ceaseless support. The heater warms the glass to a temperature of treating run between 610^0c to 640^0c. The glass at that point encounters a high-pressure cooling procedure called quenching. Among this handle, which keeps going reasonable second, high-pressure discourse impacts the surface of the glass from a cluster of gushes in moving positions. Quenching cools the outside surfaces of the glass much more quickly than the center. As the center of the glass cools, it tries to drag back from the outside surfaces. As a result, the center remains in pressure, and the outside surfaces go into compression, which gives the tempered glass its quality [1], [4], [8].

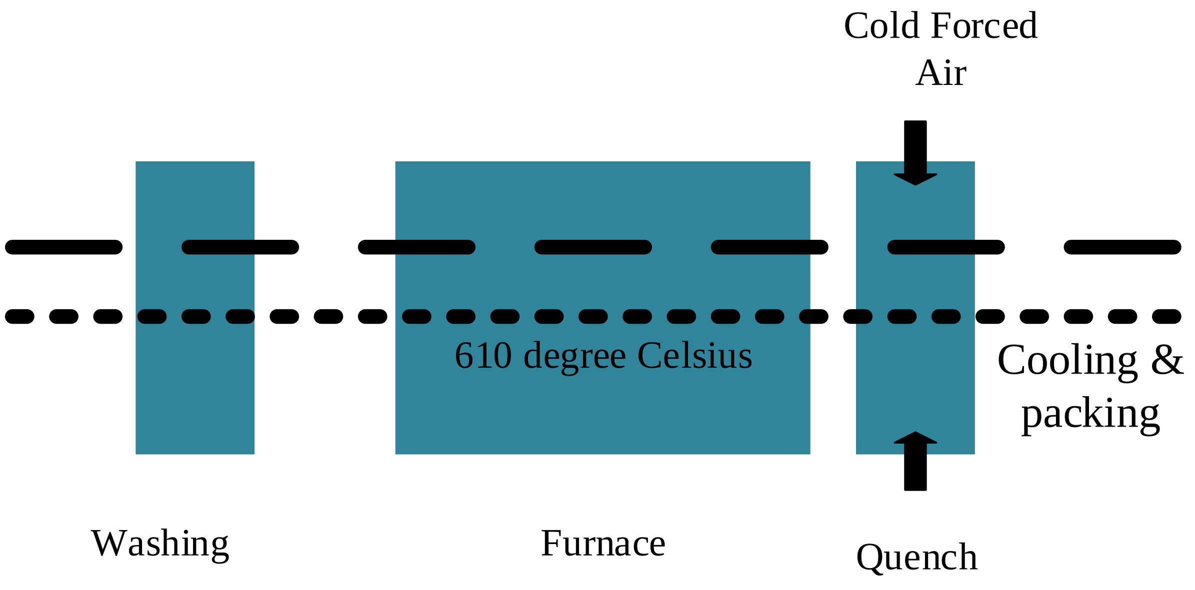


Figure 2.1 Block diagram of the thermal process of glass tempering

2.1.3. Glass Properties

Clear glass is not clear, a 6mm-thick piece of clear coast glass will capture around 13-percent of light interior the unmistakable extend, allowing 87% of the unmistakable light to pass through it. In any case, as the wavelength of light moves to miss from the self-evident run the transmission changes, and for various frequencies, glass is exceptionally thick [1], [3].

When the glass is tempered, all the external surfaces, as well as the sides of the glass, unit of measurement are put into compression. The internal parcel of the glass can get in tension, so all the strength units of measurement are balanced, however as a result of the surfaces of the strain layer unit inside the glass it must not have any micro-cracks unless there is relate the degree of pollution or air-bubble blessing, and in this way, the strain layer must not come up short. This makes tempered glass extreme to hinder compared to crude glass and it is regularly thought of to be four to five times more grounded than raw glass [1].

As the temperature of the glass is increased above the strain point and past the transition point, the glass starts to soften. That is the viscosity of the glass reduces to a level where stresses can start to relieve and the glass enters a plastic state. In the plastic state, any solid can be distorted by a force and the distortion will remain in place, unlike in its elastic state when it will return to its original shape after the force is removed [1], [3].

The viscosity of the glass reduces as it is heated up and, above the transition point. It will overcome the matter that is seen in case the glass was cooled sometime recently it comes to a sufficient temperature, the upper the temperature the extra the particles vibrate, additionally a reduced amount of inflexible of the glass [1], [7].

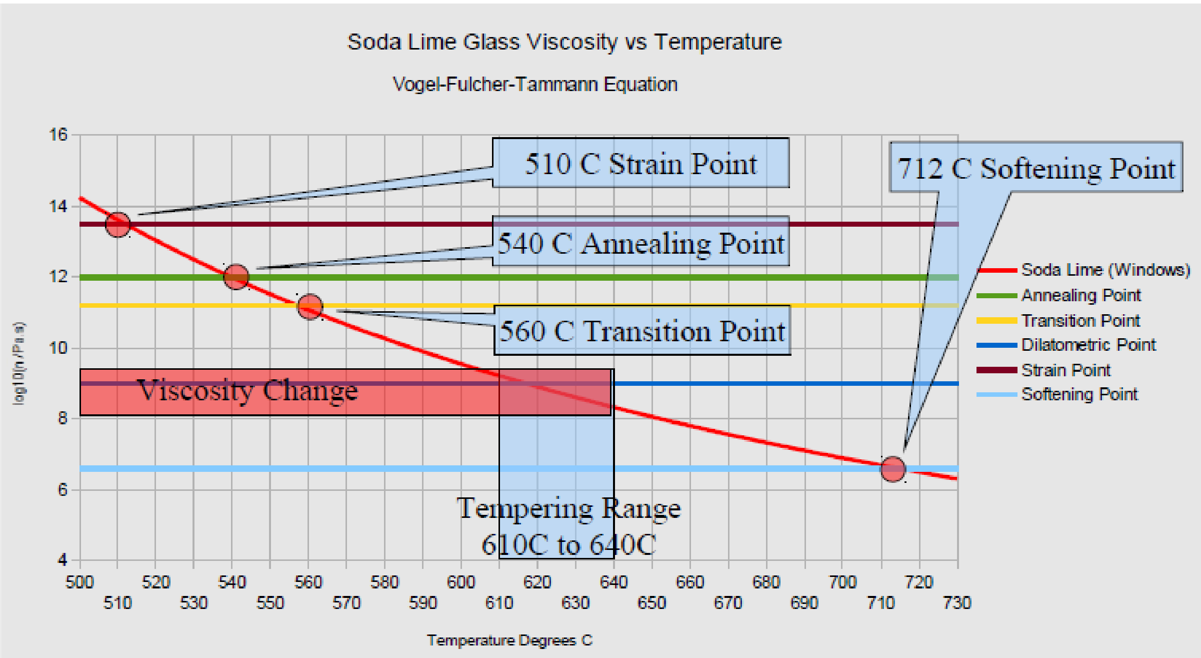


Figure 2.2 viscousness of Glass from 500^{0}C to 730^{0}C [7]

In case the glass is heated to beat the transition reason (characterized since the reason wherever any stretch inside the glass can rapidly scatter) it is found that the glass may be quenched or cooled whereas not it breaking. This may be as a result of the delicate surfaces (the unit of measurement like plasticizing) that can be extended concerning the central mass of the glass as long as their temperature is on the beat of the Transition reason. Even though the surfaces unit cooler than the center and should be shorter, they are in fact at proportionate length since of the hotter center plane of the glass. When the glass is warmed right up to it is softening reason a piece of glass can mutilate lower its weight and the temperature required is 712^0C [1], [3].

As made public on top of for tempering method, the glass has to be exiting the chamber at well on top of the Transition purpose. This demand defines the operating variables of temperature for the tempering branch of knowledge glass and is often around 610^0C to 640^0C of the glass will be heated and it lies across the dilatometric softening purpose. The glass just has to heat toward a better temperature to create it softer, if it is needed to be shaped, for instance, automotive surface and rear windows or semicircular branches of glasses. Here, the glass exit temperature range is 640^0C to 660^0C. However, the subsequent need to be achieved to get smart quality tempered glass [1], [7]:

- ✓ The unchanging temperature within the glass because it exits the chamber and passes in the quench.
- ✓ The temperature would be on top of the Transition purpose (567^0C)
- ✓ The temperature would be below the Softening purpose (710^0C)
- ✓ Possess the glass optically flat while softening if it is required
- ✓ To quench while not losing an excessive amount of temperature
- ✓ Uniformly simmer down at a controlled rate to the Transition purpose (567^0C)
- ✓ Keep the cooling rate still well below the Strain purpose (510^0C) simmer down to a handling temperature

2.2. Resistance Heating and Melting Furnace

The resistance heating method is the simplest and the oldest electric-based procedure of heating, melting nonmetals and metals. Potency will reach near a hundred percent and temperatures will surpass $1{,}982^0C$. The resistance heating will be used for each high-temperature and low-temperature application. There are two measure basic kinds: direct and indirect resistance heating. In many cases, resistance heating is chosen because of its simplicity and efficiency [3], [9].

As a result of resistance, heating is employed for thus many various kinds of applications, there are measure a good type of fuel-based processes heating systems, moreover as steam-based systems that perform identical operations. In several cases, resistance heating is chosen for its simplicity and efficiency [3], [9].

2.2.1. Direct Resistance Heating Furnace

Direct resistance (also referred to as physical phenomenon heating), an electrical current that flows through a piece of fabric and heats it directly. Naturally, metal is held to electrodes within the walls of the chamber and charged with an electrical current. Electrical phenomenon inside the load generates heat that heats or melts the metal. The temperature is controlled by adjusting the present, which might be either alternating current or direct current. The fabric to be heated should conduct a minimum of some of the electrical current for direct resistance to work. Metals with a low physical phenomenon, like steel, produce additional resistance and additional heat, which makes the method additional economical. Direct resistance heating is commonly used to raise the temperature of steel items before formation, rolling, or drawing an application [9].

2.2.2. Indirect Resistance Heating Furnace

Indirect resistance heating, constituent transfer heat to the fabric by radiation, convection, and conductivity. The component is created of a high-resistance material like graphite, Silicon carbide, or nickel chrome. Heating is typically tired of a furnace, with a lining and interior that varies counting on the board material. Typical furnace linings square measure ceramic, brick, and fiber batting, while furnace centers are air, inert gas, or a vacuum [3].

Indirect resistance heating can also be done with an encased heater, in which the resistive element is encased in an insulator. The heater is placed in liquid that needs to be heated or close to a solid that requires heating. Different kinds of resistance heating instrumentation square measure used throughout the trade, as well as strip heaters, cartridge heaters, and tubular heaters [7], [9].

2.3 Related Works

The author in reference [10] has developed "Understanding PID Temperature Control as Applied to Vacuum Furnace Performance". This paper presents the PID controller to the temperature control of the vacuum furnace at a desired value of temperature for process, such as 1500°F. The problem with this controller is that the system is not robust because which do not have an auto-tuning feature and it is set up manually.

The author in reference [7] has developed “Temperature Control of Electric Furnace for Glass Tempering Process”. This paper is the temperature control of electric furnaces for glass tempering using Feed Forward Model Adaptive Control (FFMRAC). The author is tried to solve the presented problem by using a feed-forward model reference adaptive control with MATLAB/SIMULINK but it has no enough settling time because the temperature is high. Thus, it requires some minutes to reach the desired value, and the response of the system has high chattering. This shows there is a maximum temperature difference in the process of tempering glass. Thus, it has an impact on the quality of the product.

The author in reference [11] has developed “Feedback Linearization based control of a horizontal radioactive furnace”. This paper describes the application of the feedback linearization technique for the control of a horizontal radioactive furnace which is generally used for producing tempered glass. This is not focused on the temperature control for glass tempering only on the production of tempered glass.

The author in reference [12] has developed “implementations of fuzzy logic controllers in temperature control of tempered glass furnace”. This work presents a procedure to preserve the temperature of a heater utilizing Fuzzy Logic. The heater or furnace temperature must be controlled and kept up to 1500 degrees Celsius and the reason for the heater is to consider the characteristics of welding materials at exceptionally high temperatures. Since the extend of temperature above ambient is reasonably high and the heater working zone is additionally huge, the wonder of keeping up the temperature at the specified value poses numerous challenges. In this work, the Fuzzy Logic Control (FLC) system has been actualized by appropriately considering the working temperature run, as well as the working area. The initiation of the control system is its capacity to realize the set temperature at the set time and also to preserve it. The restriction of this work is it presents a steady-state error.

The author in reference [3] has developed “Fuzzy-PID Based Temperature Control of Electric Furnace for Glass Tempering Process”. This paper could be a model of a glass tempering furnace that is arranged utilizing MATLAB/SIMULINK and after that, a fuzzy-PID controller has been outlined and simulated and its overview has been related to the conventional PID controllers. The parameters performance of the controller is compared in terms of disturbance rejection, transient response, and steady-state performance. But For a fuzzy interface system, the rule base and scale factor and PID parameter gains should be optimized with an optimizing algorithm.

The author in reference [13] has developed "Temperature Control System Using Fuzzy Logic Technique" This paper presents a fuzzy logic based-temperature control system, which comprises a microcontroller, temperature sensor, and operational speaker, Simple to Computerized Converter, show interface circuit, an output interface circuit. It contains a planned approach that employments the fuzzy logic method to attain a controlled temperature output function.

The author in reference [14] has developed "The Temperature Control of Electric Furnace Based on PID Genetic Algorithm". This paper was outlined by the rule of the conventional PI and PID tuned with Genetic Algorithm control system and dynamically simulated a mechanical furnace control system. But it is demonstrated that this control algorithm is successful in progressing the temperature control of the resistance-heated furnace but there is a high-temperature difference and most extreme time domain specifications.

The author in reference [15] has developed "The Electric Heating Furnace Temperature Control Based on SPID". A temperature system supported the electrical heating furnace, a form of single parameter PID control method is studied. The simulation shows that the one-parameter PID control, and could be a proper, and humble control technique used for the system but it is an outsized time delay. The simulation result's a similar temperature system in the electrical heating furnace in two ways, the Smith predictive control, and also the single parameter PID control. The one parameter PID control method is better than that of the Smith predictive method but both have an oversized overshot and huge delay.

From the above review, it is clear that different kinds of literature are reviewed about the glass tempering process, glass properties, electric heating furnaces, and temperature controllers that have been used in this thesis. Besides, different related works are reviewed related to glass tempering. As far as known, the Genetic Algorithm-based Fractional Order PID controller to control the temperature of the furnace for tempering process the glass is not modeled and designed. Therefore, in this thesis, a fault-tolerant FOPID with GA tuned temperature control for the glass tempering process will be modeled and simulated. To improve and optimize the measure of performance and rise time, overshoot, and to minimize the error between the output of the plant output with a better response and also, to compare the results of manual PID tuning, MRAC, GA-based PID, and GA-based Fractional Order controllers.

CHAPTER THREE

METHODOLOGIES AND MATERIALS

3.1. Materials

In this research work, software such as MATLAB/R2019b, Microsoft office 2019, and MathType 6.0 equation is used. MATLAB is the computing software that consists of technical toolboxes and Simulink. Microsoft office is used for writing the thesis documentation of the thesis. MathType 6.0 is used for writing mathematical formulas and equations in Microsoft offices word 2019 and PowerPoint presentation tools.

3.2. Methods

For the accomplishment to control the temperature of the furnace for tempering process of glass, the following methodologies have been followed:

1. The information is gathered and surveyed from related works and various techniques which are literature, reading books, journals, publications, and documents on related areas.
2. Design and Model of Controller for the system.
3. Performance evaluation and discussed the result.
4. Analyze the design with MATLAB/Simulink simulation. In general, the method followed throughout the research can be summarized in the following flow chart of Figure 3.1.

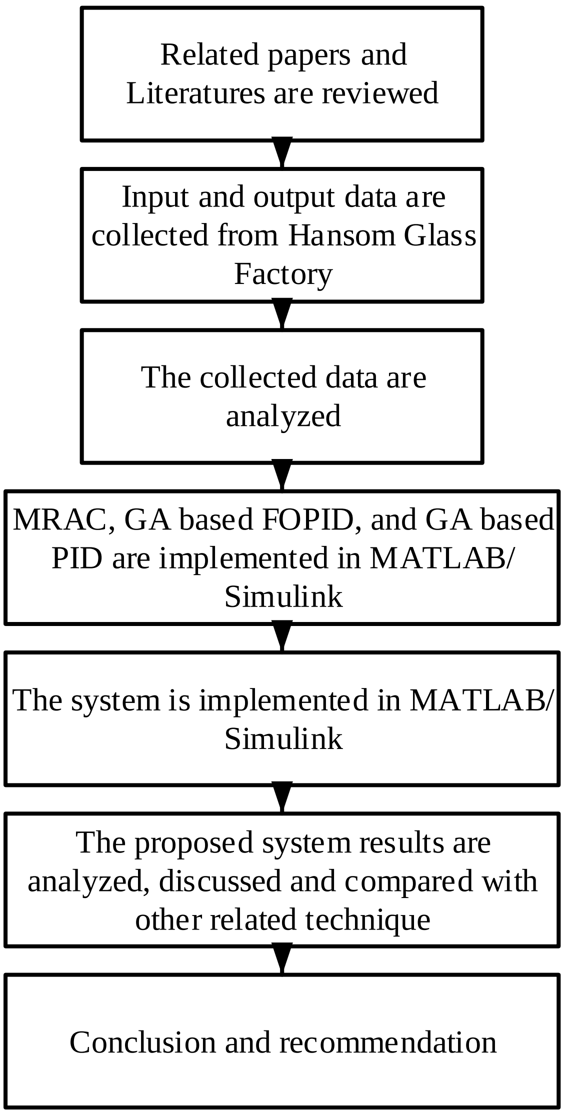


Figure 3.1: Block diagram of the research methodology

3.3. Model of the Glass Tempering Process

The main step within the design and analysis of the control system is the mathematical modeling of the different components of the system. The transfer function is broadly utilized for scheming control systems. Next appropriate assumptions and approximations are completed, the mathematical equations depicting the mechanisms, transfer functions are derived in Laplace transform. Therefore, the tempering process is modeled, designed, and simulated for temperature control of the electric furnace using the obtained transfer functions. Figure 3.2 shown below is the block diagram that describes the main components of the glass tempering process of the heating section. Before designing the temperature control system, the appropriate model for each component should be obtained [7].

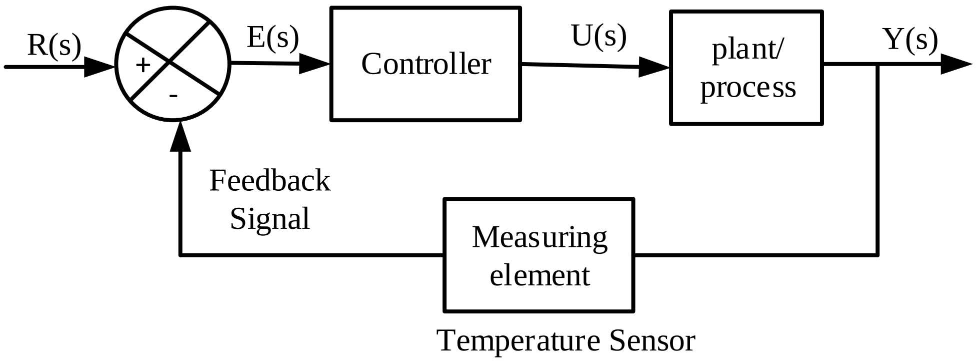


Figure 3.2 General block diagram of temperature control of furnace for glass tempering

3.4. Temperature Control Systems of Resistance Furnace

To produce the desired quality of tempered glass the temperature of the electrical furnace should be controlled. There are completely different controllers employed in the glass tempering industry.

3.4.1. PID Controller

The primary closed-loop controller is the PID Controller. It was a fundamental component of first councils and it got to be the standard instrument when process control has risen within the 1940s [16], [17], [18]. In-process control nowadays, more than 95% of the control loops are PID type controllers, most loops are PI control. PID controllers are nowadays found in all regions where control is utilized. The controllers come in numerous diverse shapes. There are standalone systems in boxes for one or several loops, which are made by the hundred thousand annually. PID controllers have survived many changes in technology, from mechanics and pneumatics to microprocessors via electronic tubes, transistors, integrated circuits.

The microprocessor has had a dramatic influence on the PID controller. Practically all PID controllers made today are based on microprocessors. This has given opportunities to provide additional features like automatic tuning, gain scheduling, and continuous adaptation [3], [19].

3.4.2. Algorithm of PID Controller

A PID controller persistently calculates error esteem as the contrast between the required setpoint and a measured process variable. The controller efforts to reduce the error over time by alteration of a control variable, like the location of a control valve, a damper, or the control provided to a heating component, to a new value determined by a weighted sum [20], [21].

$$G(t) = K_p e(t) + K_i \int_0^t e\,(t)dt + K_d \frac{de(t)}{dt} \qquad (3.1)$$

where, Kp, Ki, and Kd are non-negative, designate the coefficients or gains of the proportional, integral, and derivative terms, separately.

Proportional Term (P): this is accounts to show the error values. Then for a proportional controller non-zero error is required to drive it, for the most part, works which are called steady-state error. SSE is proportional to the method pick up and conversely proportional to the relative gain. The steady-state error may be relieved by including a compensating predisposition term to the setpoint or yield or redressed powerfully by including an integral gain or term [22].

The proportion gain is given by:

$$P_{out} = K_p \tag{3.2}$$

The Integral Term (I): accounts for former values of the mistake or error. The commitment from the integral term is proportional to both the estimate of the mistake and the length of the error or mistake. The integral in a PID controller is complete of the quick blunder over time and gives the collected adjusted that would have been balanced already. The collected blunder is at that point multiplied by the integral gain and included within the controller output [19].

The integral term is given by:

$$I_{out} = K_i \int_0^t e(t)dt \tag{3.3}$$

The integral term animates the development of the method towards the setpoint and disposes of the remaining steady-state blunder that happens with an unadulterated proportional controller. Since the integral term responds to accumulated mistakes from the past, it can cause the appear regard to overshoot the setpoint esteem [19].

Derivative Term: accounts for conceivable future values of the mistake, based on its current rate of change. The derivative of the method mistake is calculated by choosing the slant of the error over time and copying this rate of modifying by the derivative pick up or gain. The measure of the commitment of the subordinate term to the common control action is called the derivative gain, Kd [22].

The derivative term is given by:

$$D_{out} = K_d \frac{d(e)}{dt} \tag{3.4}$$

Derivative activity predicts system behavior, and in this way, progresses settling time, and soundness of the system. A perfect subordinate is not causal, so the usage of PID controllers incorporate extra low pass filtering for the derivative term, to constrain the maximum temperature gain and noise [15].

A few applications may require utilizing as it were one or two terms to supply the appropriate system control. This can be accomplished by setting the other parameters to zero. A PID controller will be called a PI, PD, P, or I controller within the nonappearance of the individual control activities. PI controllers are reasonably common, since the derivative activity is sensitive to estimation noise, while the nonappearance of a fundamental term may anticipate the system from coming to its target esteem [18], [22]. The positioning algorithm that works with the PID controller is shown in the following Figure below.

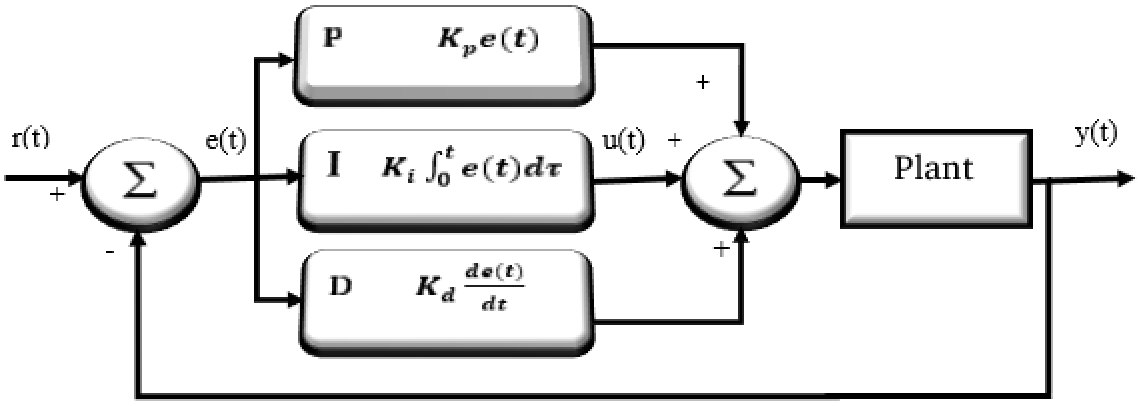


Figure 3.3 The PID controller block diagram [19]

where $u(t)$ and $e(t)$ show the controlling signal and the error signals of the control system respectively.

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \qquad (3.5)$$

Where Kp is proportion, Ki is integral, Kd is derivative of the values of the gains, and G(s) is the transfer function.

3.4.3. The PID Controller Limitation

While PID controllers apply to many control problems and often perform satisfactorily without any improvements or only coarse tuning, they can perform poorly in some applications and do not, in general, provide optimal control. The basic difficulty with the PID controller is that it may be a feedback control system, with dependable parameters, and no coordinate information of the process, and thus overall performance is reactive and a compromise. PID controllers, when utilized alone, can provide poor performance when the PID loop gains must be decreased. The most significant improvement is to incorporate feed-forward control with knowledge about the system and using the PID only to control error. On the other hand, PIDs can be adjusted in more minor ways, such as by varying the limitations (either gain planning different utilize cases or adaptively altering them based on performance), improving estimation (accuracy, precision, and low-pass filtering on the of chance that necessary), or cascading numerous PID controllers [19], [22].

3.5. Genetic Algorithm

A genetic algorithm is a robust optimization method based on natural selection that is used to optimize a function which is known as a fitness function [23]. The genetic algorithm starts with an initial population which is containing many chromosomes wherever each chromosome represents a solution to the problem and its performance is evaluated by a fitness function. The genetic algorithm consists of three main stages: Selection (reproduction), Crossover, and Mutation. Selection retains a fit set of chromosomes in the given population. Crossover and mutation involve swapping between two-parent strings and flipping a random bit in the set of chromosomes respectively [23].

The application of these three essential operations permits the creation of new people who may be better than their guardians. This algorithm is repeated for many generations and finally stops when reaching people that represent the optimum solution to the issue [24], [25].

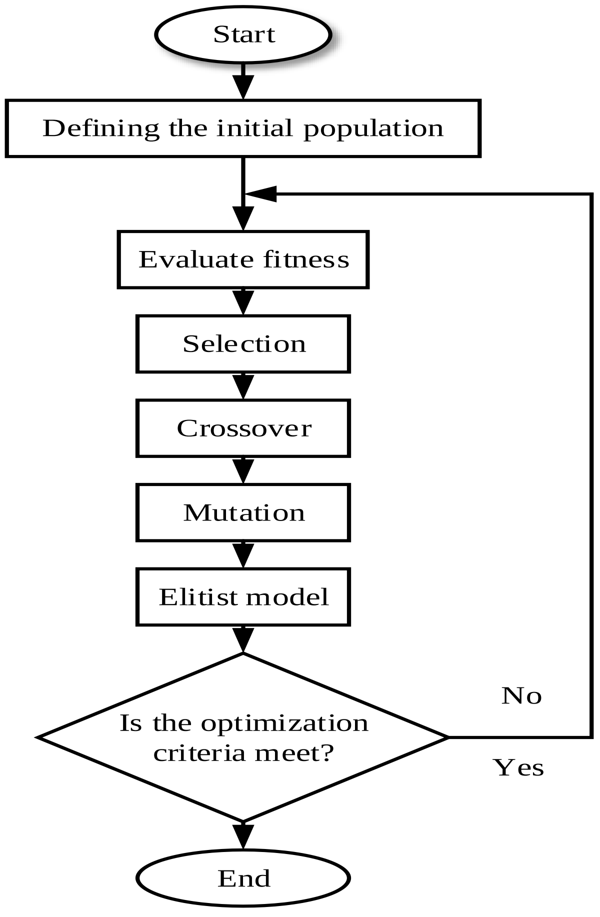


Figure 3.4 The flow chart for implementation of Genetic Algorithm

3.6. Fractional Order Calculus

Applications of fractional calculus can be obtained in the area of control systems. Fragmentary calculus permits the derivative and integral to be of any real number. The fractional-order discriminator can be indicated by a common essential operator aD^qt as a generalization of the differential and integral operators, which is characterized as follows [26], [27]:

$$aD^qt = \begin{cases} \frac{d\alpha}{dt^\alpha} & R(\alpha) > 0 \\ 1 & R(\alpha) = 0 \\ \int_0^t dt^{-\alpha} & R(\alpha) < 0 \end{cases} \quad (3.6)$$

where α is a complex number of the fractional-order, the constant 'a' is related to the initial conditions. There are two usually utilized definitions for the common fractional differentiation and integration, that is, the Grunwald Letnikov (GL) and the Rieman Liouville definitions (Oldham 1974). Perhaps the most excellent known due to its most suitability for the realization of discrete control calculations. The Grunwald-Letnikov definition is expressed as:

$$aD^qtf(t) = \lim_{h\to 0} h^{-q} \sum_{j=0}^{(t-\alpha)/h} (-1)^j (q/j) f(t-jh) \quad (3.7)$$

where $\frac{t-\alpha}{h}$ is an integer

The Riemann-Liouville definition is expressed as:

$$f(t) = \frac{d^n}{dt^n} \int_\alpha^t (f(\tau)/(t-\tau)^{q-n+1})\, d\tau \quad (3.8)$$

For a wide class of functions that show up in real physical and building applications, the Riemann-Liouville and the Grunwald-Letnikov definitions are equivalent [27].

3.6.1 Fractional Order PID Controller

A general form of the PID controller which is proposed by Podlubny is known as the fractional-order $PI^{-\lambda}D^{\mu}$ controller, where the values of λ and δ lie between 0 and 1. The fractional-order has more than two adjustable parameters when compared with the PID controller, which makes the parameter tuning more flexible, it is very important for improving the control accuracy. Therefore, the fractional-order $PI^{-\lambda}D^{\mu}$ the controller is applied in the active electric furnace temperature control system [26].

The differential equation of fractional order $PI^{-\lambda}D^{\mu}$ the controller is shown as equation (3.9).

$$u(t) = K_p e(t) + K_i D^{-\lambda} e(t) + K_d D^{\mu} e(t) \quad (3.9)$$

By converting into the Laplace transform the controller transfer function is given as:

$$G_f(s) = \frac{U(s)}{E(s)} = K_p + K_i s^{-\lambda} + K_d s^{\mu} \quad (3.10)$$

Taking $\lambda = 0$ and it is the conventional controller of PID, if $\mu = 0$ and λ =1 it is the conventional controller of PI, if $\lambda = 0$ and $\mu = 1$ it is the conventional controller of PD, and if both $\mu = 0$ and $\lambda = 0$ it is the conventional controller of P. The four kinds of the controllers are fractional order $PI^{-\lambda}D^{\mu}$controller, it can be seen that the adjustable range of the fractional $PI^{-\lambda}D^{\mu}$ the controller is additional extensive than the conventional PID controller. Therefore, the performance of the parameter of the fractional-order controller is more superior to the conventional PID controller.

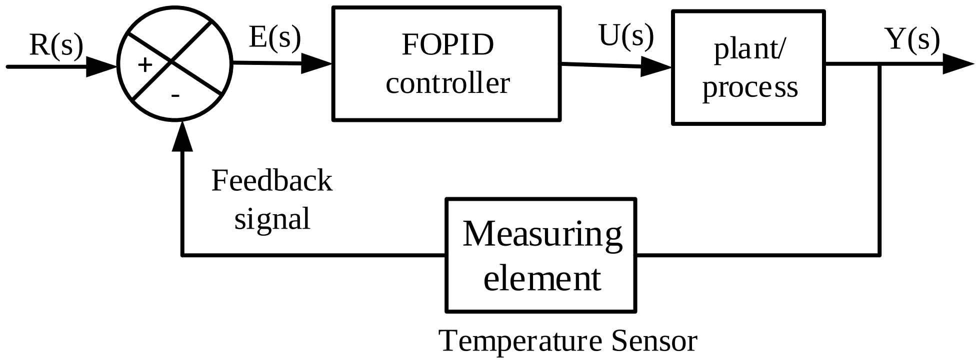


Figure 3.5 schematic diagram of FOPID controller

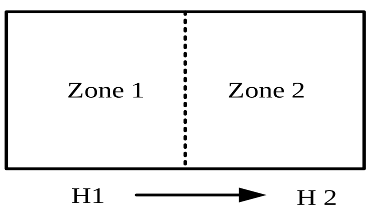
3.7. Model of Electric Furnace

In the glass tempering process, the electric furnace is used to heat the glass to the desired value. Thus, an electric furnace converts electric power or energy to heat energy and heats the glass through conduction, radiation, and convection. The electric furnace of temperature control is given by the first order plus time delay equation [7].

$$G(s) = \frac{ke^{-Ds}}{Ts+1} \tag{3.11}$$

where T is time constant, k is conduction coefficient value and D is a pure lag time.

Assuming the wall of the glass tempering furnace has ideal heat, insulation and there are no losses in the surrounding. The heat equilibrium can be written as follows [3], [12].



$$\rho CV \frac{d\theta}{dt} = H_1 - H_2 \tag{3.12}$$

where,

ρ-is the density of the material (ton/m^3)

C-is the particular heat of the material ($Mcal/\text{deg. ton}$)

V-is the volume of the furnace (m^3)

θ- is the temperature zone (^{0}C)

$H_1 = Q_1\rho_1 c\theta_1$ is the amount of heat flow with material ($Mcal/\text{hour}$)

Q_1 is the flow rate input (m^3/hour)

$H_2 = Q_2\rho_2 c\theta_2$ is the amount of heat flow with material ($Mcal/\text{hour}$)

Q_2 is the flow rate of the output (m^3/hour)

The following are taken as an assumption:

1. $\theta = \theta_1 = \theta_2$

2. $Q = Q_1 = Q_2$

Substitute this assumption into equation (3.12) and convert it into the Laplace transform of it then it gives:

$$\frac{\theta(s)}{\theta_1(s)} = \frac{1}{\frac{V}{Q}s+1} \cong \frac{1}{Ts+1} \tag{3.13}$$

where, $T = \frac{V}{Q}$ is the time constant, and considering input as H_1, then system transfer function becomes:

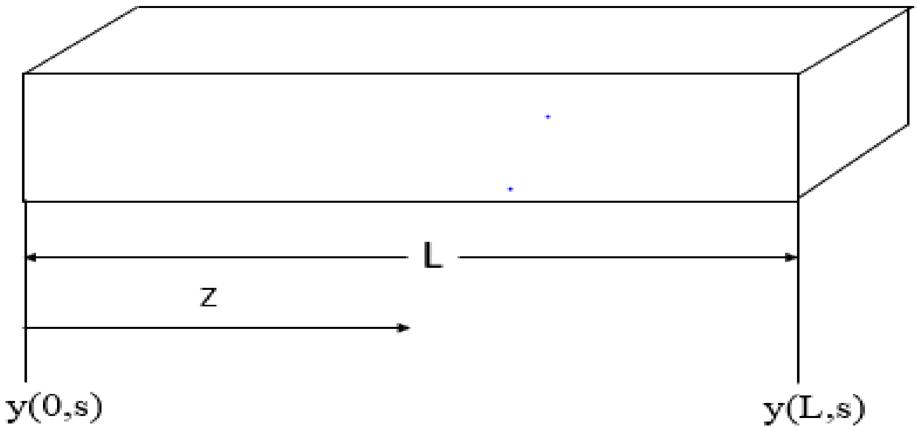
$$\frac{\theta(s)}{H_1(s)} = \frac{K}{Ts+1} \tag{3.14}$$

where K is the conduction coefficient and T is the time constant.

$$\left.\begin{aligned} K &= \frac{1}{\rho Q C_p} \\ T &= \frac{V}{Q} \end{aligned}\right\} \tag{3.15}$$

3.8. Model of the Delay Time

The process involves the flow of normal glass at a continuing rate v, over a superbly insulated furnace length L. The glass is primarily at a constant temperature T_0, uniformly, through the entire furnace length. At time $t = 0$, the temperature of the glass coming in at the furnace inlet($z = 0$) is changed to $T_i(t)$. It is now interested in investigating how the temperature at the furnace outlet ($z = L$) responds to the input temperature changes [28]



It is simple to visualize that any temperature changes enforced at the furnace inlet will not be registered instantly at the outlet. This is because it will take a finite amount of time for an individual glass element at the new temperature, to traverse the distance L from the inlet to the outlet. So that its new temperature can be observed. Since the glass rate is v, and it is supposed constant, the time for every glass part to traverse the specified distance from the furnace recess to the outlet is $\frac{L}{V}$. [14] [28].

3.9. Mathematical Model and Analysis

Consider an element of the glass of thickness ∇Z whose boundaries are arbitrarily located at the points Z and $Z + \nabla Z$ along the length of the furnace. For a furnace of constant, uniform cross-sectional area A, an energy balance over such an element gives [28].

$$\rho A C_p \nabla Z \frac{\partial T}{\partial t} = \rho A v C_p \frac{(T - T^*)}{Z} - \rho A v C_p \frac{(T - T^*)}{Z + \nabla Z} \quad (3.16)$$

where ρ, C_p and T^* are the glass density, specific heat capacity, and the usual reference temperature. By dividing both sides by ∇Z , and taking the limits as $\nabla Z \to 0$ Eq. (3.16) becomes:

$$\frac{\partial T(z,t)}{\partial t} + V \frac{\partial T(z,t)}{\partial z} = 0 \quad (3.17)$$

Such like system whose dynamic behavior is represented by PDEs is called a distributed parameter system. Therefore, the procedure presently below consideration is one of such systems. Let us now define the deviation:

$$T(z,t) = T(z,t) - T(0) \quad (3.18)$$

where it recalls that $T(0) = T_0$ - is that the initial glass temperature at $t = 0$ along the furnace length. In terms of abnormality variable, the method model in Eq. (3.17) becomes:

$$\frac{\partial y(z,t)}{\partial t} = v \frac{\partial y(z,t)}{\partial z} = 0 \quad (3.19)$$

To develop the transfer function of the model for this process must be converted into Laplace transform of the expression in Eq. (3.19).

$$sy(z,s) + v\int_0^{\infty} e^{-st}\frac{\partial y(z,t)}{\partial z}dt = 0 \tag{3.20}$$

Since the first term is simply converted to the Laplace transform (concerning time) of the indicated time derivative. For the second term in Eq. (3.20), interchanging the order of differentiation and integration, and it gives:

$$\int_0^{\infty} e^{-st}\frac{\partial y(z,t)}{\partial z}dt = \frac{d}{dt}\int_0^{\infty} e^{-st}\, y(z,t)dt = \frac{d}{dz}y(z,s) \tag{3.21}$$

Thus, the equation (3.17) becomes:

$$sy(z,s) + v\frac{d}{dt}y(z,s) = 0 \tag{3.22}$$

$$y(z,s) = C_1 e^{\left(\frac{-z}{v}\right)s} \tag{3.23}$$

We may evaluate the arbitrary constant C_1 by using the entrance condition at z $=$ 0, which is expressed as $y(z,s) = y(0,s)$. Thus Eq. (3.24) becomes:

$$y(z,s) = y(0,s)e^{\left(\frac{-z}{v}\right)s} \tag{3.24}$$

Eq. (3.24) addresses the connection between the Laplace change of the inlet temperature and the Laplace change of the temperature at some other point z along the length of the heater. Notice further that the term $^{z}/_{v}$ is the proportion of the distance that went in showing up now, to the stream speed, an immediate proportion of the time taken to cross the demonstrated distance. It is particularly concerned with the situation at the furnace exit; that is, at $z = L$. In this case, Eq. (3.24) becomes:

$$y(L,s) = y(0,s)e^{\left(\frac{-L}{v}\right)s} \tag{3.25}$$

And if it is defined as D the time taken for a glass element to traverse the entire length of the furnace, that is:

$$D = \frac{L}{v} \tag{3.26}$$

Now substitute eq. (3.26) into Eq. (3.25):

$$y(L,s) = y(0,s)e^{-Ds} \tag{3.27}$$

process output (changes in the exit temperature, y(L,s)) of Eq. (3.27) is given by:

$$g(s) = e^{-Ds} \quad (3.28)$$

Now it can be used Eq. (3.28) to investigate how this process will respond to deviations in the input function $u(t) = y(0, t)$. This task is seen to be very easy once, we recall the effect of the translation function of the Laplace transforms. Thus, for any input $u(t)$ that it can express as:

$$Y(L, s) = U(s)e^{-Ds} \quad (3.29)$$

When this eq. (3.29) is converted to the time domain, it gives the following:

$$Y(L, t) = u(t - D) \quad (3.30)$$

It indicates that the output is the same as the input, only delayed by D time units.

Heat Flow Rate:

The heat flow rate Q can be calculated as [29]

$$Q = \frac{H}{C_{pa}\rho_a(T_h - T_L)} \quad (3.31)$$

where:

H - is the amount of heat flow (input power) in material (Mcal/hour)

Q- is the flow rate of heat through an array of nozzles to the glass surface by compressed air (radiation and convention)

ρ_a - is the density of air

C_{pa} - is the specific heat of the air (Mcal/deg. ton)

T_h - is the heating temperature

T_L - is lower temp/room temperature

The required amount of heat flow H is often calculated as

1. By conduction [30] Rate of heat conduction $\propto \frac{\text{Area}*\text{Temperature}}{\text{Thickness}}$

$$H_{cond} = kA\frac{\nabla T}{\nabla x} \quad (3.32)$$

where k is the thermal conductivity of the glass

∇X is the thickness of the glass and the furnace that has physical contact with the heater.

2. By convention and radiation

By radiation heat transfer to or from a surface enclosed by a gas-like air happens parallel to conductivity (or convection, if there is bulk gas motion) between the surface and therefore the gas. Thus, the entire heat transfer is decided by adding the contributions of both heat transfer mechanisms. For simplicity and convenience, this is often done by defining a combined heat transfer coefficient $h_{combined}$ that has the consequences of both convention and radiation [30].

$$H_{conv\ and\ radi} = H_{combined} A_s (T_s - T_\infty) \quad (3.33)$$

where, $h_{combined}\ is$ the convection and radiation heat transfer coefficient in $\frac{w}{m^2, c^0}$.

T_s - is the temperature of the surface, and $(T_\infty = 30^0 c)$ is the air temperature.

A_s - is the surface area of the glass to be heated.

Therefore, the summation H_{conv} and $H_{conv\,\&\,radi}$ will give the total amount of heat transfer.

$$i.e \quad H_{total} = H_{conv} + H_{conv\,\&\,radi} \quad (3.34)$$

Taking Assumptions of the following

The volume of the furnace space is chosen to be $V = 4m^3$

The velocity of the glass is chosen to be $v = \frac{0.2m}{s}$

Length of furnace = $2m$

$\rho = \frac{2.5gram}{cm^3}$

$C_p = 0.18 \frac{cal}{gram^0 C}$

Electric Power = $120kw = \frac{120kj}{s}$

Now, substituting in Eq. (3.31)

$$Q = \frac{0.16m^3}{s}$$

Substituting the values of $Q, V, \rho\ and\ C_p$ in eq. (3.15) and it yields:

$K = 13.89\ and\ T = 25sec$ And the delay time $D = \frac{L}{v} = 10sec$

Therefore, the glass tempering resistance furnace dynamic model is:

$$G(s) = \frac{\theta(s)}{H(s)} = \frac{Ke^{-Ds}}{Ts+1} = \frac{13.89e^{-10s}}{25+1} \quad (3.35)$$

3.10. Model of the Temperature Sensor

The temperature can be measured by different apparatus or instruments which can sense or detect the temperature with an electrical signal, including thermocouples, thermistors, resistance thermometry devices (RTDs), etc. This section addresses each of the static (calibration) and dynamic (time response) and the characteristics of temperature sensors [31]. Calibration of the sensor is determining the relationship between the real quantity of interest (the temperature at a few areas within the glass) and the output given by the sensor (which can be a voltage, a current in a circuit, an advanced representation, etc., depending on the instrument). When we talk about a sensor, we usually refer to both the sensing element (such as the bimetallic junction of a thermocouple) and signal conditioning electronics. It is this latter part that produces a linear relationship between temperature and sensing element output, although the behavior of the detector itself could also be nonlinear [32].

Thus, it relates the physical temperature T_0 and its sensor reading T_s given by [32]:

$$T_s = K_s T_0 + b_s \tag{3.36}$$

In a hand-held digital measuring device, the electronics are adjusted so that gains K_s is unity and bias b_s is zero: 26^0C produces a reading of 25^0C. In a control loop, however, it is more likely to has produces an electric current that ranges over 4 to 20 mA, where these limits correspond to the expected range of temperature variation. Current loops are good over the types of distances that separate operational processes from their control rooms. [12]. The sensor range is adjusted by varying K_s and b_s. Suppose that we wish to follow over the range 622^0C to 638^0C. Then,

$$4mA = 622^0C * K_s + b_s \tag{3.37}$$

$$20mA = 638^0C * K_s + b_s \tag{3.38}$$

Therefore,

$K_s = 1mAK^{-1}$ and $b_s = -618mA$

This method is to use 620^0C as a reference operating condition. At the reference, the sensing element (sensor) output is going to be $12mA$ [32]. Sensor response is first-order. Hence, it tends to could write the sensor transfer function as follows:

$$T_s(s) = \frac{K_s}{\tau_s s+1} T_0(s) \tag{3.39}$$

where:

$T_s(s)$ - is sensor reading

K_s - is that the detector gain that resolves by activity in (3.37) and (3.38) and most of the time is unity.

$T_0(s)$ - is that the physical temperature the system

τ_s - could be a time constant that depends on the mass of the sensor and the rate of heat transfer to the sensor.

When the detector is immersed within the medium, the heat is transferred to the detector within the measure.

$$dQ = \alpha A(Q - Q_T)dt \tag{3.40}$$

The heat stored in the sensor:

$$dQ = mcdQ_T \tag{3.41}$$

Therefore,

$$\alpha A(Q - Q_T)dt = mcdQ_T \tag{3.42}$$

$$\frac{mc}{\alpha A}\frac{dQ_T}{dt} = 0 \tag{3.43}$$

Hence $\tau_s = \frac{mc}{\alpha A}$

where:

α - is the coefficient of heat transfer ;

A - is that the area of the detector or the sensor

C - is that the heating capability

m - is that the mass of the sensing element

3.11. Design of PID Controller

The tuning PID control loop effect is that the adjustment of its control parameters (proportional gain, integral gain, derivative gain) to the optimal values for the desired response. Stability may be a basic demand, however on the far side that, totally various systems have various behavior. Generally, various applications have various needs, and needs might conflict one with another. The transfer function Gc(s) of PID controller is [17], [33], [34]:

$$G_c(s) = K_p\left(1 + \frac{1}{\tau_i s} + \tau_d s\right) \tag{3.44}$$

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s) \tag{3.45}$$

3.11.1. Loop Tuning Mechanism

Tuning a control loop is the alteration of its control parameters to the optimum values for the desired control response. Stability (no unbounded oscillation) may be a necessity, but diverse systems have diverse behavior, diverse applications have diverse requirements, and requirements may conflict with one another [16], [35]. PID tuning may be a troublesome issue, indeed even though there are three parameters and in the guideline is basic to depict. Since it must fulfill complex criteria inside the limitations of PID control. There are in like manner different strategies for loop tuning [3].

3.11.2. Manual Tuning

For manual tuning in the online system, one of the methods is first setting all gains to zero. Increase the value K_p till the continuous oscillation of the output is attained and then K_p should be set to approximately half of that value for a quarter amplitude decay type response. After that increase K_i to minimize the P term offset in a particular time for the process considering that excessively much K_p can be a cause of instability. Lastly, increase K_d, if necessary, until the system becomes acceptably quick to reach the Setpoint and decrease the oscillation. However, too much K_d will lead to an overshoot in the system. A quick PID controller loop has a small overshoot to attain the desired value very fast, but in some systems cannot receive overshoot. In these systems a K_p should be set unusually less than half of the K_p setting which activated oscillation, to make over-damped in the closed-loop system [3], [22].

Table 3.1 Effects of independent P, I, and D tuning on closed-loop response

PID gains	Rise Time	Overshoot	Settling Time	Steady-State Error	Stability
Increasing K_p	Decrease	Increase	Small Increase	Decrease	Decrease
Increasing K_i	Small Decrease	Increase	Increase	Large Decrease	Decrease
Increasing K_d	Small Decrease	Decrease	Decrease	Minor Change	Improve

3.11.3. Conventional PID Tunings Method

Conventional PID is the most typical controller employed in the process industries in the last six decades, because of simplicity, ease of implementation, availability of many tuning methodologies, and minimum knowledge of the process enough to control. Thus, there is no assurance of dynamic responses, structural complexity, nonlinearities, and large time delays. The uncertainties in some realistically constrained cases in modern industries are makes increasing the attention for an improvement of PID controller [36], [37]. There are diverse tuning relationships utilized in conventional PID tuning methods,g among those a few of the methods are discussed below.

I. Ziegler-Nichols Method

The Ziegler-Nichols design approaches are the foremost prevalent systems utilized in process control to decide the parameters of a PID controller. Even though these methods were displayed within the 1940s, they are still broadly utilized [18], [33]. The open-loop step response is mostly based on the input signal step response method. Thus requiring the method to be stable, the unit step response of the method is categorized by two parameters L and T. These are determined by drawing a tangent line at the inflection point, where the slope of the step response has its most extreme value [18].

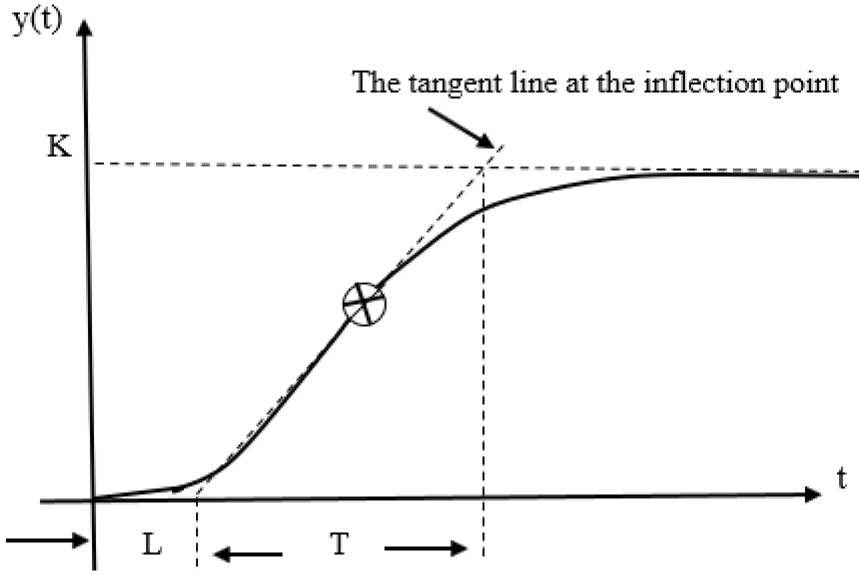


Figure 3.6 Step response for choosing L and T parameters utilizing the ZN method

The crossing points of the tangent and the coordinate axes deliver the method parameters as shown in Figure 3.6, and these are used in calculating the controller parameters. Since it handles the model is, to start by, First order plus dead time if the values of delay time D = 10, time constant T = 25, and gain K = 13.89 to calculate the values K_p, K_i and K_d through distinctive tuning relationships. The parameters for PID controllers gotten from the Ziegler-Nichols step response method are shown in Table 3.2.

Table 3.2 ZN PID controller parameter for first order plus dead time model

Controller	K_{Cz}	τ_{Iz}	τ_{Dz}
P	$(1/K)(T/D)$		
PI	$(0.9/K)(T/D)$	$3.3D$	
PID	$(1.2/K)(T/D)$	$2D$	$0.5D$

For PID controller

$K_{Pz} = 0.2160$

$K_{Iz} = 0.0108\ (\tau_{Iz} = 20)$

$K_{Dz} = 1.08\ (\tau_{Dz} = 5)$

Thus, the controller transfer function is:

$$G_c(s) = 0.216 + \frac{0.0108}{s} + 1.08s) \quad (3.46)$$

II. Cohen Coon Method

The Cohen-Coon technique is a more difficult type of the Ziegler-Nichols method [9]. This method is more sensitive than the Ziegler-Nichols technique. The controller parameters with this technique are given in table3.3 [3].

Table 3.3 Cohen Coon PID controller parameter for first order plus dead time model

Controller	K_c	τ_i	τ_d
P	$(1/K)(T/D)(1 + D/3D)$		
PI	$(1/K)(T/D)(0.9 + D/12D)$	$\frac{D[30 + 3(D/T)]}{9 + 20(D/T)}$	
PID	$(1/K)(T/D)\left(\frac{3D + 16T}{12T}\right)$	$\frac{D[32 + 6\,(D/T)]}{13 + 8(D/T)}$	$\frac{4D}{11 + 2(D/T)}$

For PID controller

$K_{Pc} = 0.258$

$K_{Ic} = 0.0122\ (\tau_{Ic} = 21.24)$

$K_{Dc} = 0.8746\ (\tau_{Dc} = 3.39)$

Thus, the controller transfer function is:

$$G(s) = 0.258 + \frac{0.0122}{s} + 0.8746s \quad (3.47)$$

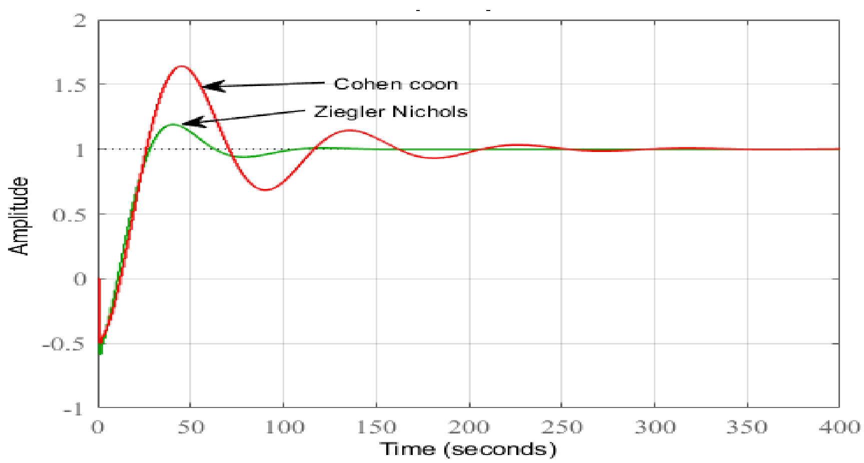


Figure 3.7 System response using PID controller of ZN and Cohen-Coon tuning

The Cohen Coon response shows the sudden change of the temperature of the furnace but the Ziegler-Nichols response shows the better temperature of the system. The Ziegler-Nichols Tuning method has a smaller overshoot than the Cohen coon. This indicates the Ziegler-Nichols is better than the Cohen coon in temperature difference but both are not good for this system because of high-temperature difference and large settling time [3].

III. ITAE Tuning Method

Another useful method for tuning PID is the ITAE method is described in [21]. Minimalizing the integral of time-weighted absolute error (ITAE) is commonly referred to as a good performance index in designing PID controllers. Integral time multiplied by absolute error (ITAE) criterion is given by:

$$ITAE = \int_0^{\infty} t|e(t)|dt \qquad (3.48)$$

where t is the time and e(t) is the error which is calculating the difference between the setpoint and the output. The controller parameters with this method are given in table 3.4. For a first-order plus dead time model [21], solve for:

$$\frac{\partial ITAE}{\partial k_c} = 0, \frac{\partial ITAE}{\partial \tau_i} = 0 \; and \; \frac{\partial ITAE}{\partial \tau_d} = 0 \qquad (3.49)$$

Table 3.4 ITAE PID controller parameter for first order plus dead time model

Type of input	Type of controller	Mode	A	B
Load	PI	P	0.859	-0.977
		I	0.674	-0.680
Load	PID	P	1.357	-0.947
		I	0.842	-0.738
		D	0.381	0.995
Set-point	PI	P	0.586	-0.916
		I	1.03	-0.165
Set-point	PID	P	0.965	-0.85
		I	0.796	-0.1465
		D	0.308	0.929

From the table,

Load Settings:

$$A(D/T)^B = KK_c = T/\tau_i = \tau_d/T \qquad (3.50)$$

Setpoint Settings:

$$A(D/T)^B = KK_c = \tau_d/T\,,\; T/\tau_i = A + B(D/T) \qquad (3.51)$$

i. Design Load Point Change

For PID controller

$A(D/T)^B = KK_{Pl} = 1.357 * (0.4)^{-0.947}$

$K_{Pl}13.89 = 3.232$

$K_{Pl} = 0.2327$

$A(D/T)^B = T/\tau_{Il} = 0.842 * (0.4)^{-0.738} = 1.6557, \tau_{Il} = T/1.6557 = 15.099$

$K_{Il} = 0.0154$

$A(D/T)^B = \tau_{Dl}/T = 0.381 * (0.4)^{-0.995} = 0.948, \tau_{Dl} = T * 0.948 = 23.704$

$K_{Dl} = 5.516$

Thus, the controller transfer function is

$$G(s) = 0.2327 + \frac{0.0154}{s} + 5.516s \qquad (3.52)$$

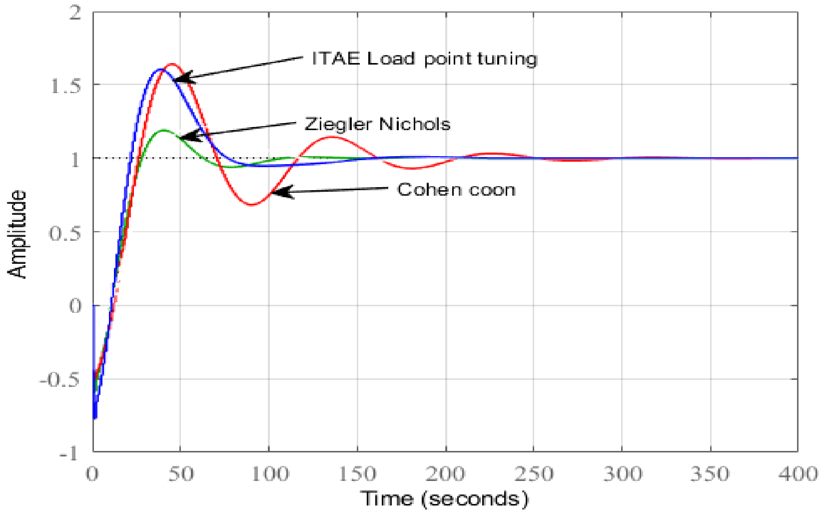


Figure 3.8 System step response using ZN, Cohen Coon, and ITAE load setting tuning

ii. Design Setpoint Change

For PID controller

$K_{Ps}K = 0.965 * (0.4)^{-0.85}$

$K_{Ps} * 13.89 = 2.103$

$K_{Ps} = 0.1514$

$T/\tau_{Is} = 0.796 * (0.4)^{-0.1465} = 0.9104, \tau_{Is} = T/0.9104 = 27.46$

$K_{Is} = 0.0055$

$\tau_{Ds}/T = 0.308 * (0.4)^{-0.929} = 0.7215, \tau_{Ds} = T * 0.3790 = 18.038$

$$K_d = 2.731$$

Thus, the controller transfer function is

$$G(s) = 0.1514 + \frac{0.0055}{s} + 2.731s \quad (3.53)$$

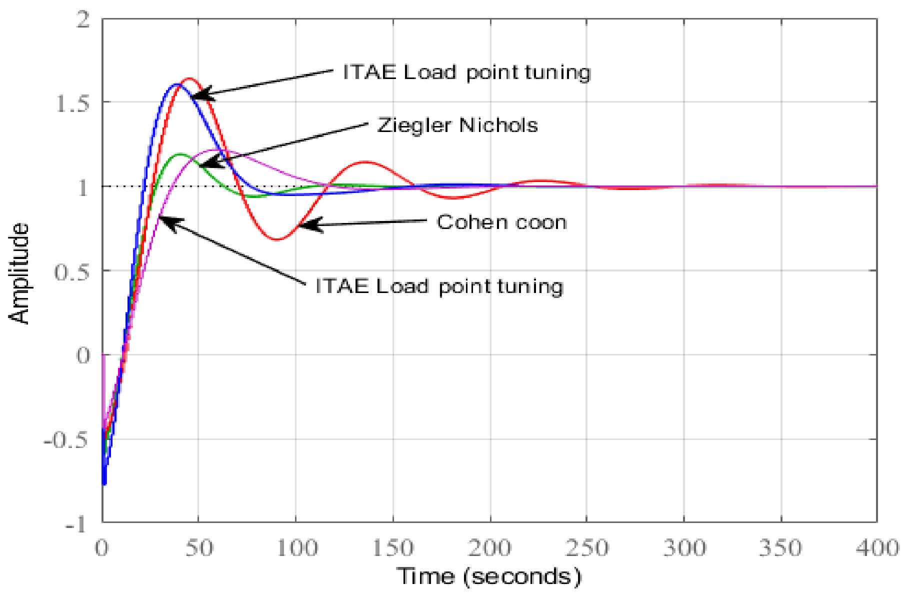


Figure 3.9 Responses of ZN, Cohen Coon, setpoint change, and load point change

Table 3.5 The specifications of ZN, Cohen coon, setpoint, and load point change

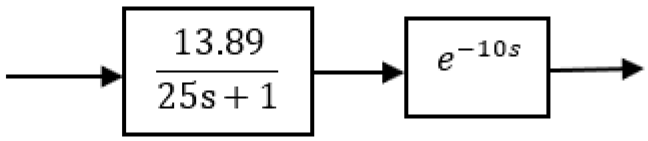
The specifications of the parameter	ZN Tuning	Cohen-coon tuning	ITAE Load point change	ITAE setpoint change
Rise time(sec)	10.2	13.2	10.9	24.8
Pick time(sec)	49.7	53	52.5	68.1
Overshoot (%)	42.4	98.8	85.9	46.3
Settling time(sec)	151.7	423.6	345.2	245.9

The closed-loop system step response with a conventional PID controller is shown in Figure 3.9. Also, different tuning mechanisms for PID controller tuning methods like Ziegler-Nichols, Cohen-Coon, Integral of time-weighted absolute error (ITAE) both load point and setpoint changes, and Matlab/Simulink PID tuning is shown. As it is observed from the figure, with the Matlab/Simulink PID tuning extension toolbox tuned PID controller the step response of the system increased in percent overshoot, this shows the temperature is not in the required value. This has a great influence on the production value. Therefore, manual tuning of the PID Controller is not good for this system [3].

3.12. Digital controller

The time delay cannot be represented by the transfer function of a dynamic model. Due to this, it can be used Pade order approximation using MATLAB to simplify the mathematical model. Continuous-time delay models can approximate, utilizing the Pade command to compute a Pade estimation. Therefore this method is important to compare the true response and approximated response to select the correct estimation to arrange and check the estimation validity. Steps that utilize in Pade approximation [7]:

1. Create an open-loop system model with an output delay (eq. 3.35).



2. By using MATLAB command, it can compute the first-order Pade approximation of G(s).

```
s=tf('s');
P=13.89*exp(-10*s)/(25*s+1);
G1=pade(P,1)
G1 =
  -13.89 s + 2.778
 ------------------
 25 s^2 + 6 s + 0.2
Continuous-time transfer function.
>> bode (G1)
```

This command replaces the all-time delay in G through a first-order approximation.

Therefore, G1 is a second-order transfer function without delays.

3. Compare the Temperature response of the original and approximate models using a bode plot.

```
>> h = bodeoptions;
h.PhaseMatching = 'on';
bodeplot(P,'-b',G1,'-.r',{0.1,10},h)
legend('Exact delay','First-Order Pade','Location','SouthWest')
```

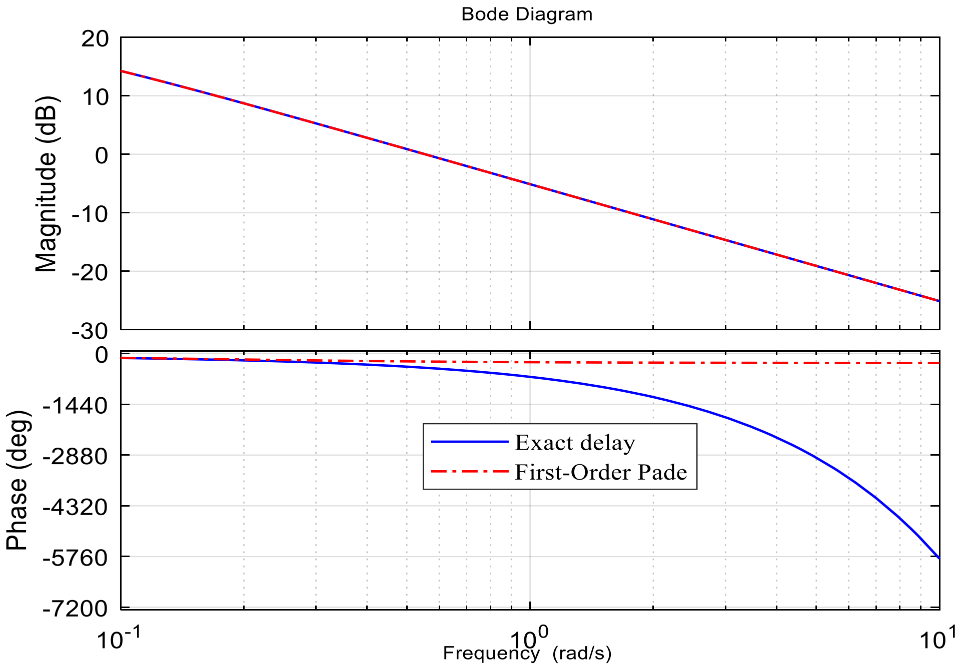


Figure 3.10 Bode diagram for first-order Pade approximation

4. The increment of the Pade approximation order to extend the temperature band in which the phase approximation is good.

```
>> s=tf('s');
P=13.89*exp(-10*s)/(25*s+1);
>> G3=pade(P,3)
G3 =
  -13.89 s^3 + 16.67 s^2 - 8.334 s + 1.667
  ----------------------------------------
  25 s^4 + 31 s^3 + 16.2 s^2 + 3.6 s + 0.12
Continuous-time transfer function.
```

5. Compare the Temperature response of P, G1, and G3.

```
>> bodeplot(P,'-b',G3,'-.r',G1,':g',{0.1 10},h)
legend('Exact delay','Third-Order Pade','First-Order Pade',... 'Location','SouthWest')
```

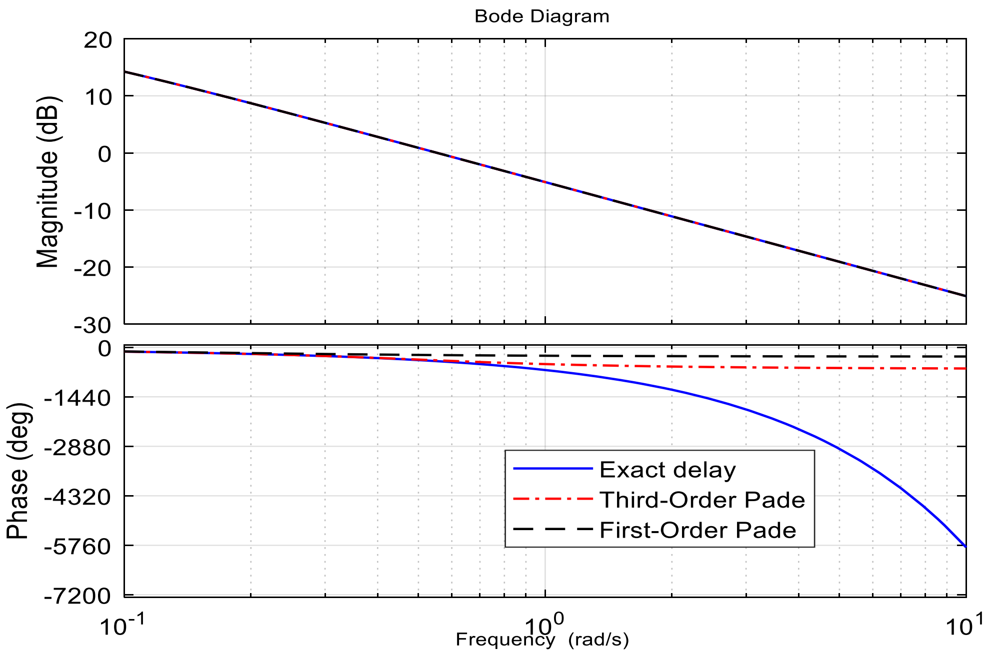


Figure 3.11 Bode diagram for third-order Pade approximation

The phase approximation error cannot be changed by employing the third-order Pade approximation. There is a small improvement using a ninth-order Pade approximation but using too high an approximation order may result in numerical problems and possibly unstable poles.

6. Compare the time-domain responses of the original and approximated system using step input signal.

```
>> step(P,'-b',G3,'-.r',G1,':k')
legend('Exact delay','Third-Order Pade','First-Order Pade',... 'Location','Southeast')
```

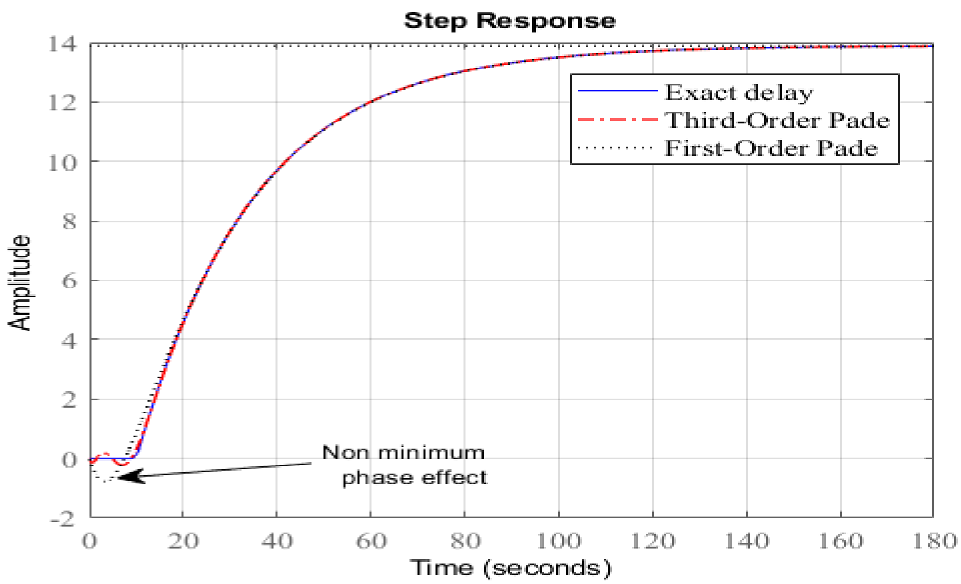


Figure 3.12 Step response of the original and Pade Approximated transfer function

The phase approximation error is not improved by using a third-order Pade Approximation. There is a small development in using a ninth-order Pade approximation but using very high an approximation order may result in numerical issues and may be unstable poles will occur. Using the Pade estimation presents a non-minimum phase artifact ("wrong way" impact) within the beginning temporal response. The impact is reduced within the higher-order estimation. Increasing the Pade estimation arrangement expands the frequency band where the estimation is good. Thus for this Simulink model, it can be used first-order Pade Approximation of the process model [7].

$$G(s) = \frac{-13.89s + 2.778}{25s^2 + 6s + 0.2} \qquad 3.54$$

3.13. MATLAB/Simulink Model of the System

The SISO Open-loop system of MATLAB/Simulink model of the PID and Fractional Order controllers for the closed-loop system is discussed below.

3.13.1. MATLAB/Simulink Model of Open-Loop System

The Simulink model of the SISO systems of the plant without a controller is shown in figure 3.13.

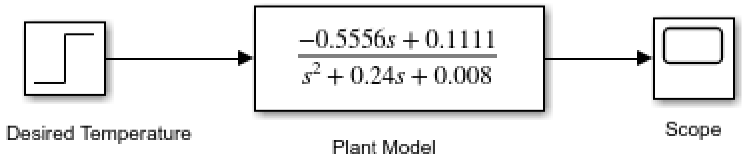


Figure3.13 The Simulink model of the plant without controller

3.13.2. MATLAB/Simulink Model of PID Controller

In this thesis work, the gains of the PID controller are tuned by using the GA in MATLAB. The robustness states that the capacity to withstand or the system response shows the desired value or performance despite the existence of important procedure uncertainty. The closed-loop plant for SISO temperature control of electric furnace with the PID controller is implemented in MATLAB/Simulink as described in figure 3.14 below.

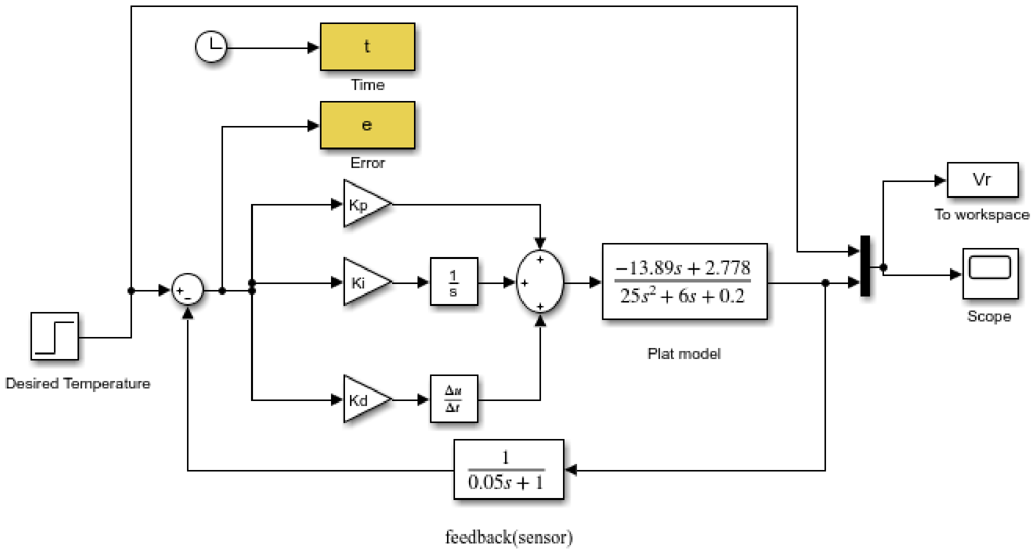


Figure 3.14 Simulink model of the electric furnace with PID controller

3.13.3. GA Tuned of PID Controller

The GA-based PID temperature control of the electric furnace for the tempering process of glass is implemented. The details of the GA optimization parameters setting used in the simulation for the tuning of PID controller of temperature control for electric furnace are shown below in Table 3.6.

Table 3.6 Parameter settings of the genetic algorithm optimization

Parameter	Type and value
Maximum generations	40
Population size	30
Selection	Uniform
Mutation	Uniform
Crossover	Single point crossover
No. of parameters	3
Lower boundary	[0.001 0.001 0.001]
Upper boundary	[10 10 10]

The graphical result of the optimized object function (IAE) parameter of the PID controller of the system is shown in Figure 3.15. Its value after many times re-optimization is 223.648.

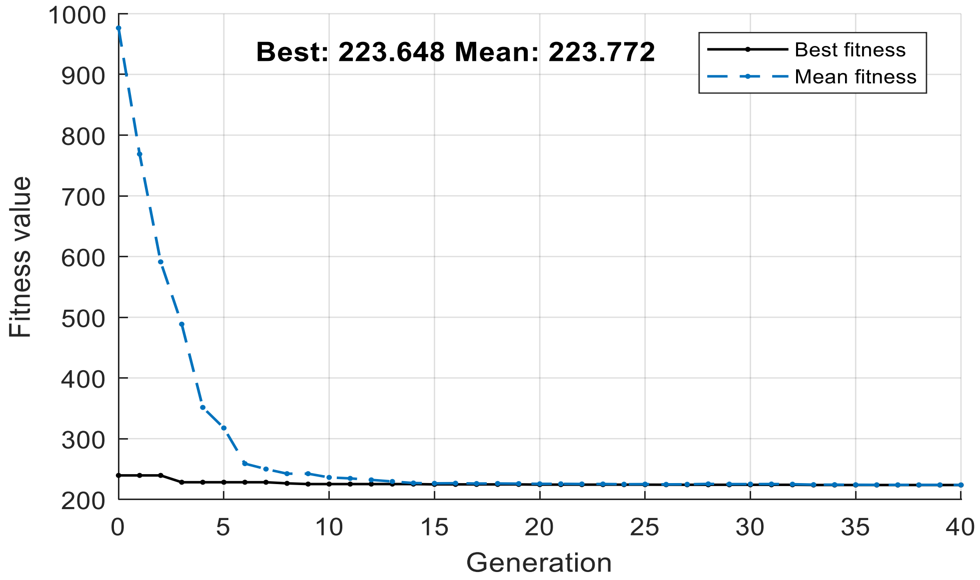


Figure 3.15 Fitness values vs generation for object function

The model of the electric furnace temperature control of PID is performed in GA optimization. The parameters of the furnace temperature control with GA-based PID controller gains (Kp, Ki, Kd) are obtained as shown below in Table 3.7.

Table 3.7 The GA optimized parameters of the PID controller

Controller	Optimized parameters	Optimized values
PID	Kp	0.042
	Ki	0.0019
	Kd	0.898

3.13.4. MATLAB/Simulink Model of Fractional Order Controller

In this thesis work, the gains of the Fractional Order controller are tuned by using the GA in MATLAB. The robustness states that the capacity to withstand or the system response shows the desired value or performance despite the existence of important procedure uncertainty. The closed-loop plant for SISO temperature control of electric furnace with the Fractional Order PID controller is implemented in MATLAB/Simulink as described in figure 3.16 below.

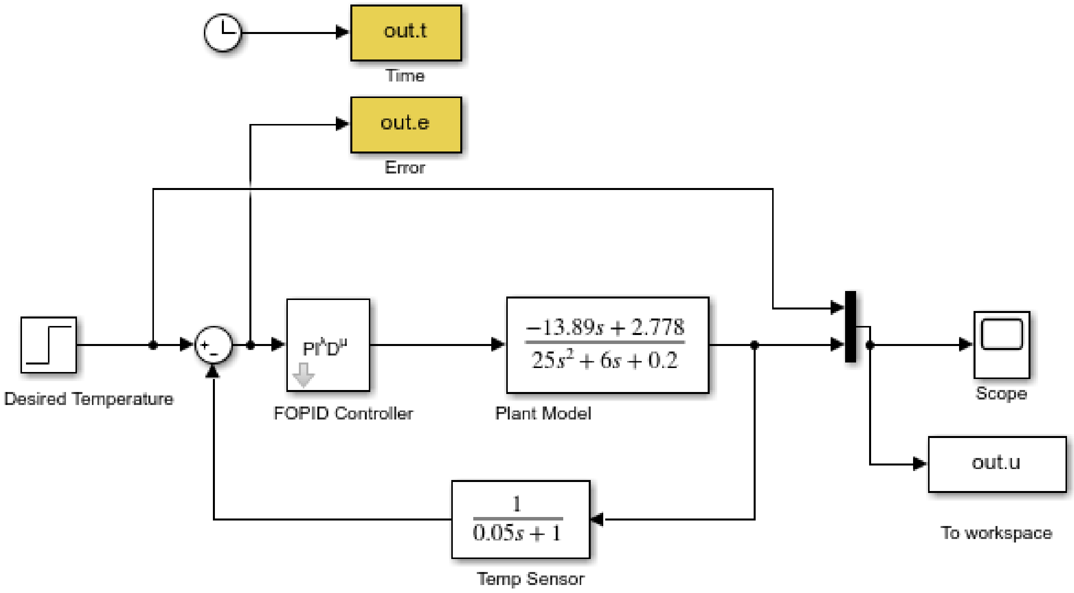


Figure 3.16 Simulink model of the electric furnace with FOPID controller

3.13.5. GA Tuned of FOPID Controller

The GA-based FOPID temperature controller is implemented. The details of the GA optimization parameters setting used in the simulation for the tuning of the FOPID controller of temperature control for electric furnace are shown below in Table 3.8.

Table 3.8 Parameter settings of the genetic algorithm optimization

Parameter	Type and value
Maximum generations	40
Population size	30
Selection	Uniform
Mutation	Uniform
Crossover	Single point crossover
No. of parameters	5
Lower boundary	[0.001 0.001 0.001 0.001 0.001]
Upper boundary	[10 10 10 10 10]

The graphical result of the optimized object function (IAE) parameter of the FOPID controller of the system is shown in Figure 3.17. Its value after many times re-optimization is 81.9136.

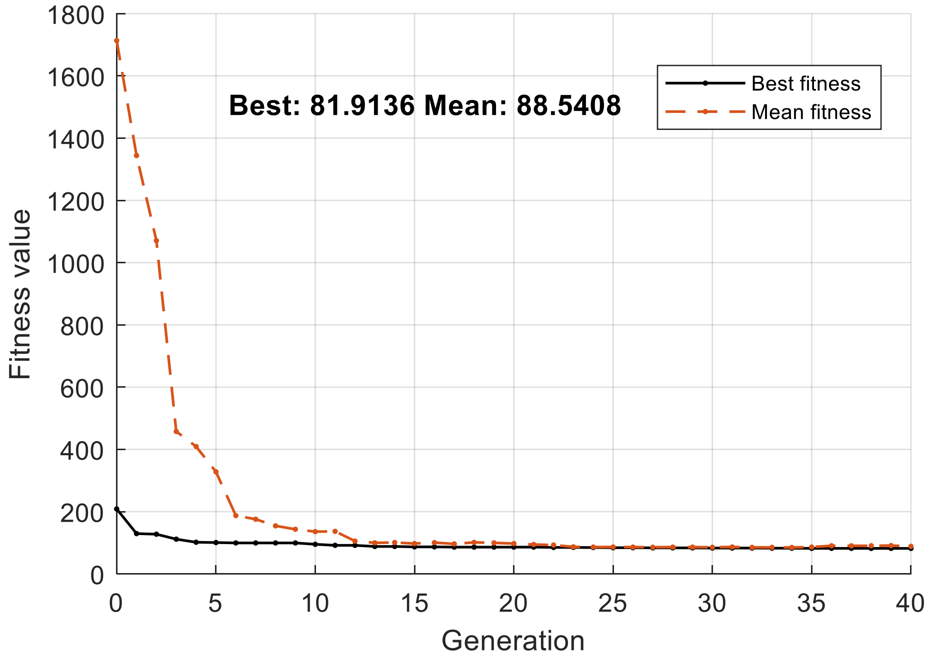


Figure 3.17 Fitness values vs generation for object function of FOPID

The model of the temperature control of the FOPID controller is performed in GA optimization. The parameters of the FOPID temperature controller based on GA (Kp, Ki, Kd, λ, μ) are obtained as shown below in Table 3.9.

Table 3.9 The GA optimized parameters of the FOPID controller

Controller	Optimized parameters	Optimized values
Fractional Order PID	Kp	0.0364
	Ki	0.002
	Kd	0.0256
	λ	0.9989
	μ	0.1

3.14. Design of Model Reference Adaptive Control

The common idea of MRAC is to form a closed-loop controller with parameters that can be upgraded to alter the response of the system. MRAC is an adaptive control technique where the performance purposes are given in terms of a model reference. The model is selected in such a way that it gives the idealized reaction of the method which is desired to the system. In this category, the adaptation component exactly adjusts the controller parameter which can control the given plant system to make following to the model reference which includes a desired performance [7].

The error or error arises from the contrast between the plant output and the model reference output which is acts as input to the adjustment mechanism. The parameters of the controller are consequently adjusted by the adaptation component or the adjustment mechanism in such a way that the execution of the method yield matches that of the model. The parameter modification component can be the MIT rule or the Lyapunov stability approach. A steady-state closed-loop system cannot be guaranteed by the MIT rule. So the Lyapunov stability technique is utilized in this proposition, which avoids the uncertainty issues which is shown inside the slope approaches or MIT rule. MRAC has two loops: an inner loop that is a standard control loop comprising of the plant and the controller, and an external loop that changes the parameters of the controller in such a way as to drive the error between the model yield and plant yield to zero. The MRAC construction comprises four essential parts: the plant, the controller, the reference model, and the adjustment mechanism as shown in figure 3.18 [7].

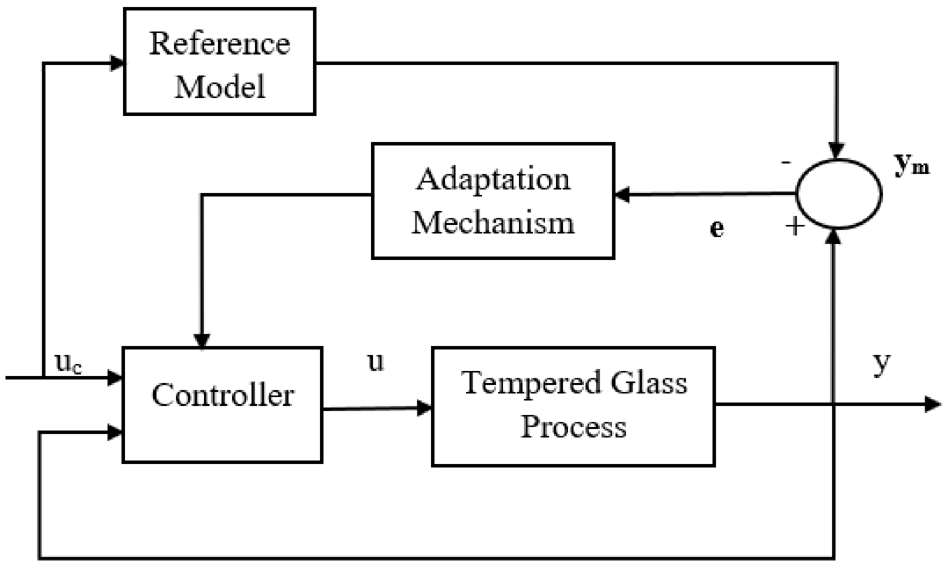


Figure 3.18 Model Reference Adaptive Control structure

Reference Model: This portion of the controller indicates the specified performance of the system. It should reflect the performance details of the system in control tasks. In reality, the adaptive controller compelled the real system to act as a reference model. In this thesis work, the determination of the model reference is chosen as a second-order system [7].

$$Y_m(s) = \frac{25}{s^2 + 100s + 25}$$

The dynamic behavior of the second-order system can be described in terms of two parameters damping ratio and natural frequency (ω_n) if the damping ratio is between zero and one, the closed-loop poles are conjugate and lie in the left-half s plane. The system is called underdamped, and the transient response is oscillatory. The damping ratio equals zero; the system is called critically damped. Over damped system corresponds to damping ratio greater than one. The transient response of a critically damped and overdamped system does not oscillate. When the damping ratio equals zero, the transient response does not die out.

Controller: It is commonly parameterized by a few adaptable parameters. The control law is linear in terms of the adaptable parameters. Adaptive controller design commonly requires linear parameterization to obtain an adjustment component with guaranteed stability and following convergence. The values of these control parameters are dependent on adjustment gain which in turn changes the control algorithm of the adjustment component.

Adaptation Mechanism: It is utilized to alter the parameters inside the control law. Adjustment law looks for the parameters such that the response of the model should be the same as the reference model. It is outlined to guarantee the stability of the control system as well as the combining of the following errors to zero [19].

The component for altering the parameters in a model reference adaptive system can be obtained in two ways: by utilizing a gradient method or by applying stability theory. The adjustment law uses the mistake between the method and the show yield, the strategy yield, and the input flag to vary the parameters of the control system. These parameters were changed to play down the mistake between the method and the reference. All The basic logic of designing a linear controller assumes information of the plant dynamic show and required performances [7], [38].

$$J(\theta) = \frac{1}{2}e^2 \tag{3.55}$$

To reduce J, it is sensible to move the parameters within the heading of the negative slope of J, that is:

$$\frac{d\theta}{dx} = -\gamma\frac{\partial J}{\partial \theta} = -\gamma e\frac{\partial e}{\partial \theta} \tag{3.56}$$

where J is the optimization function, γ is the adaptation gain

The $\frac{\partial e}{\partial \theta}$, which is so-called the affectability derivative of the system, tells how the error is subjective by the flexible parameter. In case it is expected that the parameter changes are slower than the other factors within the system, at that point the derivative $\frac{\partial e}{\partial \theta}$ can be assessed under the assumption that θ is consistent.

$$G(s) = \frac{Y(s)}{U(s)} = \frac{-13.89s + 2.778}{25s^2 + 6s + 0.2}, \qquad G_m(s) = \frac{Y_m(s)}{U_m(s)} = \frac{s+5}{s^2+4s+5}$$

The adaptive control law is:

$$u(t) = \theta_1 u_c(t) - \theta_2 y(t) \qquad (3.57)$$

where:

u(t) is the adaptive control, θ_1 and θ_2 are the adaptive control parameters, $y(t)$ is the system output.

$$U(s) = \frac{Y(s)}{G(s)} = \theta_1 u_c(s) - \theta_2 Y(s) \qquad (3.58)$$

$$Y\frac{25s^2 + 6s + 0.2}{-13.89s + 2.778} = \theta_1 U_c(s) - \theta_2 Y(s)$$

$$\frac{Y(s)}{U_c(s)} = \frac{-13.89s + 2.778}{25s^2 + (6 + \theta_2)s + \theta_2 \ + 0.2}$$

$$E(s) = Y(s) - Y_m(s) \qquad (3.59)$$

where, $E(s)$ is the error between the actual system output and the model reference output.

$$\frac{Y(s)}{U_c(s)} = \frac{-13.89s\theta_1 + 2.778\theta_1}{25s^2 + (6 + \theta_2)s + \theta_2 \ + 0.2}$$

But $G_m(s) = \frac{Y_m(s)}{U_c(s)}, Y_m(s) = G_m(s)U_c(s)$

Equation (3.60) becomes:

$$\mathrm{E(s)} = \frac{-13.89s\theta_1 + 2.778\theta_1}{25s^2 + (6+\theta_2)s + \theta_2 + 0.2}\ \mathrm{U(s)} - \frac{s+5}{s^2+4s+5} \qquad (3.60)$$

$$\frac{\partial E}{\partial \theta_1} = \frac{-13.89s\theta_1 + 2.778\theta_1}{25s^2 + (6+\theta_2)s + \theta_2 + 0.2} = G_m(s)U_c(s) \qquad (3.61)$$

$$\frac{\partial E}{\partial \theta_2} = \frac{\theta_1(-13.89s + 2.778)^2}{[25s^2 + (6+\theta_2)s + \theta_2 + 0.2]^2}\ U_c(s) = \frac{Y^2(s)}{\theta_1 U_c(s)} \qquad (3.62)$$

3.14.1. MIT Rule

The MIT rule can be used to design a controller with a MRAC technique for any system.

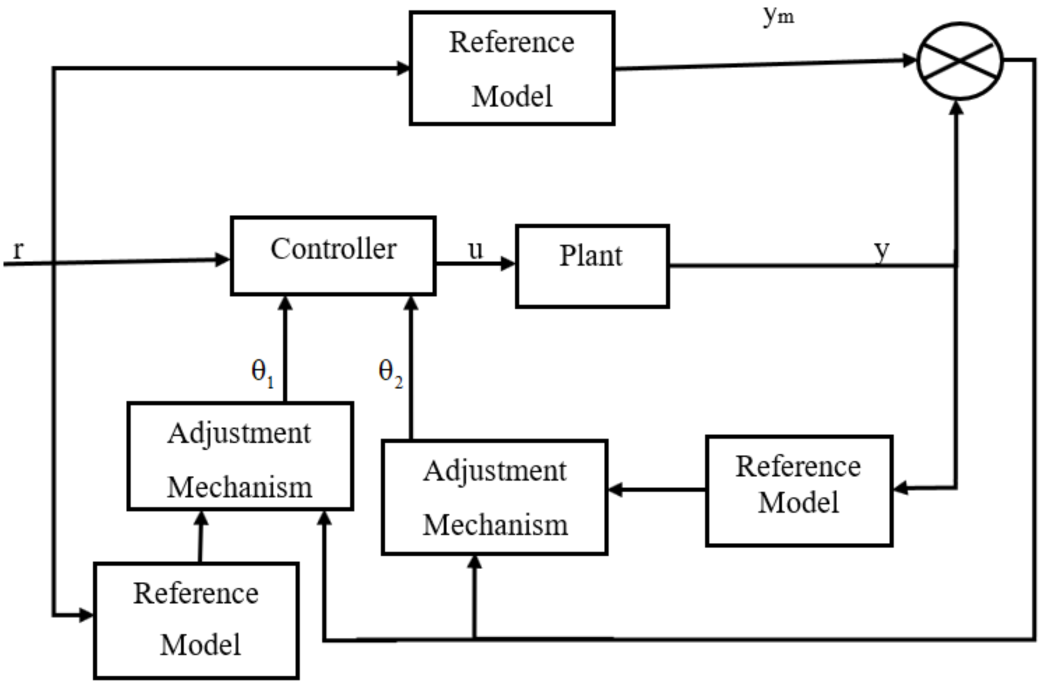


Figure 3.19 The Model of MRAC using MIT rule

The following rules indicate how the error is changing for a parameter θ.

$$\frac{d\theta}{dt} = -\gamma \frac{\partial e}{\partial \theta} \tag{3.63}$$

$$\frac{d\theta_1}{dt} = -\gamma \frac{\partial e}{\partial \theta_1} e = -\gamma_1 e G_m(s) U_c(s) \tag{3.64}$$

$$\frac{d\theta_2}{dt} = -\gamma \frac{\partial e}{\partial \theta_2} e = -\gamma_2 e \frac{Y^2}{\theta_1 U(s)} = \gamma_1 e G_m(s) Y(s) \tag{3.65}$$

3.14.2. Lyapunov Stability Method

The Lyapunov stability procedure is a basic course of adaptive control. This activity is to find the Lyapunov function and an adjustment component in such a way that the error between the plant and the model goes to zero. Moreover, this method guarantees the stability of the control parameters of the system. When planning a MRAC utilizing the Lyapunov rule, it is chosen: the reference model, the controller structure, and the tuning picks up for the alteration component [28].

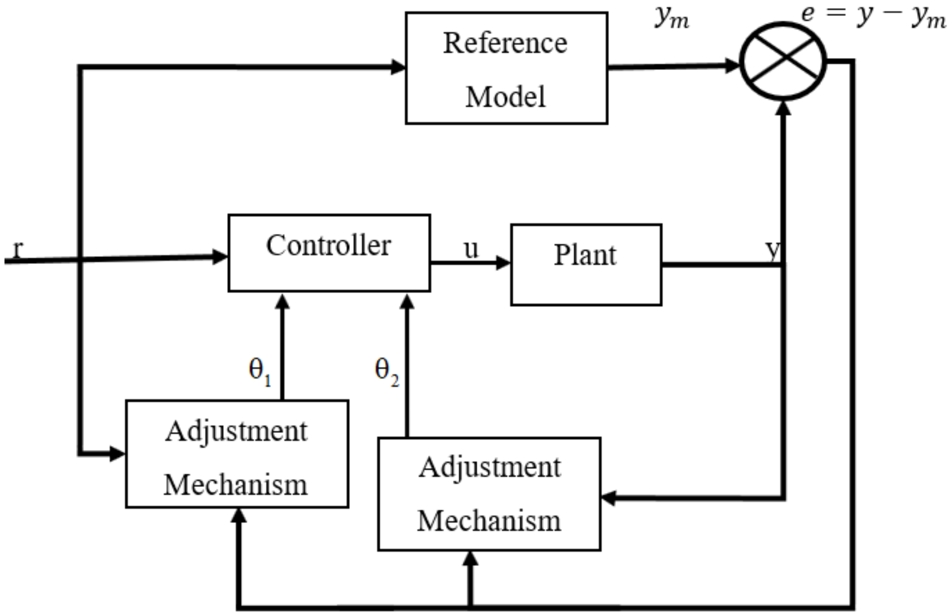


Figure 3.20 The Block diagram of MRAC using Lyapunov rule

$$\frac{dy_{m(t)}}{dt} = -a_m y_m(t) + b_m u_c(t) \quad (3.66)$$

The system to be controlled is expressed as a first-order model:

$$\frac{dy_{(t)}}{dt} = -ay(t) + bu(t) \quad (3.67)$$

From the combination of feedforward and feedback of control law:

$$u(t) = \theta_1 u_c(t) - \theta_2 y(t) \quad (3.68)$$

And also, the error will be:

$$e(t) = y(t) - y_m(t) \quad (3.69)$$

where,

y - is an output of the plant

y_m - is output model reference

θ_1 and θ_2- are the controller parameters

u_c - is an input of the model reference

e - is the error difference of actual plant and model reference

u - is the control signal output

Consequently, it is required to create the error goes to zero as much as conceivable, it is characteristic to drive a differential condition for the mistake as takes after:

$$\dot{e} = \dot{y} - \dot{y}_m x \quad (3.70)$$

$$= -a_y + b_u - (a_m y_m + b_m u_c)$$

$$= -a_y + b_u - a_m y_m - b_m u_c$$

Substitute the control law u:

$$\dot{e} = -a_y + (\theta_1 u_c - \theta_2 y) + a_m y_m - b_m u_c$$

$$= -a_y - b\theta_2 y + a_m y_m - (b\theta_1 - b_m)u_c$$

$$= -a_m e - (b\theta_2 + a - a_m) + (b\theta_1 - b_m)u_c$$

To carried out the Lyapunov function, although the parameter alteration component or adjustment mechanism drives the parameters θ_1 and θ_2 to their desired values, a quadratic function is selected as a Lyapunov rule as discussed below:

$$V(e, \theta_1, \theta_2) = \frac{1}{2}[e^2 + \frac{1}{b\gamma}(b\theta_2 + a - a_m)^2 + \frac{1}{b\gamma}(b\theta_1 - b_m)^2 \qquad (3.71)$$

This Lyapunov work is zero when e is zero and the controller parameters are rise to the adjusted values. The overall derivative of V yields:

$$\dot{V} = e\frac{de}{dt} + \frac{1}{\gamma}(b\theta_2 + a - a_m)\frac{d\theta_2}{dt} + \frac{1}{\gamma}(b\theta_1 - b_m)\frac{d\theta_1}{dt} \qquad (3.72)$$

$$= -a_m e^2 + \frac{1}{\gamma}(b\theta_2 + a - a_m)\left(\frac{d\theta_2}{dt} - \gamma y e\right) + \frac{1}{\gamma}(b\theta_1 - b_m)\left(\frac{d\theta_1}{dt} + \gamma U_c e\right)$$

It is negative, the above-mentioned quadratic work may be a Lyapunov work. If the parameter is updated by equating zero.

$$\frac{d\theta_1}{dt} = \gamma_1 y e \qquad (3.73)$$

$$\frac{d\theta_2}{dt} = \gamma_2 U_c e \qquad (3.74)$$

3.14.3. RHP Zeros

There are distinctive causes for the RHP zeros. Physical non-collocation of linearization of nonlinear systems around a working point may deliver RHP zeros in a nonstop single-input-single-output system (SISO), such as the four-wheeled car, time-delay system, and the altered pendulum on a cart given in [39] in a multi-input multi-output (MIMO) system, however, the RHP zeros are the effects of competing slow and fast dynamics which have opposite signs [40]. Perfect tracking for systems with RHP zeros requires either the unstable pole-zero Cancellation which should be avoided because it will make the system internally unstable or a non-causal controller. To overcome this constraint and achieve a better, faster transient response, the only option seems to be the open-loop compensation, known as the feed-forward method [41].

3.14.4. Design of Feed-Forward Control

As shown in figure 3.21 the design of feedforward control with feedback controls.

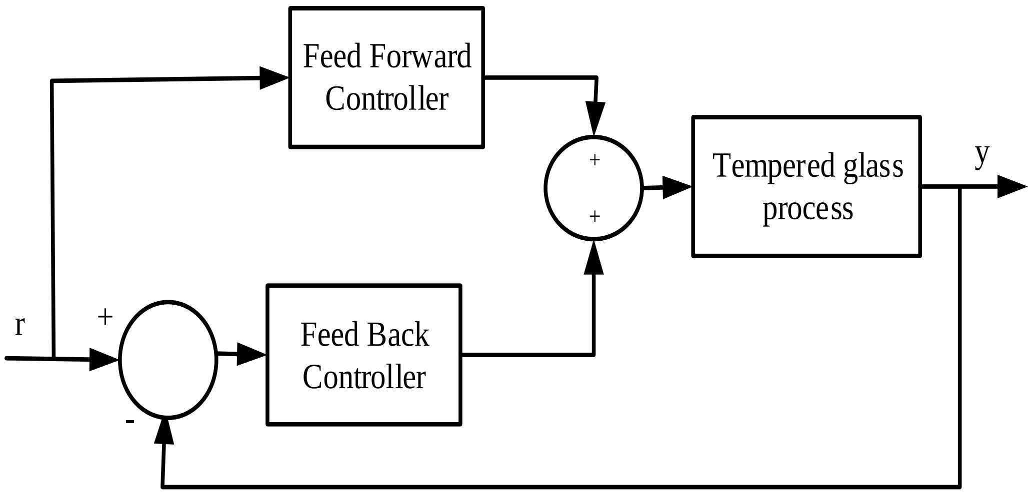


Figure 3.21 combination of feedback and feed-forward control design

In the above figure 3.21, the overall system of feedforward and feedback control for the tempering process glass of temperature control for the electric furnace is designed. The well-known optimal control solution for a precise system with RHP zero will be derived by using the Hurwitz row stability technique [7]. As shown in the above figure 3.21 feedforward control is intended to parallel with feedback control by updating the control law u_1 and feedforward is a PI type of form u_2, that is:

$$u_1 = \theta_1 u_c + \theta_2 (u_c - y), \text{ and } u_2 = k_p u_c(t) + k_i \int u_c\,(t)dt$$

3.15. MATLAB/Simulink Model of MRAC with Feedforward PI Controller

The first order Pade approximated transfer function for SISO Tempering process of Electric Furnace which is given above in equation (3.54) as,

$$G(s) = \frac{-13.89s + 2.778}{25s^2 + 6s + 0.2}$$

3.15.1. MATLAB/ Simulink Model of Open-Loop Reference Model

The Simulink model of the systems of model reference without a controller is shown below in figure 3.22.

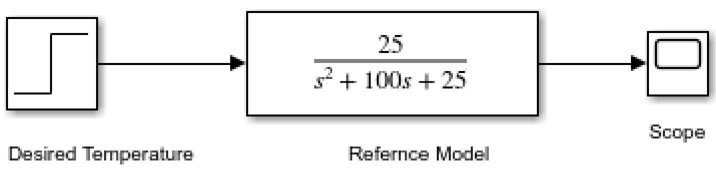


Figure 3.22 The Simulink model of open-loop SISO Reference Model

3.15.2. MATLAB/Simulink Model of MIT Rule with FFPI Controller

The MIT Feed-Forward control developed in MATLAB Simulink for controlling the Temperature of the Electric Furnace is given in the figure below.

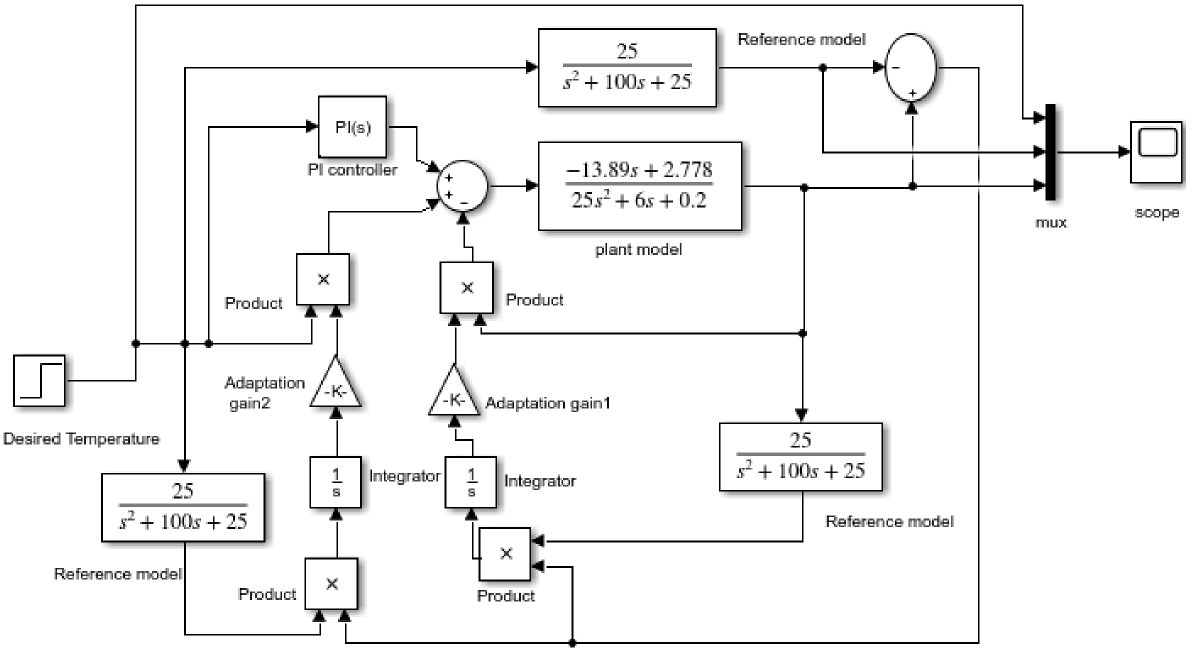


Figure 3.23 Simulink model of MIT rule with Feed-Forward controller

3.15.3. MATLAB/Simulink Model of Lyapunov Rule with FFPI controller

The Lyapunov rule with Feed-Forward controller is developed in MATLAB Simulink for controlling SISO Temperature of Electric Furnace is shown in the figure below.

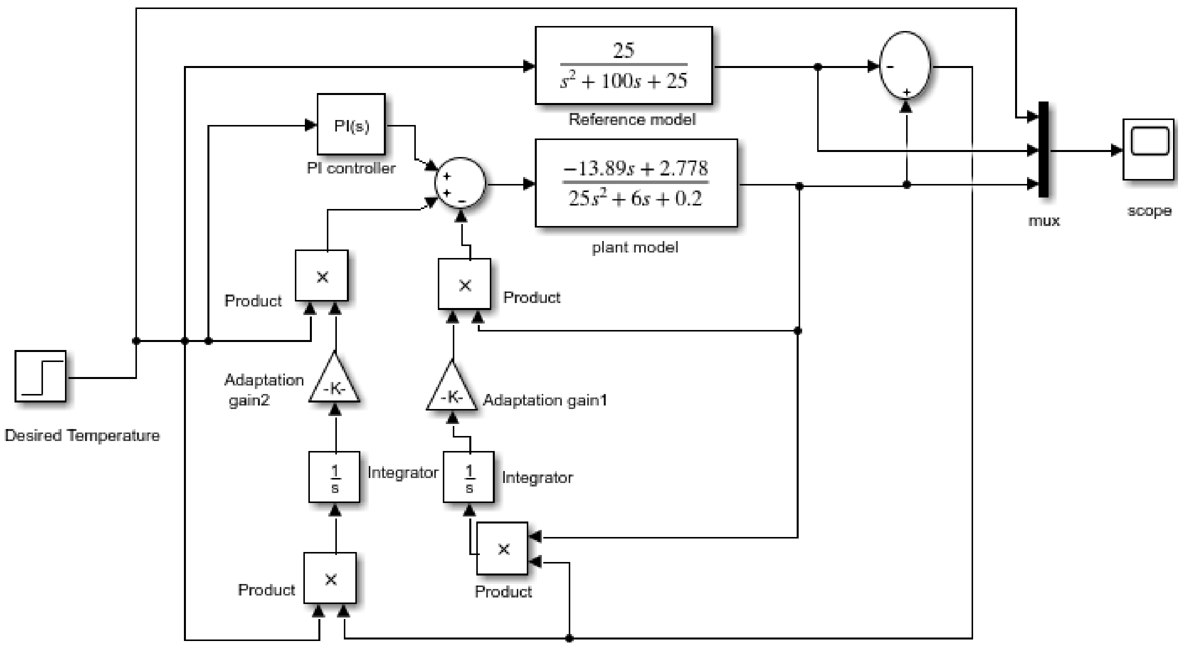


Figure 3.24 Simulink model of Lyapunov rule with Feed-Forward controller

CHAPTER FOUR

RESULTS AND DISCUSSIONS

The electric furnace temperature control for glass tempering or hardening was designed, modeled, and analyzed. The results and discussions are in the section of this chapter by using the MATLAB simulation output results. This is concerned with the Matlab simulations output results of the closed-loop control system utilizing GA-based PID, and GA-based FOPID controllers. The simulation is analyzed by step response, parameter variation, and disturbance of the closed-loop system. Simulink is a tool kit expansion of the MATLAB program and design. The mathematical model for control of the temperature for the electric furnace for tempering process of the glass and the design of different techniques of MRAC like the MIT rule and Lyapunov stability theory and RHP Zeros are discussed in the previous chapter three in detail. Furthermore, the design and MATLAB/Simulations of MRAC with Feed-Forward controller of the temperature for the electric furnace of tempering process of the glass is done.

4.1. Open-Loop Response of Electric Furnace with Time Delay Consideration

The SISO system response of the open-loop or without feedback controller is shown in the figure below. The response has strictly increased, this means the output response does not get settled for desired temperature value of the electric furnace for the tempering glass process. This increasing temperature value has an excessive influence on the product quality of the glass. In this manner, there is the requirement for a closed-loop system of a controller designed to temperature control at the desired value and decay to the output value at the required time to reach the desired temperature and with minimum overshoot. The temperature to reach a desired value required small time domain specifications.

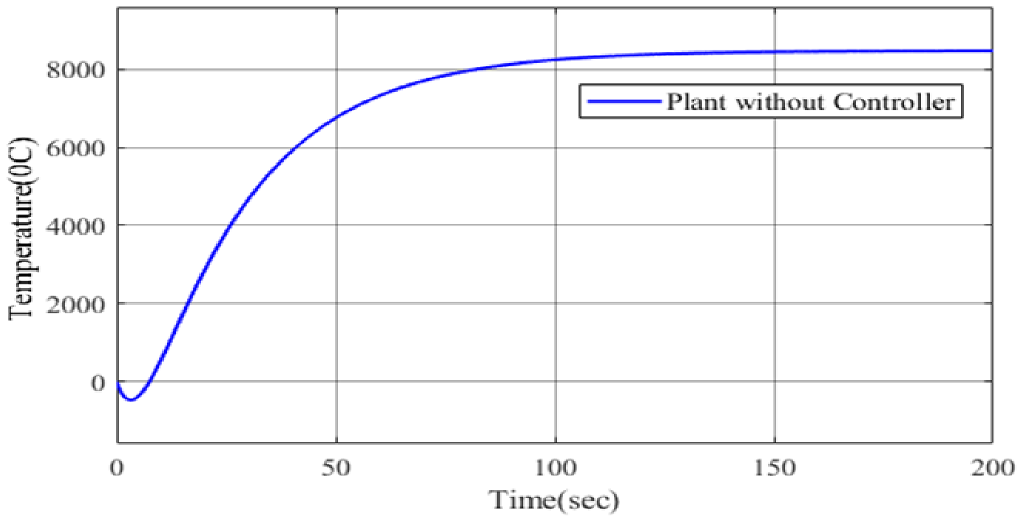


Figure 4.1 The open-loop step response with time delay consideration

4.2. The Result of Open Loop system Response of the Model Reference

The result of the open-loop output response of the SISO system of the model reference is shown in the figure below. The system response without a controller is become stable and tracks the desired value of the system. The performance specifications of the response are: rise time is 51.65 sec, settling time is 140.4 seconds, and percentage overshot is 0%.

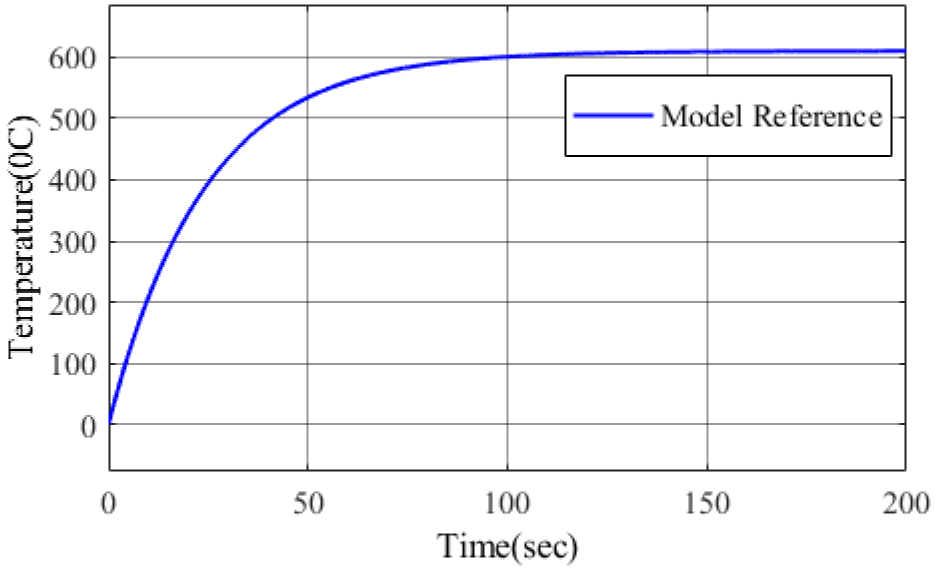


Figure 4.2 Step response of the open-loop system of the model reference

4.3. The Result of GA Based PID Controller

In this work, the GA-based PID controller is used as a benchmark to compare the MRAC with the Feed-Forward controller and GA-based FOPID controllers of the performance with that of the desired temperature control of the electric furnace for tempering glass. The SISO system response of closed-loop temperature control of the electric furnace for glass tempering process in figure 4.3 is shown below, which is found from the Matlab/Simulink of the SISO system model, to control the temperature of the electric furnace for glass tempering. It takes 135.7 seconds to reach the desired temperature of the system. The response has a high settling time to reach the temperature of 610^0C and there is no temperature difference. The temperature of the system is very high, it takes time to reach this desired value of temperature. Thus, the GA-based PID controller response is good for this system.

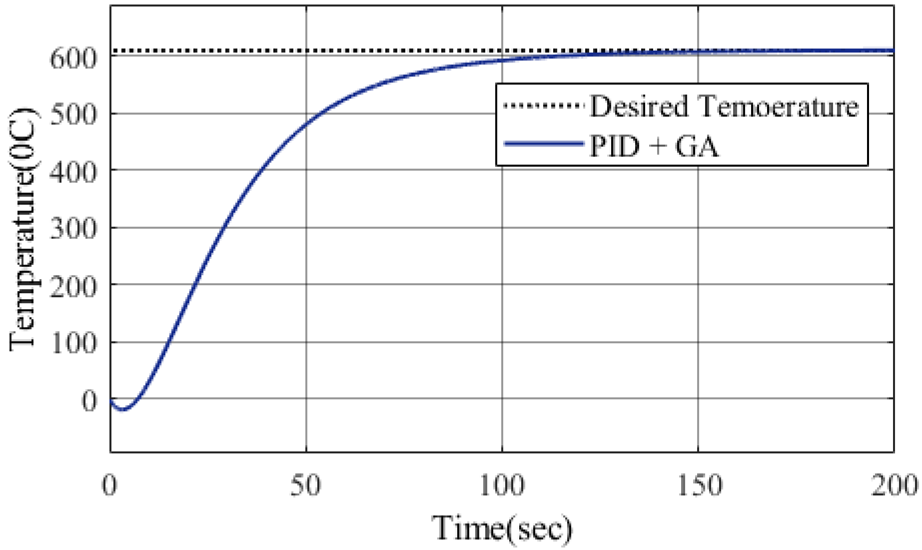


Figure 4.3 The Response of GA-based PID temperature Controller of the system

4.4. The Result of MIT Rule without Feed-Forward Controller

The result of the SISO system response of the closed-loop for temperature control of the electric furnace for the glass tempering process is shown in figure 4.4, which is gained from the Matlab/Simulink of the SISO system model to the temperature control of the electric furnace. The response as shown in figure 4.4 does follow the model reference but it has a temperature difference at the initial point before reached to the desired value of the temperature, and it has a large settling time to reach the maximum temperature value of the system. Hence, the MIT rule without FFPI is not a good response compared to the GA-based PID controller which is discussed above.

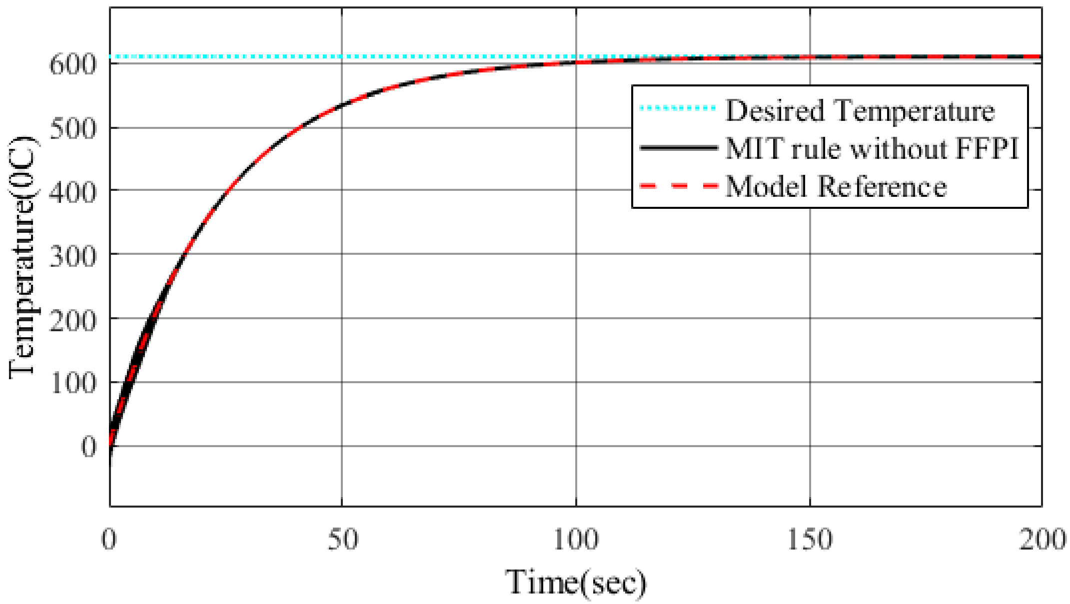


Figure 4.4 The response of MIT rule without FFPI Controller

4.5. The Result of MIT Rule with Feed-Forward PI Controller

The MIT rule with FFPI controller result of the SISO closed-loop system response of the temperature control of electric furnace for glass tempering process shown in figure 4.5 below. It is found from the Matlab/Simulink of SISO system model to control the temperature of the electric furnace. The output response is shown in figure 4.5 tracks the model reference and slowly approaches the desired value temperature of the system. Therefore, MIT's rule-based feed-forward controller has almost similar responses to that of MIT rule without feed-forward PI controller but there is a steady-state error in MIT rule with feed-forward PI controller of the closed-loop system response.

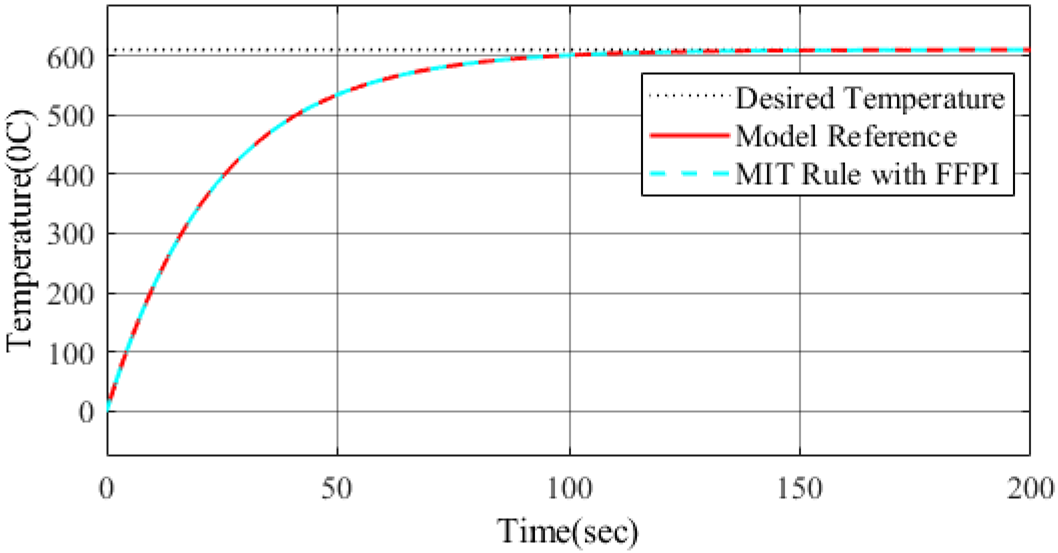


Figure 4.5 The system response for MIT Rule with FFPI Controller

4.6. The Result of Lyapunov Rule without Feed-Forward Controller

The Lyapunov rule result of SISO system response of the closed-loop for temperature control of electric furnace for glass tempering process is shown in figure 4.6, which is gained from the Matlab/Simulink of SISO system model to control the temperature of the electric furnace of the glass tempering process. The response as shown in the figure below does follow the model reference and attends the desired temperature. But it has a temperature error or difference at the initial. So, the Adaptive Model Reference Controller based on the Lyapunov rule without FFPI does not have a good response for a system having RHP zeros.

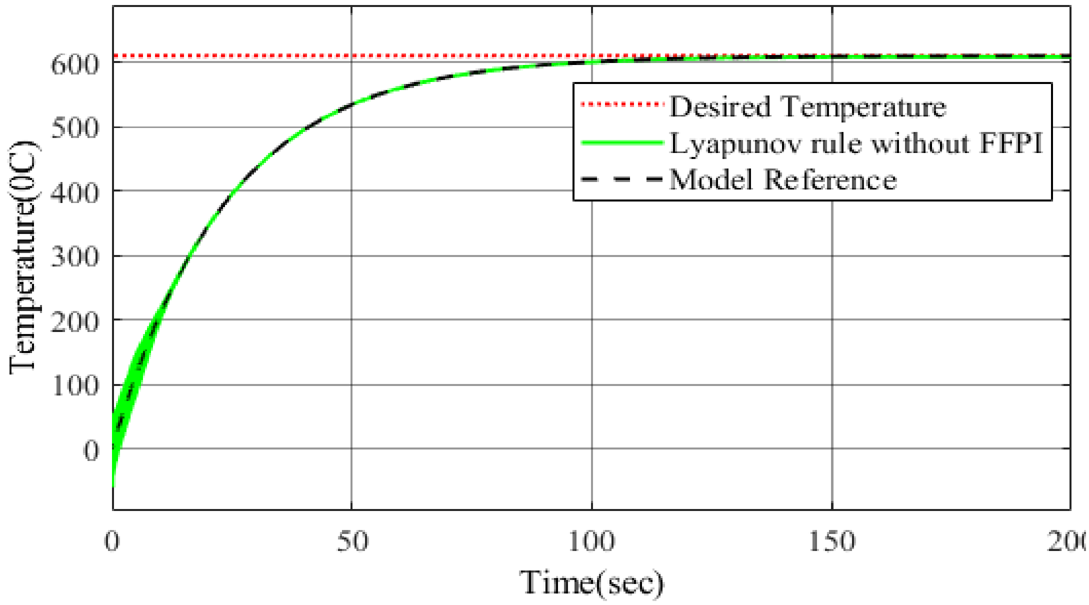


Figure 4.6 The system response of Lyapunov rule without FFPI controller

4.7. The Result of Lyapunov Rule with Feed-Forward PI Controller

The result of the Lyapunov rule for SISO system response of the closed-loop for temperature control of electric furnace for glass tempering process is shown below in figure 4.7. The response is gained from the Matlab/Simulink model of the SISO system of the temperature control of the electric furnace. The response as shown in figure 4.7 follows the reference model and slowly attends to the required value of the temperature. Thus, the Lyapunov rule based on Feed Forward PI Controller has not a good response when compared with the GA-based PID controller.

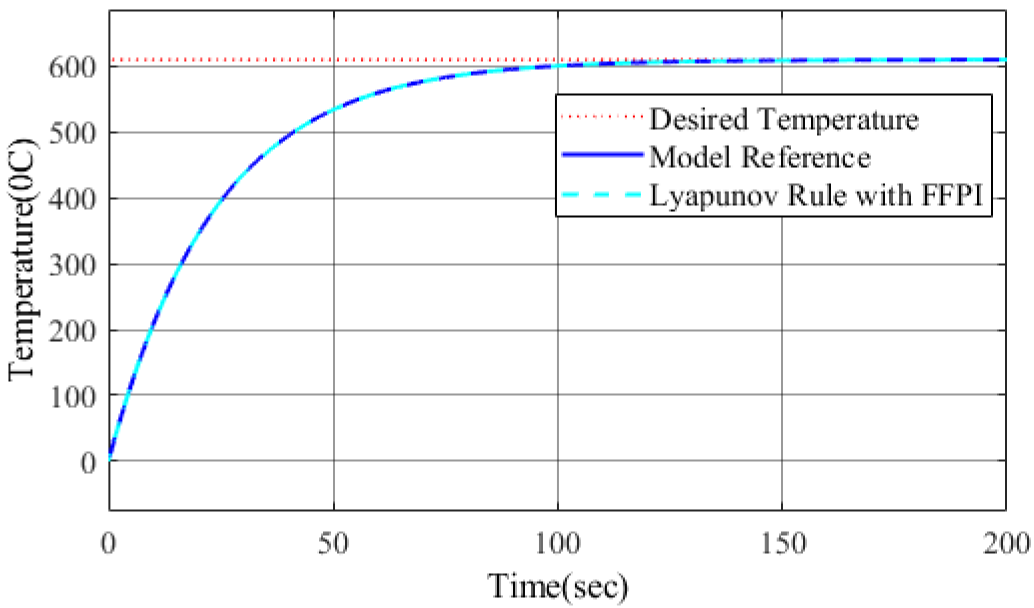


Figure 4.7 The Lyapunov rule with FFPI controller system response

4.8. The Result Comparison of MIT Rule with FFPI and Lyapunov Rule with FFPI

The result of the MIT rule and Lyapunov rule with feed-forward PI controllers of the SISO system responses of the closed-loop for temperature control of the electric furnace for the glass tempering process is shown in figure 4.8. The response is gained from the Matlab/Simulink SISO system model to the temperature control of the electric furnace. The responses as shown in figure 4.8 follow the model reference and slowly approach the required value of temperature. The responses of the MIT rule and Lyapunov rule are followed the model reference. Almost they have a similar response but only a very small difference on some value of performance specifications. Thus, as shown in table 4.1, the time domain specification of the Lyapunov rule with FFPI Controller has a better response than the MIT rule with FFPI controller response as discussed below. But both responses are followed the model reference and overlapped each other, thus it can be taken the MIT rule response and Lyapunov rule response are the same for this system.

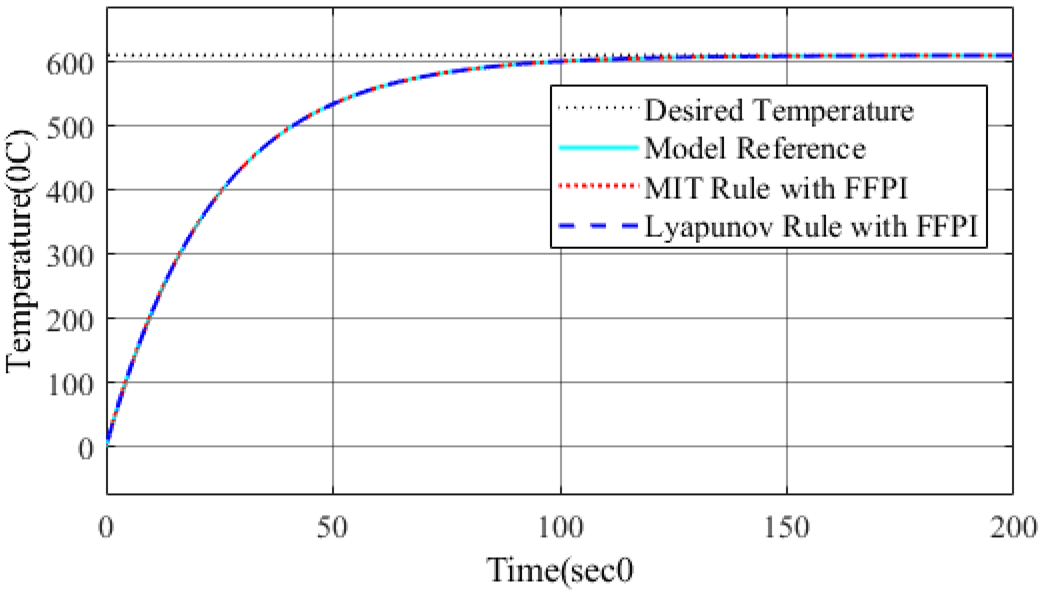


Figure 4.8 Comparison of MIT rule and Lyapunov rule of the system responses

Table 4.1 Performance of MIT rule and Lyapunov rule with FFPI

Performance Specifications	MIT rule with FFPI	Lyapunov rule with FFPI
Rise time	51.65 sec	51.63 sec
Settling time	140.37 sec	140.34 sec
Overshot (%)	0%	0%
Steady state error	0%	0%

4.9. The Result of GA Based Fractional Order Controller

The result of the SISO system response of the closed-loop temperature control of electric furnace for glass tempering process is obtained from the Matlab/Simulink model using Fraction Order PID with GA Tuned as shown below in figure 4.9. The performance or time-domain specifications of the response are the best for this system compared with conventional PID controller, MIT rule with Feed-Forward controller, and Lyapunov rule with Feed-Forward controller. As seen from the response of the system, there is no temperature difference during the tempering glass process. Thus, as seen from the figure below, there is no temperature difference between the various parts of the glass. So, the product of tempered glass will be the best if the temperature is controlled with GA based Fractional Order controller.

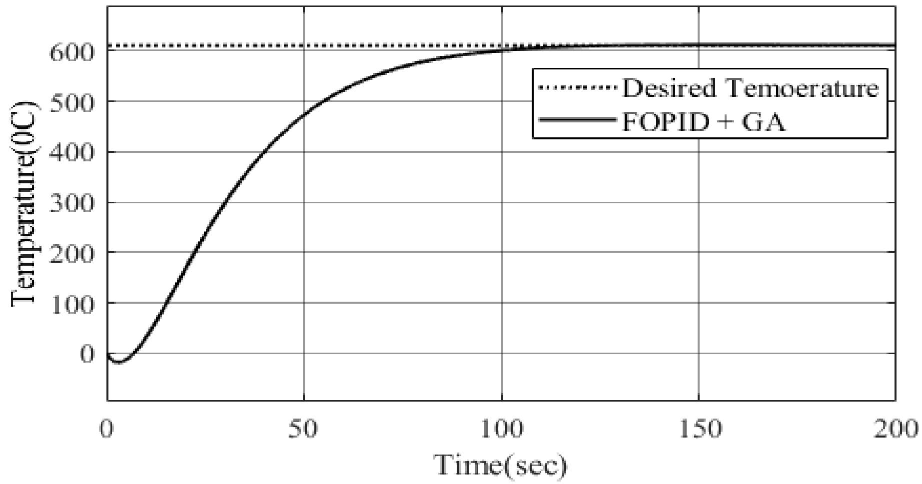


Figure 4.9 The result of GA based FOPID Controller

4.10 Controller Results for Parameter Variation

4.10.1 The Result MIT Rule with Feed-Forward Control for Parameter Variation

The dynamic model of the MIT rule temperature controller of the furnace for the tempering glass process will have the following form:

$$G(s) = \frac{as + b}{cs^2 + ds + e} = \frac{-13.89s + 2.778}{25s^2 + 6s + 0.2}$$

To deal with the robustness of MIT rule with feedforward controller for parameter variation of increasing or decreasing the parameter. From the above transfer function, if parameters a and d decreased by 10%, if parameters b and c increased by 5%, and if parameter e does not change, the response is the same as the previous system output with that of the MIT rule based on FFPI discussed above. As shown in figure 4.10 the output response of the MIT rule-based feed-forward controller has a large time to reach the desired temperature value of the system.

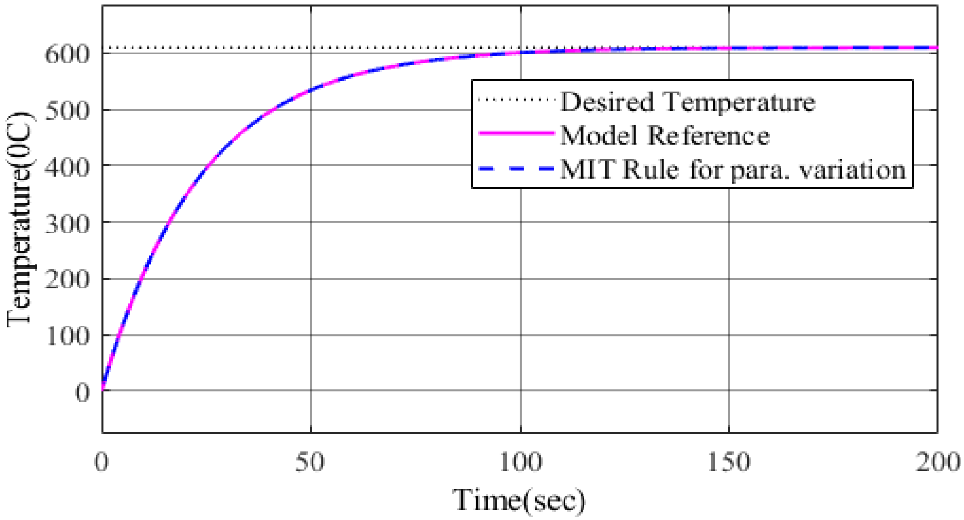


Figure 4.10 The MIT rule with feed-forward PI controller for parameter variation

4.10.2. Lyapunov Rule with Feed-Forward Controller for Parameter Variation

The dynamic model of the Lyapunov rule to control the temperature of the furnace for the tempering glass process will have the following form:

$$G(s) = \frac{as + b}{cs^2 + ds + e} = \frac{-13.89s + 2.778}{25s^2 + 6s + 0.2}$$

To deal with the robustness of the Lyapunov rule with feedforward controller for parameter variation of increasing or decreasing the parameter. From the above transfer function, if parameters a and d decreased by 10%, if parameters b and c increased by 5%, and if parameter e does not change, the response is not changed. As shown in figure 4.11 the time-domain specifications of the system are the same as the original system response of the Lyapunov rule with FFPI.

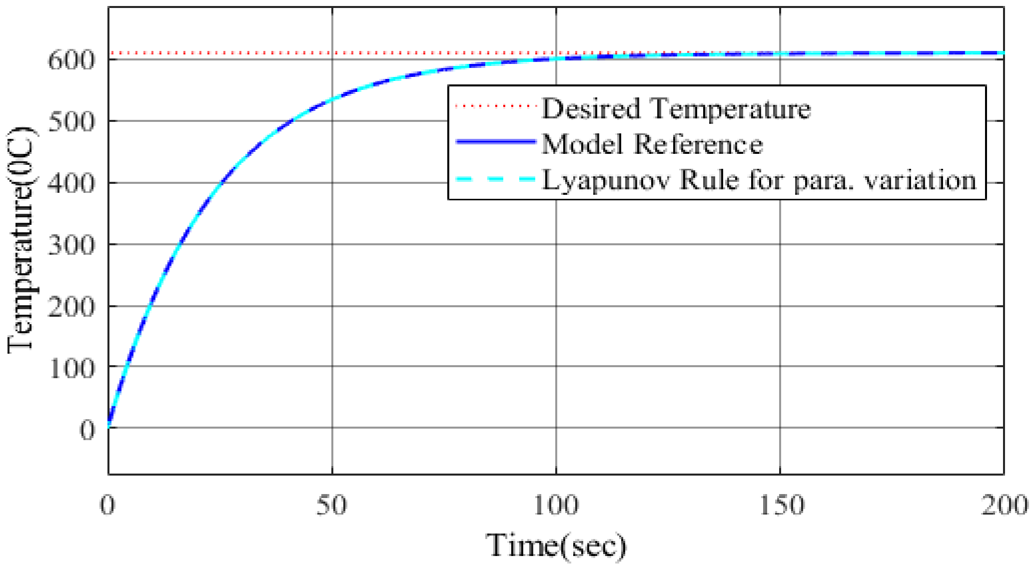


Figure 4.11 The Lyapunov rule with feed-forward PI controller for parameter variation

4.10.3. The Result of GA Based PID Controller for Parameter Variation

The dynamic model of GA based PID to control the temperature of the electric furnace for the tempering glass process will have the following form:

$$G(s) = \frac{as + b}{cs^2 + ds + e} = \frac{-13.89s + 2.778}{25s^2 + 6s + 0.2}$$

Now, to deal with the robustness of PID controller with parameter variation of increasing or decreasing the parameter of the system. From the above transfer function, if parameters a and d decreased by 10%, if parameters b and c increased by 5%, and e does not change. As shown in figure 4.12 GA-based PID controller for parameter variation, the performance character like percentage overshoot and rise time, and settling is almost the same as the original GA-based PID controller response.

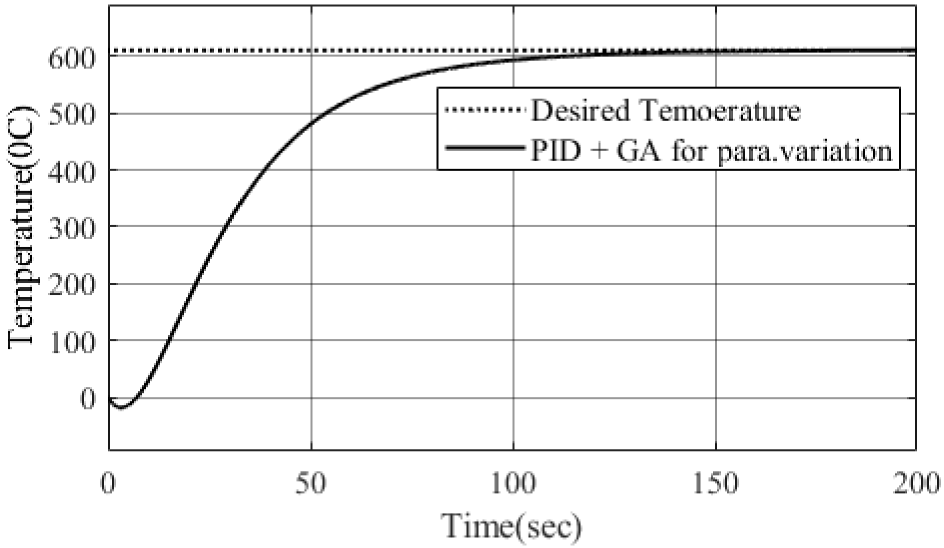


Figure 4.12 PID with GA Tuned controller for parameter variation

4.10.4. The Result of GA Based FOPID Controller for Parameter Variation

The dynamic model of GA based Fractional Order to control the temperature of electric furnace for the tempering glass process will have the following form:

$$G(s) = \frac{as + b}{cs^2 + ds + e} = \frac{-13.89s + 2.778}{25s^2 + 6s + 0.2}$$

Now, to deal with the robustness of PID controller with parameter variation of increasing or decreasing the parameter of the system. From the above transfer function, if parameters a and d decreased by 10%, if parameters b and c increased by 5%, and if parameter e does not change the system response is almost the same as the original response of FOPID with GA tuned. As shown in figure 4.13 GA-based FOPID controller for parameter variation, the time domain performance specifications have very small differences.

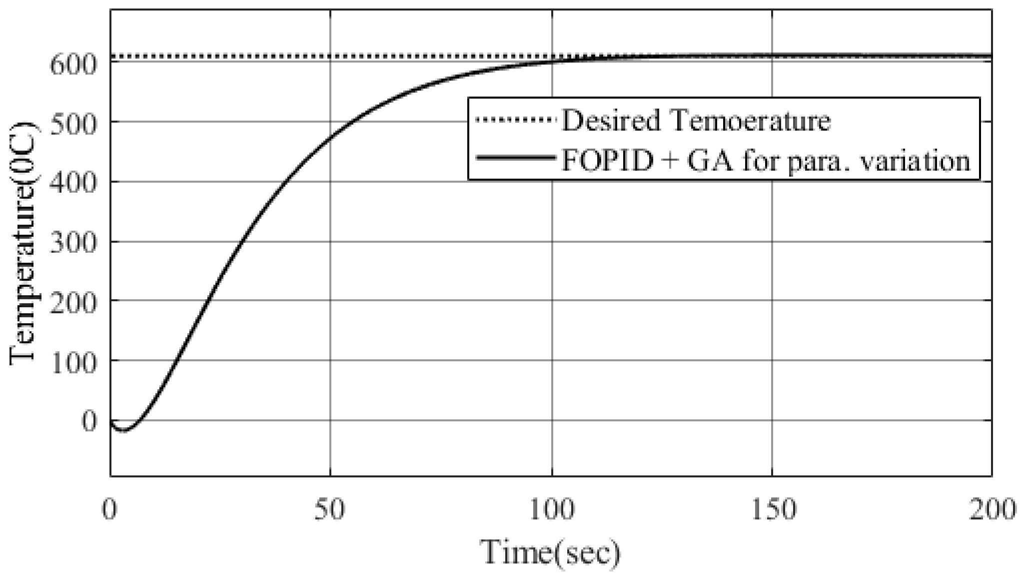


Figure 4.13 GA based FOPID for parameters variations

Table 4.2 Performance specifications for parameter variations of controller

Performance of parameters	FOPID + GA Controller	PID + GA controller	MIT rule with FFPI controller	Lyapunov rule with FFPI controller
Settling time(sec)	128.51 sec	136.3 sec	140.48 sec	140.46 sec
Rise time(sec)	55.21 sec	55.15 sec	51.73 sec	51.68 sec
Overshot (%)	0%	0%	0%	0%
Steady-state error(%)	0%	0%	0%	0%

4.11. The Result of System with Impulse Disturbance

4.11.1. The Result of PID Controller with Impulse Disturbance

The result of the Genetic Algorithm based on PID Controller with Impulse disturbance as shown in figure 4.14 is increasing the overshoot and undershoots, and the system did not track immediately the desired input of the temperature. But finally, the response is decayed to the desired input of the temperature value.

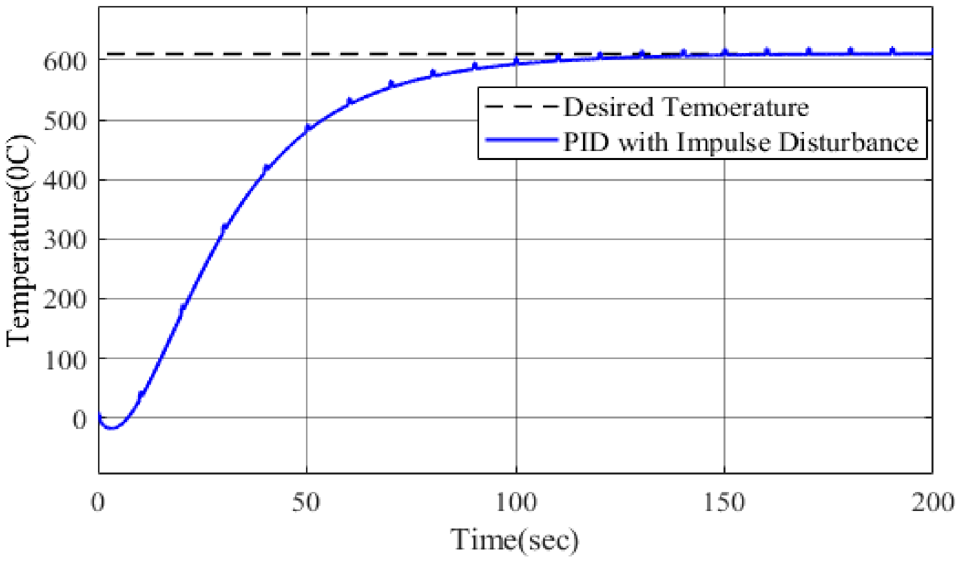


Figure 4.14 The result of GA based PID with impulse disturbance

4.11.2. The Result of Fractional Order Controller with Impulse Disturbance

The result of the GA-based Fractional Order PID Controller with Impulse disturbance as shown in figure 4.15 below is increasing the overshoot and undershoot. This shows the temperature varies between different parts of the furnace. At this time the quality of the product of the glass decreased. The system did not track immediately the desired input of the temperature. But finally, the system response is approached to the desired input of the temperature value.

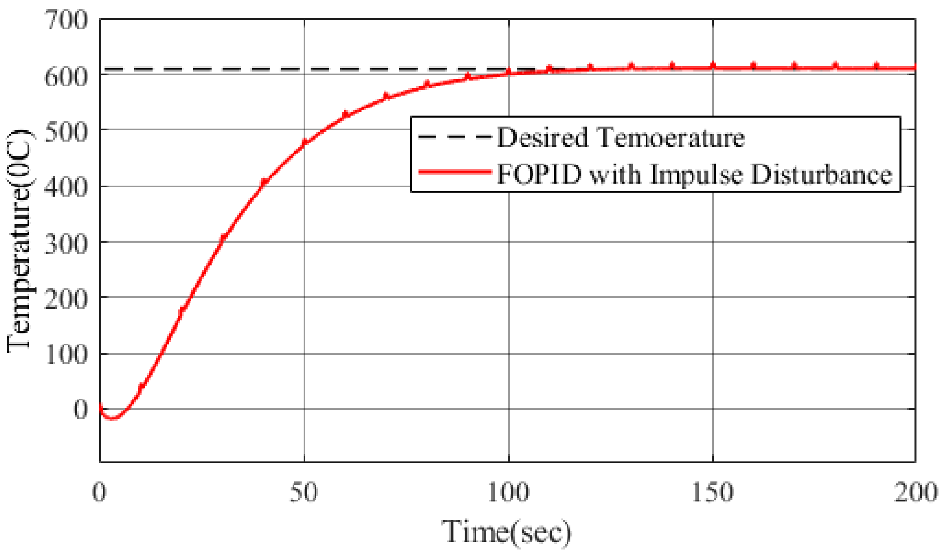


Figure 4.15 The result of GA based FOPID with impulse disturbance

4.11.3. Comparison of GA-FOPID and GA-PID Controller with Impulse Disturbance

The capability of the controllers in disturbance rejection is checked by adding an impulse disturbance signal as shown below in figure 4.16. As shown from the figure GA-based Fractional Order controller has a better disturbance rejection than the GA-based PID controller.

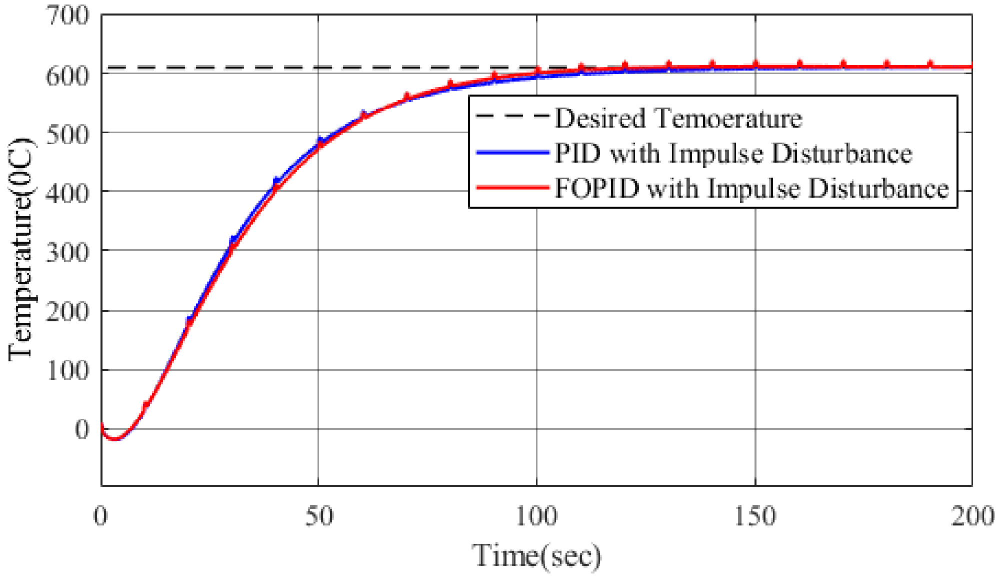


Figure 4.16 The result of GA based FOPID, and PID with impulse disturbance

Table 4.3 Performance of GA based FOPID and PID with impulse disturbance

Performance Specifications	GA based FOPID Controller	GA based PID Controller
Settling time	188.42 sec	189.76 sec
Rise time	56.21 sec	56.3 sec
Overshot (%)	0.64%	0.68%
Steady-state error	6.08 %	6.32%

4.12. The Response of Control Signal

As shown in figure 4.17 below the control signal of the PID controller will have an initial value of 610 and from that, it will come to zero within 124.3 seconds.

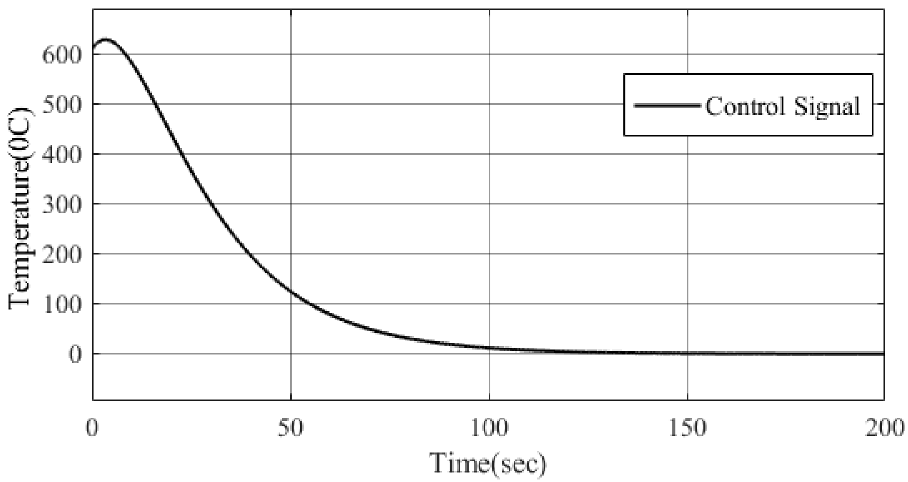


Figure 4.17 The Control signal of the PID controller

4.13. Comparison of the Results of GA Based FOPID and GA Based PID

From the control system theory, the system is said to have better dynamic performance or best response if it has a small rise time, a small peak (overshoot), a small settling time, and a small peak time. Thus, as seen from figure 4.18 shown below, the FOPID with GA Tuned has a smaller time domain specification than PID with GA Tuned. But for this system, the temperature must be between 610^0C and 640^0C. Therefore, the temperature to reach these values takes some minutes. The FOPID with GA tuned has a small settling time than PID with GA Tuned. So that based on this the FOPID with GA Tuned is a better controller than PID with GA Tuned to control the temperature of the furnace for the tempering process of the glass.

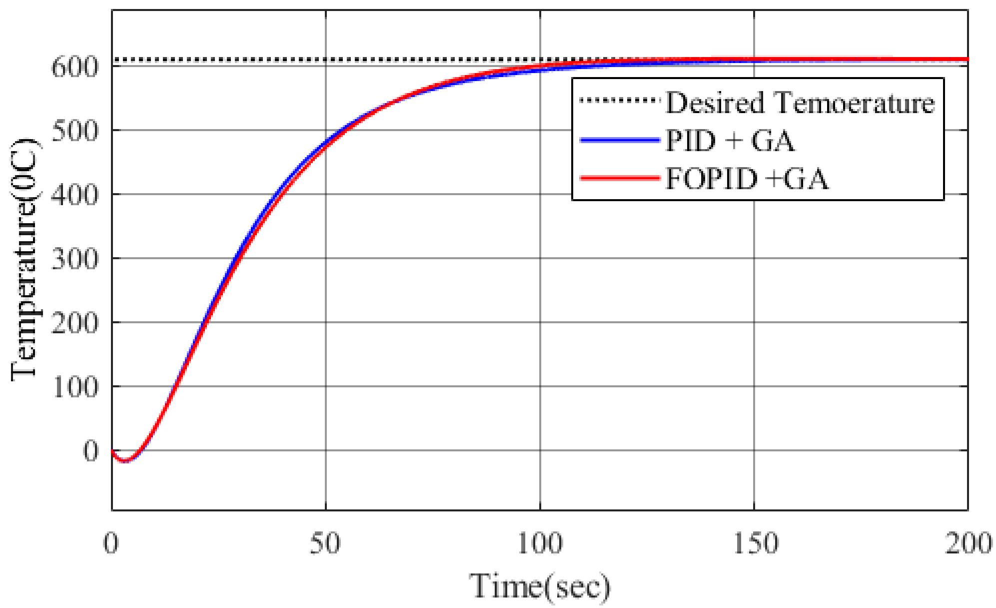


Figure 4.18 Comparison system responses of GA-FOPID and GA-PID

Table 4.4 The Time-domain specifications of FOPID-GA and PID-GA controllers

Performance Specifications	GA based FOPID Controller	GA based PID Controller
Settling time (sec)	128.46 sec	135.7 sec
Rise time (sec)	55.19 sec	55.09 sec
Overshoot (%)	0%	0%
Steady-state error (%)	0%	0%

4.14. Comparison of GA Based FOPID, GA Based PID, and MRAC Controllers

As seen from figure 4.19 shown below, the GA-based FOPID has a better time domain specification than the GA-based PID Controller. The planned GA-based Fractional Order PID controller has the best transient response from the GA-based PID and MRAC controller. So that based on this the FOPID with GA Tuned is the best controller compared with GA-based PID and MRAC (Lyapunov rule with FFPI and MIT rule with FFPI Controllers) to control the temperature of the electric furnace for the tempering process of the glass.

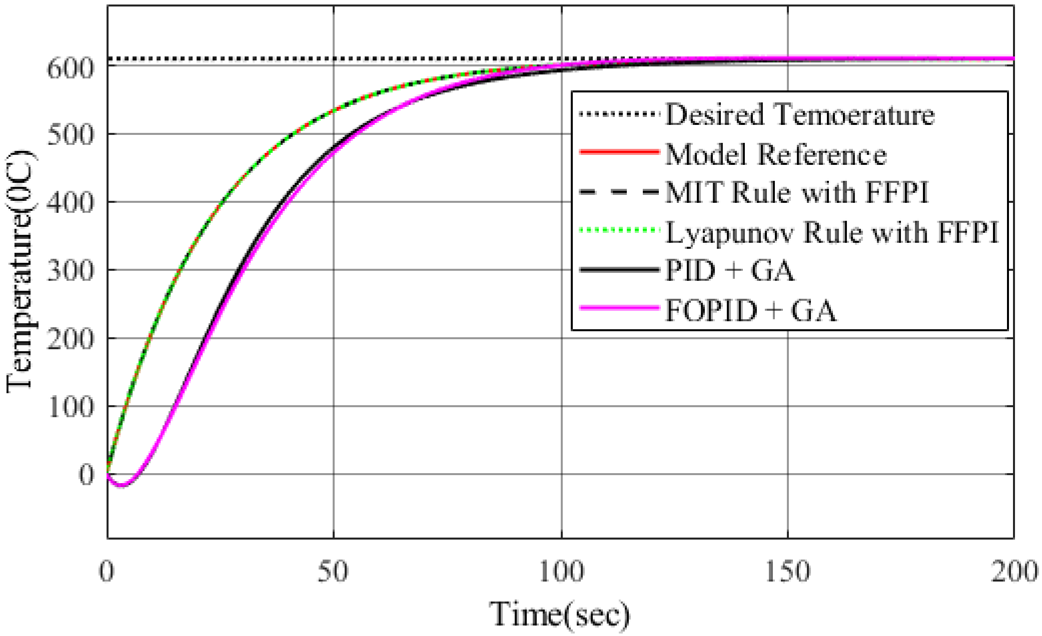


Figure 4.19 Comparison of FOPID-GA, PID-GA, Lyapunov rule, and MIT rule

Table 4.5 Performance of MIT rule, Lyapunov rule, GA based FOPID, and PID

Performance Specifications	GA-FOPID controller	GA-PID controller	MIT rule with FFPI controller	Lyapunov rule with FFPI controller
Settling time (sec)	128.46 sec	135.7 sec	140.42 sec	140.39 sec
Rise time (sec)	55.19 sec	55.09 sec	51.67 sec	51.65 sec
Overshot (%)	0%	0%	0%	0%
Steady-state error(%)	0%	0%	0%	0%

5. CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

In this thesis, the control of the temperature to the electric furnace for the tempering process of glass is presented. The furnace is a nonlinear and time-varying complex system and uses high temperatures. Afterward, appropriate assumptions and estimations are considered in the mathematical calculations, and the electric furnace transfer function is obtained in Laplace transform. The results of manual tuning PID controllers for temperature control of furnace are not acceptable to the higher degree of accuracy condition, because there is a high-temperature difference has occurred. High power is supplied to the furnace to make the temperature values at 610^0C. This takes more seconds to reach the desired value of the temperature.

In this thesis work, different controllers are developed in MATLAB/Simulink model. Model Reference Adaptive Control (MRAC), GA-based PID, and GA-based Fractional Order PID controllers are developed for normal conditions, parameter variations, and disturbance conditions. For the MRAC controller, the MIT rule and the Lyapunov rule are developed for both with FFPI and without FFPI controllers, but time domain specifications have a small difference. Thus, it can be taken there is no maximum variation between the MIT rule and the Lyapunov rule for this system. But the MRAC controller has a large settling time to reach the desired temperature value of the furnace.

The performance specifications of the proposed MRAC, GA-based PID, and GA-based FOPID controllers are tested through simulation results using MATLAB/Simulink. It is observed from MRAC the simulation results, the overshoot is 0%, the rising time 51.66 seconds, the settling time is 140.4 seconds, and steady-state error 0%. From the GA-based PID, the simulation results of performance specifications are: the overshoot is 0%, the rising time 55.09 seconds, the settling time is 135.7 seconds, and steady-state error 0%. From the proposed GA-based FOPID controller, the performance specifications of the system are the simulation results that the overshoot is 0%, rising time 55.19 seconds, the settling time is 128.46 seconds, and steady-state error 0%.

As it is observed from the performance specifications described above, the MRAC response has settled after 140.4 seconds, the GA-based PID controller has settled after 135.7 seconds, and the FOPID-GA has settled after 128.46 seconds. Thus, the GA-based PID controller response is better than the MRAC controller. But the best response used to control the temperature of the furnace for the tempering process of the glass is the Genetic Algorithm with

Fractional Order PID controller response because it has the best settling time to reach the desired temperature value.

5.2 Recommendations

In this thesis work, to control the temperature of the electric furnace for the tempering process of glass is well designed and simulated by using the controller the Genetic Algorithm based on Fractional Order PID with MATLAB/Simulink. The obtained result of the system response is the best as required. Therefore, for further improvement, the hardware of the self-tuning Genetic Algorithm-based Fractional Order PID Controller can be developed to control the temperature of the furnace for the tempering process of glass.

Research fund Acknowledgement

This research project is funded by Adama Science and Technology University under the grant number ASTU/SM-R/167/20, Adama, Ethiopia.

6. REFERENCES

[1] Jonathan Barr, "Understanding the Glass Tempering Process", Germany, February 3, 2015.

[2] Windows, Andersen, "glass making process, and review the tempering process," Stillwater, Minnesota, America, January 27, 2021.

[3] Achenef Sisay, "Fuzzy PID Based Temperature Control of Electric Furnace for Glass Tempering Process," March 2017.

[4] Mark Ford, "How is tempered glass made?" In Scientific American," 2001. [Online].

[5] Mikko Rantala, "Heat Transfer Phenomena in Float Glass Heat Treatment Processes," Tampere University of Technology, vol. 1355, November 2015.

[6] Gilbert Rancoule., "Glass handling in tempering furnaces," July/ August 2011.

[7] Abate Nebil, "Temperature Control of Electric Furnace for Glass Tempering Process using MRAC," 01 December 2020.

[8] Jeroen Kuenen, "Manufacture of glass and glass products," in Guidebook, 2013.

[9] Berkeley, Ajax Electric Company, Improving Process Heating System Performance, S. Edition, Ed., Lawrence Berkeley National Laboratory, California, 2007.

[10] Fradette, Real J., "Understanding PID Temperature Control as Applied to Vacuum Furnace Performance," 1996.

[11] Nazarian, Peyman, "Feedback Linearization based control of a horizontal radiative furnace," 2010.

[12] P. Mazarin, "Fuzzy Logic Controller Implementation for a Tempered Glass Furnace," in In: Proceedings of the international conference on mathematical sciences &computer engineering, Kuala Lumpur, Malaysia, 2012.

[13] Isizoh A. N., Okide S. O., Anazia A.E., and Ogu C., "Temperature Control System Using Fuzzy Logic Technique," International Journal of Advanced Research in Artificial Intelligence, vol. 1, November 3, 2012.

[14] Di Lu, Jian Xin Wang, Jia Feng Li, "The Temperature Control of Electric Furnace Based on PID Genetic," Vols. Advanced Materials Research (Volumes 490-495), pp. 828-834, March 2012.

[15] Ji-bin Chang, De-Biao Wang, and Tai-fu Li, "The Electric Heating Furnace Temperature Control Based on SPID," Chongqing University of Science and Technology, January 2009.

[16] Diwakar T. Korsane1, Vivek Yadav2, Kiran H. Raut3, "PID Tuning Rules for First Order Plus Time Delay System," in International Journal of Innovative Research In Electrical, Electronics, Instrumentation And Control Engineering, vol. 2, Issue 1, January 2014.

[17] K. J. Astrom and T. Haglund, "PID Controllers Theory, Design and Tuning", 2nd ed., Research Triangle Park, NC: Instrum. Soc. Amer., 2006.

[18] J. C. Basilio and S. R. Matos, Design of PI and PID Controllers With Transient Performance Specification, vol. 45, No.4, November 2002.

[19] Ayenom Hafte, "Design of Lyapunov Rule-Based Model Reference Adaptive Control With Decoupler for Boiler-Turbine-Generator," January 2019.

[20] Karl Johan Astrom., "PID Control" control system design," 2002.

[21] Controls, Automatic, "Feedback Control of Linear SISO systems Open-Loop Process," Process Dynamics and Control, 2012.

[22] Wikipedia, the free encyclopedia, "PID controller," https://en.wikipedia.org/wiki/PID, accessed on August 9, 2016. [Online].

[23] Md Mustafa Kamal and Lini Mathew and S.Chatterji, "Speed Control of Brushless DC Motor Using Intelligent Controllers," IEEE, p. 2, December 2018.

[24] R.G.P.V. and Bhopal India, "Speed Control of DC Motor Using Genetic Algorithm Based PID Controller," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3, no. 7, July 2013.

[25] DUSAN PETKOVIC and MIROSLAV RADOVANOVIC, "Using Genetic algorithms for optimization of turning machining process," Journal of Engineering Studies and Research, vol. 19, pp. 580-587, Dec 12, 2013.

[26] S. E. Petlenkov, " Fractional-order Calculus-based Identification and Control of Linear Dynamic Systems," in Faculty of Information Technology Department of Computer Control, 2011.

[27] Yibeletal Kassahun, "Design of Fractional Order PID & Neural Network Controller for Magnetic Levitation Train System," December 2018.

[28] Babatunde A. Ogunnaike and W. Harmon Ray, "process dynamics, modeling, and control", New York: Oxford University Press, 1994.

[29] Al Falahi, Moayad Salim Canlas, Freddie Quiroz Nair, Sarchandran, "International Conference on Computing, Electronics & Communications Engineering (ICCECE)," 2020.

[30] McGraw, "Introduction to Thermodynamics and Heat Transfer", chapter two heat equation and part two transfer, Second ed., Hill 375 Companies, 2008.

[31] YOON and Heenam, Medical Physics, "Dynamic Temperature Measurement", 2012.

[32] Process Dynamics, Operations, and Control, Heated Tank, Lesson 5 spring 2006.

[33] K. J. Astrom and T. Haglund, "Automatic Tuning of PID Controllers”," 1st ed., Research Triangle Park, NC: Amer., Instrum. Soc., 2006.

[34] J. G. Ziegler and N. B. Nichols, "Optimal settings for automatic controllers," vol. 64, ASME, Trans., 2015, p. 759–768.

[35] Babak Rooholahi* and P. Lokender Reddy, "Concept and Application of PID Control and Implementation of Continuous PID Controller in Siemens PLCs," in Indian Journal of Science and Technology, Vol. 8, Issue: 35, DOI: 10.17485/ijst/2015/v8i35/82262, December 2015.

[36] R. Rajesh1, "Optimal tuning of FOPID controller based on PSO algorithm," July 2019.

[37] I. Podlubny, L. Dorcak, I. Kostial, "Advances in Computing and Network Communications - Zenodo," in Proceedings of the 36th IEEE Conference on Decision and Control, San Diego, CA, USA, October 2020.

[38] Ajith.B.Singh and Kaushik S., "Identification and real-time control of a hybrid tank system through virtual instrumentation," IEEE International Conference on Green Computing Communication and Electrical Engineering, March 2014.

[39] J. B. Hoagg and D. S. Bernstein, "Nonminimum-phase zeros - much to do about nothing - classical control - revisited part II," 2007.

[40] S. Skogestad and I. Postlethwaite, "Multivariable feedback control: analysis and design," New York: Wiley, Chichester.

[41] Vidyasagar M., "On undershoot and nonminimum phase zeros," in Automatic control, IEEE, 2010.

[42] K. J. Åström, "Zeros of sampled systems," vol. 20, pp. 31-38, 2004.

7. APPENDIXIES

7.1. APPENDIX A

%MATLAB code for different PID controller tuning system

```
s=tf('s');
TF=exp(-10*s)*13.89/(25*s+1); % model of the furnace to be controlled
H=1/(10*s+1);%transfer function of sensor (feedback), if τs=10 sec
T=25; D=10;
K=13.89;
z=tf('z');
p=pade(TF,1);
pd=c2d(p,1);
hd=c2d(H,1);
%% PID controller of Ziegler-Nichols tuning
KcZN=(1.2/K)*(T/D);
TdZN=0.5*D;
KdN=KCZN*TDZN;
TiZN=2*D;
KiN=KCZN/TIZN;
CZN=KCZN+KIN*(z/(z-1))+KDN*((z-1)/z);
SYSZN=feedback(pd*CZN,hd);
step(SYSZN,'G')
legend('ZN tuning');
hold on;
%% PID controller Cohen-Coon tuning
KcC=(1/K)*(T/D)*[(3*D+16*T)/(12*T)];
TiC=(D*(32+6*(D/T)))/(13+8*(D/T));
Ki=Kc/Ti;
TdC=4*D/(11+2*(D/T));
KdC=KcC*TdC;
CCC=KcC+KiC*(z/(z-1))+KdC*((z-1)/z);
SYSCC=feedback(CCC*pd,hd);
step(SYSCC,'r')
hold on;
```

```
%% PID controller ITAE load setting tuning
kcl=0.2327;
til=15.099;
KIl=kcl/til;
tdl=23.704;
kdl=kcl*tdl;
CITAEL=kcl+KIl*(z/(z-1))+kdl*((z-1)/z);
SYSITAEl=feedback(CITAEL*pd,hd);
step(SYSITAEl,'B')
tuned tool box');
hold on;
%% continuous time PID controller ITAE set point tuning
kcs=0.1514;
tis=27.46;
KIs=kcs/tis;
tds=18.038;
kds=kcs*tds;
CITAEs=kcs+KIs*(z/(z-1))+kds*((z-1)/z);
SYSITAEs=feedback(CITAEs*pd,hd);
step(SYSITAEs,'m')point');
grid on;
```

7.2. APPENDIX B

MATLAB Simulink Model of Temperature Control of Electric Furnace

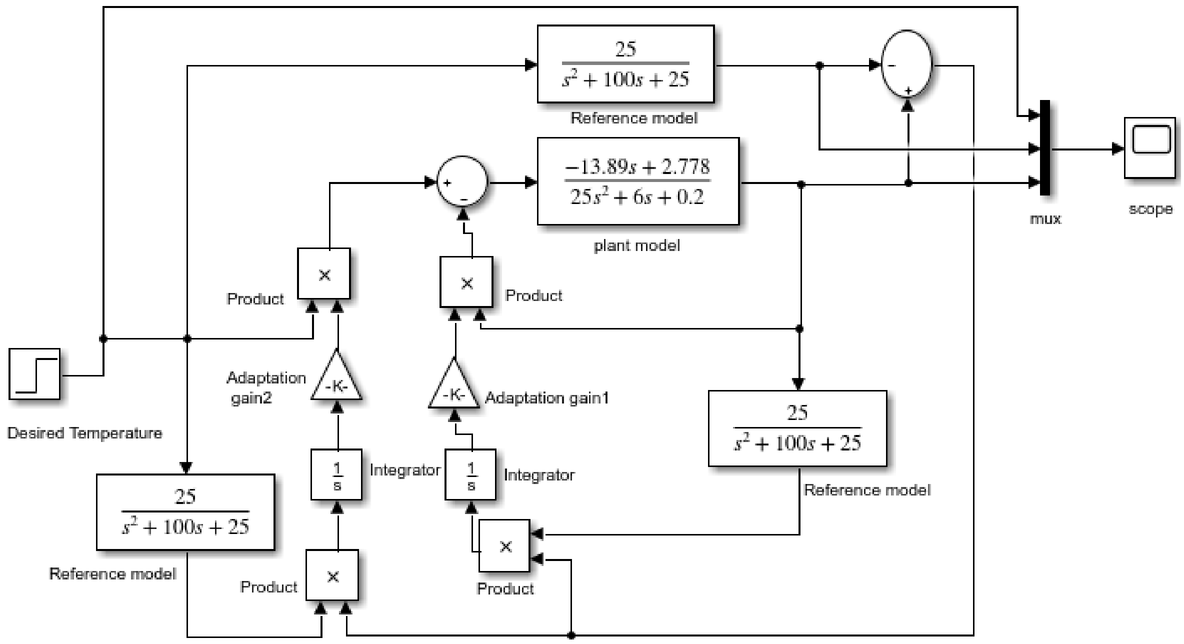


Figure A1: MATLAB Simulink model of MIT rule without FFPI

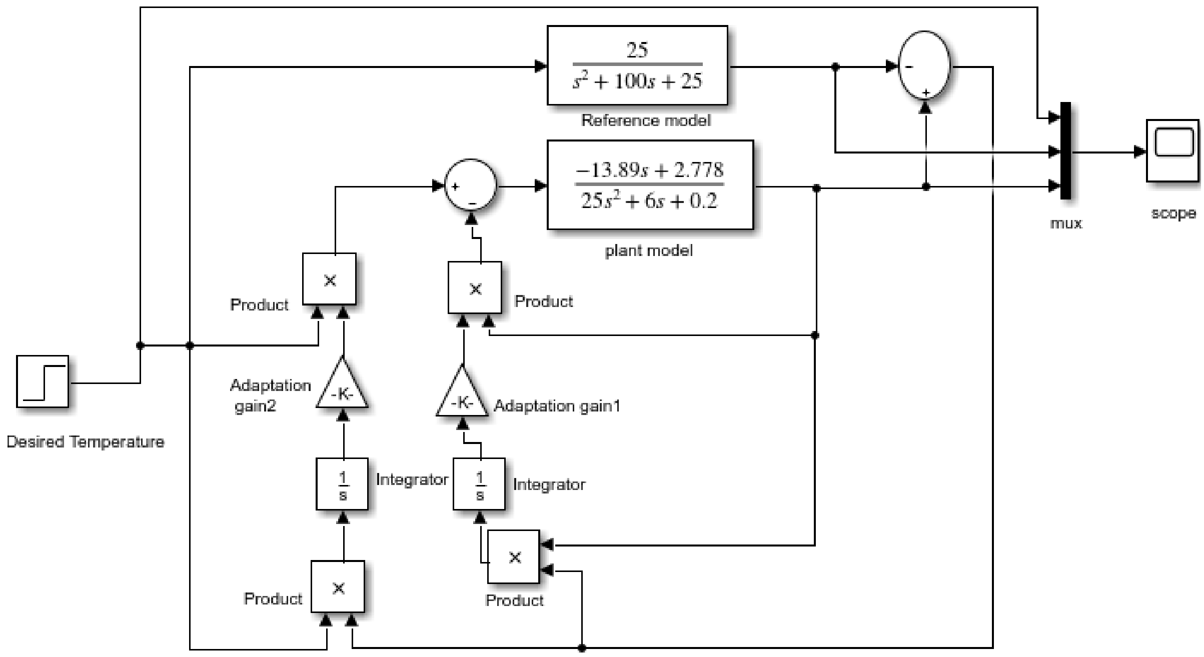


Figure A2: MATLAB Simulink model of Lyapunov rule without FFPI