Addis Ababa University

Addis Ababa Institute of Technology

Fuzzy PID Based Temperature Control of Electric Furnace for Glass Tempering Process

A Thesis Submitted to the School of Graduate Studies, Addis Ababa Institute of Technology, Addis Ababa University

In Partial Fulfillment of the Requirements for the Degree of Master of Science in Electrical Engineering (Control Engineering)

By

Sisay Achenef

Advisor: Dr. Dereje Shiferaw

March, 2017

Addis Ababa University

Addis Ababa Institute of Technology

Fuzzy PID Based Temperature Control of Electric Furnace for Glass Tempering Process

By

Sisay Achenef

Approved by Board of Examiners

Chairman, Department of Graduate committee Signature

Dr. Dereje Shiferaw

Advisor Signature

Dr.Mengesha Mamo

Internal Examiner Signature

Dr. Getachew Alemu

External Examiner Signature

DECLARATION

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

Sisay Achenef

Name

Signature

Addis Ababa, Ethiopia

Place

March , 2017

Date of submission

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

Dr. Dereje Shiferaw

Advisor Name

Signature

ACKNOWLEDGMENTS

First of all, I am highly grateful to my advisor, Dr. Dereje Shiferaw for his encouragement, insight guidance, and professional expertise. My special thanks go to all instructors who thought me during my graduate study. I take this opportunity to thank Deber Markos University for giving me the chance to continue my graduate study in Electrical Engineering. No words of thanks are enough for my family for their unwavering support and love offered throughout the thesis work and my graduate study. Last but by no means least; I am grateful to all colleagues for the support they give.

TABLE OF CONTENTS

Contents **page**

LIST OF ABBREVIATIONS

AAiT	Addis Ababa Institute of Technology
AAU	Addis Ababa University
AMIGO	Approximate M-constrained integral gain Optimization
CE	Error rate/Change Error
E	Error
FLC	Fuzzy Logic Controller
FPID	Fuzzy Proportional Integral derivative
ITAE	Integral Time Absolut Error
NB	Negative Big
NM	Negative Medium
NS	Negative Small
PB	Positive Big
PID	Proportional Integral derivative
PM	Positive Medium
PS	Positive Small
Z	Zero

LIST OF TABLES

LIST OF FIGURES

Figure **Page**

LIST OF APPENDIXES

ABSTRACT

Tempered glass is a safety glass processed by a controlled thermal or chemical treatment to increase its strength compared with the ordinary (or annealed) glass. In thermal tempering process, the glass is heated to a temperature just below the softening point and is then quenched rapidly in air. Thus, in thermal tempering manufacturing process temperature control of tempering furnace is the main task. Our country Ethiopia has some glass manufacturers hence we can use thermal tempering process to manufacture tempered glass.

This thesis deals with the temperature control of electric furnace. The temperature of electric furnace is controlled to a desired value. In tempered glass furnace industry, the widely used controller for temperature control is conventional PI and PID controllers. Since electric furnaces are nonlinear and time varying processes, using PID controllers is difficult to meet the control requirement. Because they are not robust, haven't good dynamic response and have large overshoot. Hence in this thesis, fuzzy logic controller which is fault tolerant and nonlinear controller is used to improve the PID and achieve steady state and transient requirements.

Therefore, in this thesis a model of glass tempering furnace is prepared using MATLAB/SIMULINK and then fuzzy PID controller has been designed and simulated and its performance has been compared with conventional PID controllers. The performance of the controllers is compared in terms of disturbance rejection, transient and steady sate performance.

It is observed from the simulation results that the average overshoot is 2.129%, rising time is 44 seconds and the settling time is 170s seconds with the proposed fuzzy PID controller while overshoot is 16.129%, rise time is 45 second and settling time is 290 seconds with PID controller. Moreover, the robustness and disturbance rejection of the controllers is checked by parameter variation like time constant, delay time & DC gain and giving disturbance signal after settling time respectively. It is further observed that the proposed controller has better disturbance rejection and more robust.

Key words: Glass tempering furnace temperature control, Fuzzy PID controller, PID controller.

CHAPTER ONE

INTRODUCTION

1.1 Background

Tempered glass is a safety glass processed by a controlled thermal or chemical treatment to increase its strength compared with the ordinary (or annealed) glass. In general, tempered glass is about four times stronger than the normal glass. This property of tempered glass causes the broken glass to crumble into small granular chunks rather than splintering into jagged shards. As result of it strength, tempered glass is used in environments where safety is an important issue. Applications of tempered glass include the side and the rear windows of vehicles, entrance doors, shower and tub enclosures, patio furniture, microwave furnaces and skylights, etc. [1][2][3][4][5].

Tempered glass was first patented in June/July1874 by the Frenchman Francois Royer de la Bastie but not developed until the early 1900s and the first patent for a process to produce tempered glass is said to be held by Rudolf Seiden, an Austrian chemist who immigrated to the United States in 1935. Prior to this time the only documented use of deliberately tempered glass was the production of Prince Rupert Drops which were novelty items [1][2].

In the 1940's, Henry Ford started using tempered glass for the side and rear windows of his automobiles and its use in vehicles became common place in the 1950's. Architectural tempered glass started to become popular in the 1960's as the benefits of the stronger, safer glass became more widely known.

Tempered glasses can be made from annealed glass via a thermal tempering process. In the thermal tempering process we use electric furnace for heating the glass. Another approach for making tempered glass is chemical tempering. The thermal tempering process is a very simple process, just heat the glass up to 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 [2][4].

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. An abrasive such as sandpaper takes sharp edges off the glass, which is subsequently washed [1][3].

Next, the glass begins a heat treatment process in which it travels through a tempering furnace, either in a batch or continuous feed. The furnace heats the glass to a temperature of tempering range between 610 degrees Celsius to 640 degrees Celsius. Then the glass undergoes a high-pressure cooling procedure called quenching. During this process, which lasts just seconds, high-pressure air blasts the surface of the glass from an array of nozzles in varying positions. Quenching cools the outer surfaces of the glass much more quickly than the center. As the center of the glass cools, it tries to pull back from the outer surfaces. As a result, the center remains in tension, and the outer surfaces go into compression, which gives tempered glass its strength [1][2][3][5].

Glass in tension breaks about five times more easily than it does in compression. Annealed glass will break at 6,000 pounds per square inch. Tempered glass, according to Germany federal specifications, must have a surface compression of 10,000 per square inch or more; it generally breaks at approximately 24,000 pounds per square inch [2].

Electric furnace has a large hysteresis, nonlinearity and time variability, to such an object, only taking fuzzy control or PID control can't achieve better result. Lots of theoretical research and engineering cases has also confirmed this point. Recent implementations of fuzzy logic controllers in temperature control of tempered glass furnace can be found in [2] and Understanding PID Temperature Control as Applied to Vacuum Furnace Performance is given in [3]. Thus Fuzzy PID control has fuzzy control's characteristics, such as robustness, good dynamic response, short rising time, small overshoot, and it also has PID controller's advantages, the quality of dynamic tracking and stable high accuracy.

1.2 Statement of the problem

In thermal tempering glass production process the task temperature control is the main part and it has an important influence on the quality of the products. Therefore, to produce tempered glass with a high quality, temperature in the glass tempering furnace must be controlled very well. Inadequate temperature control in the tempering furnace will lead to some product defects. Thus, in order to successfully control the temperature in the tempered furnace, the glass must be heated to a given temperature (tampering temperature) quickly. In addition, the temperature difference between the various parts of the glass surface must be very small, i.e., the glass surface must be heated evenly.

In the tempered glass furnace industry, the widely used controller for temperature control is the conventional PI and PID controllers which are not robust haven't good dynamic response, long rise time and large overshoot are used to control the temperature. Also fuzzy logic controller which introduces steady state error in the system is used. In chemical tempering method, various chemicals exchange ions on the surface of the glass in order to create compression [1][2]. But because of its higher cost when compared to the tempering method using furnaces and quenching, chemical tempering is not widely used.

Therefore, in this thesis, fuzzy PID controller is proposed. The fuzzy PID controller will avoid the above problems because it mixes advantage of fuzzy and conventional PID controllers.

1.3 Relevance and scope of the Thesis

To have good understanding the use of fuzzy PID controllers in controlling nonlinear activities like electric furnace. And to Study and design temperature controller for glass tempering and propose implementable design for the future in order to produce cost effective and good quality glass.

1.4 Objective of the thesis

1.4.1 General objective

The main objective of this Msc thesis work is study, design, and simulate self-tuning scheme fuzzy PID temperature controller of electric furnace which is used for glass tempering process.

1.4.2 Specific objective

The specific objective of this thesis work is:

- ✓ To study different temperature control mechanisms of glass tempering furnace.
- ✓ To model electric furnace for temperature control.
- ✓ To design temperature control system for glass tempering process.
- ✓ To simulate the temperature control system using MATLAB.
- ✓ To analyze and compare the performance of fuzzy PID controller with conventional PID controller for glass tempering.

1.5 Outline of the thesis

This thesis is organized into five chapters. The first chapter presents the overview of temperature control mechanisms of glass tempering furnace, statement of the problem, relevance and objectives of the study.

In chapter two, different literatures, related to glass tempering systems and their temperature control system, are reviewed. Besides, the basic of fuzzy and PID controllers are also reviewed.

System modeling, designing and analysis are presented in chapter three. Simulation and experimental results are presented and discussed in chapter four. Finally, chapter five presents conclusions and recommendations.

CHAPTER TWO

2 LITERATURE REVIEW

This chapter will review the basics of thermal treated glass tempering process focusing on the control systems. Different temperature controllers of electric furnace will also be reviewed.

2.1 Introduction glass Tempering Process

2.1.1 History of Tempered Glass

Deliberately tempered glass was first patented in June/July1874 by the Frenchman Francois Royer de la Bastie but not developed until the early 1900s and the first patent for a process to produce tempered glass is said to be held by Rudolf Seiden, an Austrian chemist who immigrated to the United States in 1935. Prior to this time the only documented use of deliberately tempered glass was the production of Prince Rupert Drops which were novelty items [1][2].

Back in the early 1600s a German in Mecklenburg discovered that if he dropped molten glass into a bucket of cold water he produced tadpole shaped pieces of glass that were unbreakable until the tail was fractured. When the tail was broken the tadpole of glass then disappeared, as if by magic, in a cloud of dust [1][2].

How these were made was kept a secret and they were sold as novelties throughout Europe.
In 1660 Prince Rupert, Duke of Bavaria, gave some to King Charles II, who in turn gave them to the Royal Society for further investigation and they discovered that the 'drops' were in fact molten glass that had been stressed by dropping into cold water. The resultant temperature differences present in the glass as it became solid produced very high compressive stress in the surface and high tensile stress in the core of the glass. When the tail was fractured a shock wave that is followed by a fracture front, travelled through the glass breaking it completely into extremely small particles [1][2].

Tempered glass has greater mechanical strength and thermal resistance than heat strengthened glass. It is about four to five times more impact resistant than regular non-tempered glass. In the event of breakage, tempered glass will break with a high energy release of small fine round pieces [2]. In the 1940's, Henry Ford started using tempered glass for the side and rear windows of his automobiles and its use in vehicles became common place in the 1950's. Architectural tempered glass started to become popular in the 1960's as the benefits of the stronger, safer glass became more widely known.

2.1.2 Tempering process

Tempered glasses can be made from annealed glass via a thermal tempering process. In the thermal tempering process we use electric furnace for heating the glass. Another approach for making tempered glass is chemical tempering. The thermal tempering process is a very simple process, just heat the glass up to 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 [1][2][3][4][5].

Whether tempering large architectural panels, automotive glasses, or small highly stressed industrial glasses, the basic tempering process is the same – heat the glass to a temperature above its Transition Point, where it becomes plastic, keep it optically flat (or form it to a shape if required) then cool it at a controlled rate to a temperature below the Strain Point so there is a temperature difference between the surfaces and the central plane of the glass as it becomes solid again. The differential contractions between the surface and the central plane will introduce the final stress into the glass as it cools down to ambient temperature [2].

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. An abrasive such as sandpaper takes sharp edges off the glass, which is subsequently washed [1].

Next, the glass begins a heat treatment process in which it travels through a tempering furnace, either in a batch or continuous feed. The furnace heats the glass to a temperature of tempering range between 610 degrees Celsius to 640 degrees Celsius. The glass then undergoes a high-pressure cooling procedure called quenching. During this process, which lasts just seconds, high-pressure air blasts the surface of the glass from an array of nozzles in varying positions. Quenching cools the outer surfaces of the glass much more quickly than the center. As the center of the glass cools, it tries to pull back from the outer surfaces. As a result, the center remains in tension, and the outer surfaces go into compression, which gives tempered glass its strength [1][3][6].

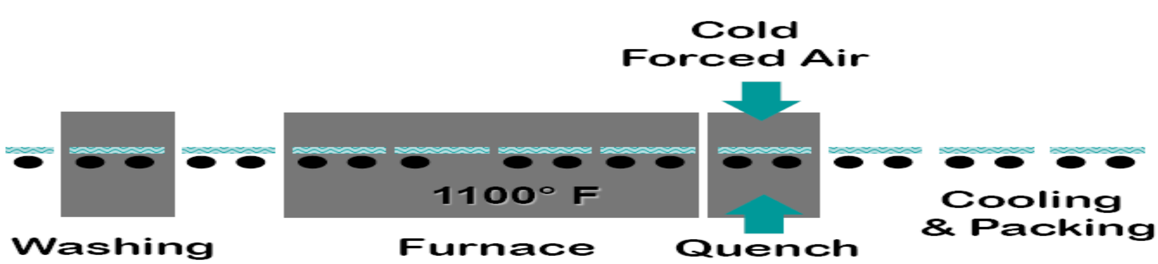


Figure 2. 1 Block diagram of thermal glass tempering process

2.1.3 Glass properties

Clear glass is not completely transparent, a 6mm-thick piece of clear float glass will capture around 13-percent of light within the visible spectrum, allowing 87-percent of the visible light to pass through it. However, as the wave-length of light moves away from the visible range the transmission changes, and for many frequencies glass is quite opaque [2].

Glass is relatively transparent to short wave infra-red but opaque to long-wave infra-red. Raw float glass is very strong when in compression – it can easily withstand 1034MPa (150,000psi). In other words, it is extremely difficult (almost impossible) to break glass when it is in compression.

Theoretically, glass with a perfect surface (no micro-flaws) can also withstand about 1034MPa (150,000 psi) tensile stress. However, in practice, glass is relatively weak when in tension and can only take 25 - 34MPa (4000 to 5000psi) of tensile stress before it fails due to micro-flaws in the surface migrating into and through the glass.

When glass is tempered, all the outer surfaces, including the edges of the glass, are put into compression. The inner portion of the glass will naturally go into tension, so that all the forces are balanced, but because the 'surfaces' of the tension layer are inside the glass it should not have any micro-cracks, unless there is an impurity or air-bubble present, and therefore the tension layer should not fail. This makes tempered glass very difficult to break compared to raw glass and it is generally considered to be 4 to 5 times stronger than raw glass [2].

As the temperature of glass is increased above the Strain Point and past the Transition Point the glass starts to soften i.e. the viscosity of the glass reduces to a level where stresses can start to relieve and the glass enters a plastic state [2][3].

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 it's original shape after the force is removed.

The viscosity of the glass reduces as it is heated up and, above the Transition Point. It can overcome the problem that is seen if the glass was cooled before it reached a high enough temperature. The higher the temperature the more the molecules vibrate and the less rigid the glass becomes. There are defined points for the stiffness of a material, and figure 2.2 below shows the viscosity curve for normal soda-lime glass versus temperature [2].

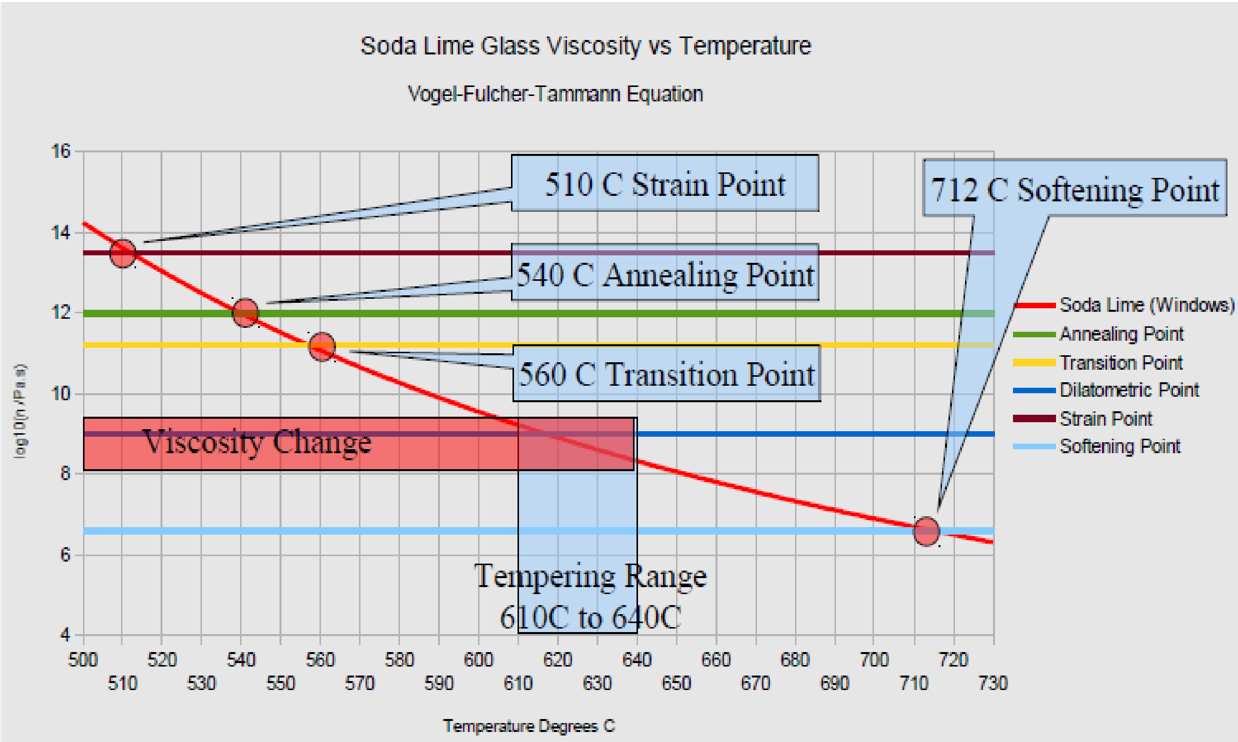


Figure 2. 2 Viscosity of Glass from 500°c to 730°c

If the glass is heated to above the Transition Point (defined as the point where any stress in the glass will quickly dissipate) it is found that the glass can be quenched without it breaking. This is because the soft surfaces (they are like plasticine) can be stretched about the central mass of the glass as long as their temperature is above the Transition Point. Although the surfaces are cooler than the center, and therefore should be shorter, they are in fact at the same length as the hotter center plane of the glass [2].

When glass is heated right up to it's Littleton Softening Point a block of glass will distort under its own weight and the temperature required is approximately 712°c [2].

As outlined above for tempering process, the glass needs to be exiting the furnace at well above the Transition Point but not so hot as to cause it to be too soft and therefore easily distorted. This requirement defines the working range of temperature for tempering architectural glass and is typically around 610 to 640°c depending on glass thickness and it lies across the dilatometric Softening Point.

The glass only needs to go to a higher temperature to make it softer if it is required to be formed, for example automotive side and rear windows or curved architectural glasses. In these cases the exit temperature of the glass is in the region 640^{0}c to 660^{0}C [2].

However, the following has to be achieved in order to obtain good quality tempered glass [2]:

- Very uniform temperature in the glass as it exits the furnace and enters the quench.
- Temperature well above the Transition Point (567^{O}C)
- Temperature well below the Softening Point (710^{O}C)
- Keep the glass optically flat whilst soft (or form it if required)
- Transfer it to a quench without losing too much temperature
- Uniformly cool it at a controlled rate to the Transition Point (567^{O}C)
- Maintain the cooling rate until well below the Strain Point (510^{O}C) Cool it down to a handling temperature

2.2 Resistance Heating and Melting furnace

Resistance heating is the simplest and oldest electric-based method of heating and melting metals and nonmetals. Efficiency can reach close to 100% and temperatures can exceed 3,600ºF. With its controllability, and rapid heat-up qualities, resistance heating is used in many applications from melting metals to heating food products. Resistance heating can be used for both high-temperature and low-temperature applications. There are two basic types of this technology: direct and indirect resistance heating [7].

Resistance heating applications are precisely controlled, easily automated, and have low maintenance. Because resistance heating is used for so many different types of applications, there are a wide variety of fuel-based processes heating systems, as well as steam-based systems, which perform the same operations. In many cases, resistance heating is chosen because of its simplicity and efficiency [7].

2.2.1 Direct resistance heating furnace

Direct resistance (also known as conduction heating), an electric current flows through a material and heats it directly. This is an example of the Joule Law or effect5 at work. Typically, metal is clamped to electrodes in the walls of the furnace and charged with electric current. Electric resistance within the load generates heat, which heats or melts the metal. The temperature is controlled by adjusting the current, which can be either alternating current or direct current. The material to be heated must conduct at least a portion of the electric current for direct resistance to work. Metals with low conductivity, such as steel, create more resistance and more heat, which makes the process more efficient. Direct resistance heating is used primarily for heat treating, forging, extruding, wire making, seam welding, glass heating, and other applications. Direct resistance heating is often used to raise the temperature of steel pieces prior to forging, rolling, or drawing applications [7].

2.2.2 Indirect resistance heating furnace

Indirect resistance heating, a heating element transfers heat to the material by radiation, convection and/or conduction. The element is made of a high-resistance material such as graphite, silicon carbide, or nickel chrome. Heating is usually done in a furnace, with a lining and interior that varies depending on the target material. Typical furnace linings are ceramic, brick, and fiber batting, while furnace interiors can be air, inert gas, or a vacuum.

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. Numerous other types of resistance heating equipment are used throughout industry, including strip heaters, cartridge heaters, and tubular heaters[7].

2.3 Temperature control systems of resistance electric furnace

To produce the required quality of tempered glass the temperature of electric furnace should be controlled. There is different controller's used in glass tempering industry. PI, PID, Fuzzy, Fuzzy-PID, feedback linearization and sliding mode controllers are used.

2.3.1 PID Controller

The PID controller is the most common form of feedback. It was an essential element of early governors and it became the standard tool when process control emerged in the 1940s [8-10] [13][14]. In process control today, more than 95% of the control loops are of PID type, most loops are actually PI control. PID controllers are today found in all areas where control is used. The controllers come in many different forms. There are standalone systems in boxes for one or a few loops, which are manufactured by the hundred thousands yearly. PID control is an important ingredient of a distributed control system. The controllers are also embedded in many special purpose control systems. PID control is often combined with logic, sequential functions, selectors, and simple function blocks to build the complicated automation systems used for energy production, transportation, and manufacturing. Many sophisticated control strategies, such as model predictive control, are also organized hierarchically. PID control is used at the lowest level; the multivariable controller gives the set points to the controllers at the lower level [16].

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 [8].

2.3.1.1 Algorithm of PID controller

A PID controller continuously calculates an *error value* as the difference between a desired set point and a measured process variable. The controller attempts to minimize the error over time by adjustment of a control variable, such as the position of a control valve, a damper, or the power supplied to a heating element, to a new value determined by a weighted sum [8] [9] [15].

$$U(t) = K_p e(t) + K_i \int_0^{\tau} e(\tau)d\tau + K_d \frac{de(t)}{dt} \text{ --2.1}$$

Where K_p , K_i and K_d all non-negative, denote the coefficients for the proportional, integral and derivative terms, respectively (sometimes denoted *P, I,* and *D*).

- *Proportional term:* accounts for present values of the error. Because a non-zero error is required to drive it, a proportional controller generally operates with a so-called steady-state error. Steady-state error (SSE) is proportional to the process gain and inversely proportional to proportional gain. SSE may be mitigated by adding a compensating bias term to the set point or output, or corrected dynamically by adding an integral term [15].

  The proportion gain is given by

$$P_{out} = K_p \text{---2.2}$$

- *Integral term I* accounts for past values of the error. The contribution from the integral term is proportional to both the magnitude of the error and the duration of the error. The integral in a PID controller is the sum of the instantaneous error over time and gives the accumulated offset that should have been corrected previously. The accumulated error is then multiplied by the integral gain and added to the controller output.

  The integral term is given by:

$$I_{out} = K_i \int_0^{\tau} e(\tau)d\tau \text{--2.3}$$

  The integral term accelerates the movement of the process towards set point and eliminates the residual steady-state error that occurs with a pure proportional controller. However, since the integral term responds to accumulated errors from the past, it can cause the present value to overshoot the set point value [15].

- Derivative term accounts for possible future values of the error, based on its current rate of change. The derivative of the process error is calculated by determining the slope of the error over time and multiplying this rate of change by the derivative gain. The magnitude of the contribution of the derivative term to the overall control action is termed the derivative gain, Kd [15].

The derivative term is given by:

$$D_{out} = K_d \frac{\mathrm{de(t)}}{dt} \quad \text{--} 2.4$$

Derivative action predicts system behavior and thus improves settling time and stability of the system. An ideal derivative is not causal, so that implementations of PID controllers include an additional low pass filtering for the derivative term, to limit the high Temperature gain and noise [15].

As a PID controller relies only on the measured process variable, not on knowledge of the underlying process, it is broadly applicable. By tuning the three parameters of the model, a PID controller can deal with specific process requirements. The response of the controller can be described in terms of its responsiveness to an error, the degree to which the system overshoot set point, and the degree of any system oscillation. The use of the PID algorithm does not guarantee optimal control of the system or even its stability [8].

Some applications may require using only one or two terms to provide the appropriate system control. This is achieved by setting the other parameters to zero. A PID controller will be called a PI, PD, P or I controller in the absence of the respective control actions. PI controllers are fairly common, since derivative action is sensitive to measurement noise, whereas the absence of an integral term may prevent the system from reaching its target value [14][15].

The PID controller algorithm that operates as a position algorithm is shown in the following Figure [15].

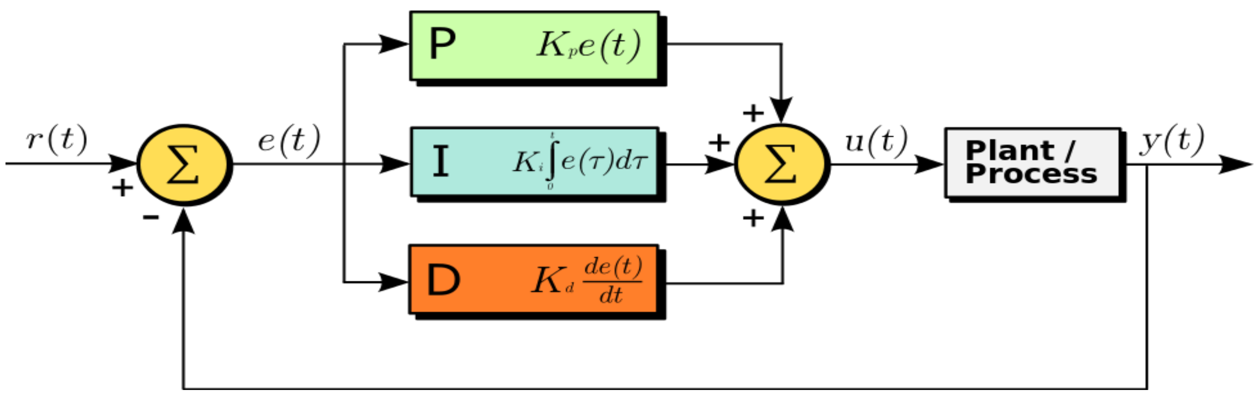


Figure 2. 3 General block diagram of PID controller

Where, $u(t)$ and $e(t)$ denotes the control and error signals of the system. K_p, K_i *and* K_d is the proportion ,integral and derivative gain values. The corresponding PID controller transfer function Gc (s) is given as

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \text{------------------------} 2.5$$

2.3.1.2 Limitation of PID controller

While PID controllers are applicable 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 [15]. The fundamental difficulty with PID control is that it is a feedback control system, with constant parameters, and no direct knowledge of the process, and thus overall performance is reactive and a compromise.

PID controllers, when used alone, can give poor performance when the PID loop gains must be reduced so that the control system does not overshoot, oscillate or hunt about the control set point value. They also have difficulties in the presence of non-linearity's do not react to changing process behavior (say, the process changes after it has warmed up), and have lag in responding to large disturbances [15].

The most significant improvement is to incorporate feed-forward control with knowledge about the system, and using the PID only to control error. Alternatively, PIDs can be modified in more minor ways, such as by changing the parameters (either gain scheduling in different use cases or adaptively modifying them based on performance), improving measurement (higher sampling rate, precision, and accuracy, and low-pass filtering if necessary), or cascading multiple PID controllers [15].

2.3.2 Fuzzy logic controller

Fuzzy logic is a branch of artificial intelligence that deals with reasoning algorithms used to emulate human thinking and decision making in machines. These algorithms are used in applications where process data cannot be represented in binary form. For example, the statements “the air feels cool” and “he is young” are not discrete statements. Fuzzy logic interprets vague statements like these so that they make logical sense. In the case of the cool air, a PLC with fuzzy logic capabilities would interpret both the level of coolness and its relationship to warmth to ascertain that “cool” means somewhere between hot and cold. In straight binary logic, hot would be one discrete value (e.g., logic 1) and cold would be the other (e.g., logic 0), leaving no value to represent a cool temperature [17].

In contrast to binary logic, fuzzy logic can be thought of as gray logic, which creates a way to express in-between data values. Fuzzy logic associates a grade, or level, with a data range, giving it a value of 1 at its maximum and 0 at its minimum [17-21].

Fuzzy logic requires knowledge in order to reason. This knowledge is provided by a person who knows the process or machine (the expert), is stored in the fuzzy system. For example, if the temperature rises in a temperature regulated batch system, the expert may say that the steam valve needs to be turned clockwise a "little bit." A fuzzy system may interpret this expression as a 10-degree clockwise rotation that closes the current valve opening by 5%. As the name implies, a description such as a "little bit" is a fuzzy description, meaning that it does not have a definite value [17].

Fuzzy logic has existed since the ancient times, when Aristotle developed the law of the excluded middle. In this law, Aristotle pointed out that the middle ground is lost in the art of logical reasoning; statements are either true or false. When PLCs were developed, their discrete logic was based on the ancient reasoning techniques. Thus, inputs and outputs could belong to only one set (i.e., ON or OFF); all other values were excluded. Fuzzy logic breaks the law of the excluded middle in PLCs by allowing elements to belong to more than just one set.

Around the 1920s, independent of Bertrand Russell, a Polish logician named Jan Lukasiewicz started working on multivalued logic, which created fractional binary values between logic 1 and logic 0. In a 1937 article in Philosophy of Science, Max Black, a quantum philosopher, applied this multivalued logic to lists (or sets) and drew the first set of fuzzy curves, calling them vague sets. Twenty-eight years later, Dr. Lofti Zadeh, the Electrical Engineering Department Chair at the University of California at Berkeley, published a landmark paper entitled "Fuzzy Sets," which gave the name to the field of fuzzy logic. In this paper, Dr. Zadeh applied Lukasiewicz's logic to all objects in a set and worked out a complete algebra for fuzzy sets. Due to this groundbreaking work, Dr. Zadeh is considered to be the father of modern fuzzy logic [17] [18] [19].

Around 1975, Ebrahim Mamdani and S. Assilian of the Queen Mary College of the University of London (England) published a paper entitled "An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller," where the feasibility of fuzzy logic control was prfurnace by applying fuzzy control to a steam engine. Since then, the term fuzzy logic has come to mean mathematical or computational reasoning that utilizes fuzzy sets.

2.3.2.1 Fuzzy Sets

Define a universe of discourse, X, as a collection of objects all having the same characteristics. The individual elements in the universe X will be denoted as x. In classical, or crisp, sets the transition for an element in the universe between membership and non-membership in a given set is abrupt and well-defined (said to be "crisp"). For an element in a universe that contains fuzzy sets, this transition can be gradual. This transition among various degrees of membership can be lead to the fact that the boundaries of the fuzzy sets are vague and ambiguous [17] [19]. Hence, membership of an element from the universe in this set is measured by a function that attempts to describe vagueness and ambiguity. Elements of a fuzzy set are mapped to a universe of membership values (degree to which a quality is possessed) using a function-theoretic form in the range of [0, 1]. If an element in the universe, say x, is a member of fuzzy set A, then this mapping is given by μ(x) ∈ [0, 1][17].

Fuzzy Set Operations

Define three fuzzy sets A, B, and C on the universe X. For a given element x of the universe, the following function-theoretic operations for the set-theoretic operations of union, intersection, and complement are defined for A, B, and C on X [19].

Union

$$\mu_{A\cup B}(x) = \mu_A(x) \vee \mu_B(x) = \mathrm{Max}(\mu_A(x), \mu_B(x)) \text{--} 2.6$$

Intersection

$$\mu_{A\cap B}(x) = \mu_A(x)\ \Lambda\ \mu_B(x) = \mathrm{Min}(\mu_A(x), \mu_B(x)) \text{---} 2.7$$

Complement

$$\mu_{\hat{A}}(x) = 1 - \mu_A(x) \text{---} 2.8$$

Excluded middle axioms, extended for fuzzy sets, are expressed by A∪A≠ X (axiom of excluded middle) and A∩A≠ ∅ (axiom of contradiction).

2.3.2.2 Fuzzy logic operation

The input to the fuzzy system is the output of the process, which is entered into the system via input interfaces. For example, in a temperature control application, the input data would be entered using an analog input module [17][18]. This input information would then go through the fuzzy logic process, where the processor would analyze a database to obtain an output. Fuzzy processing involves the execution of IF...THEN rules, which are based on the input conditions. An input's grade specifies how well it fits into a particular graphic set (e.g., too little, normal, too much). Note that input data may also be represented as a count value ranging or as a percentage of error deviation. If the fuzzy logic system utilizes an analog input that has a count range from 0 to 4095, the graphs representing the input will cover the span from 0 to

4095 counts. Furthermore, the analog input information (0–4095 counts) may represent an error range, from –50% to +50%, of a process [17].

The output of a fuzzy controller is also defined by grades, with the grade determining the appropriate output value for the control element.

Figure 2.4 illustrates a fuzzy logic cooling system chart with both input and output grades, where the horizontal axis is the input condition (temperature) and the vertical axis is the output (air-conditioner motor speed). In this chart, a single input can trigger more than one output condition. For example, if the input temperature is 137.5°F, then the temperature is part of two input curves—it is 50% too cool and 50% normal. Consequently, the input will trigger two outputs—the too cool input condition will trigger a less speed output, while the normal input will trigger a normal speed output condition. Since the fuzzy logic controller can have only one output, it completes a process called defuzzification to determine the actual final output value.

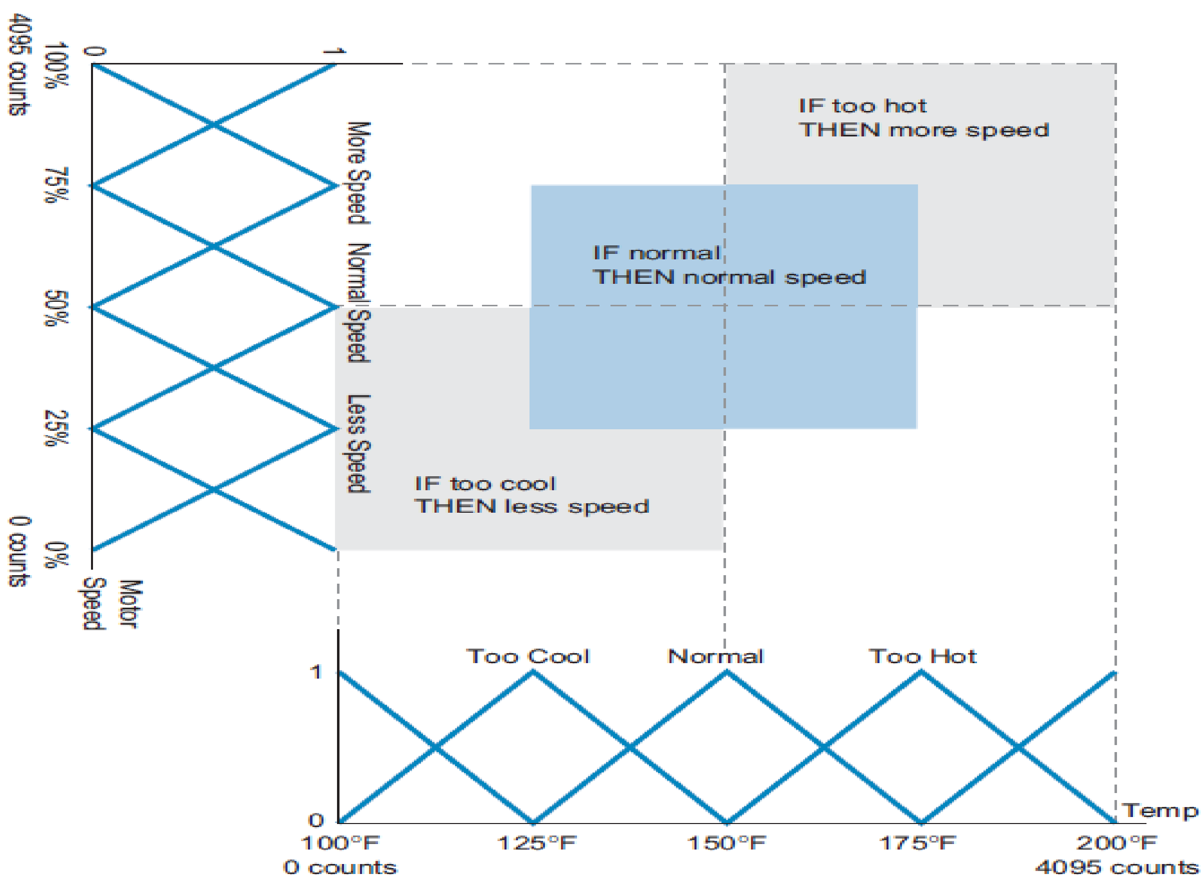


Figure 2. 4 Fuzzy logic system chart showing both input and output chart

2.3.2.3 FUZZY LOGIC CONTROL COMPONENTS

The three main actions performed by a fuzzy logic controller are:

- Fuzzification
- Fuzzy processing
- Defuzzification

As shown in Figure 2.5, when the fuzzy controller receives the input data, it translates it into a fuzzy form. This process is called fuzzification [17-19]. The controller then performs fuzzy processing, which involves the evaluation of the input information according to IF…THEN rules created by the user during the fuzzy control system's programming and design stages. Once the fuzzy controller finishes the rule-processing stage and arrives at an outcome conclusion, it begins the defuzzification process. In this final step, the fuzzy controller converts the output conclusions into "real" output data and sends this data to the process via an output module interface. If the fuzzy logic controller is located in the PLC rack and does not have a direct or built-in I/O interface with the process, then it will send the defuzzification output to the PLC memory location that maps the process's output interface module.

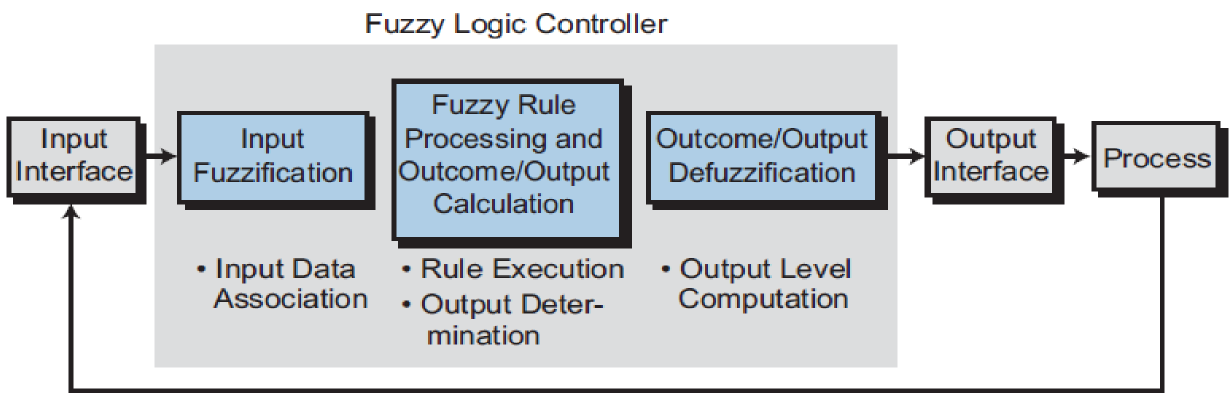


Figure 2. 5 Fuzzy logic controller operations

FUZZIFICATION

Fuzzification is the process of making a crisp quantity fuzzy[17][18]. It transforms the physical values of the error signal, rate of change of error which is input to the fuzzy logic controller, into a fuzzy set consisting of an interval for the range of the input values and an associate membership function describing the degrees of the confidence of the input belonging to this range. The conversion process is performed by a membership function. The purpose of this fuzzification step is to make the input physical signal compatible with the fuzzy control rule base in the core of the controller. Fuzzification consists of two main components:

1. **Membership Functions:** During fuzzification, a fuzzy logic controller receives input data, also known as the fuzzy variable, and analyzes it according to user-defined charts called membership functions .Membership functions can have many shapes, depending on the data set, but the most common are the S, Z, Π and Λ shapes shown in Figure 2.6. Note that these membership functions are made up of connecting line segments defined by the lines' end points. Each membership function can have up to three line segments with a maximum of four end points. The grade at each end point must have a value of 0 or 1.

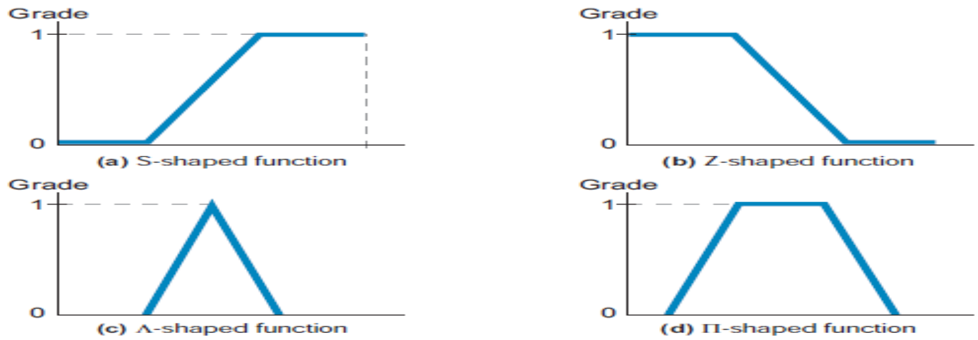


Figure 2. 6 Membership function shapes: (a) S, (b) Z, (c) Λ, and (d) Π.

2. **Labels**: Each fuzzy controller input can have several membership functions, with seven being the maximum and the norm, that define its conditions. Each membership function is defined by a name called a label. For example, an input variable such as temperature might have five membership functions labeled as cold, cool, normal, warm, and hot. Generically, the seven membership functions have NL (negative large), NM (negative medium), NS (negative small), ZR (zero), PS (positive small), PM (positive medium) and PL (positive large) labels.

A group of membership functions forms a **fuzzy set**. Figure 2.7 shows a fuzzy set with five membership functions. Although most fuzzy sets have an odd number of labels, a set can also have an even number of labels. For example, a fuzzy set may have four or six labels in any shape, depending on how the inputs are defined in relationship to the membership function.

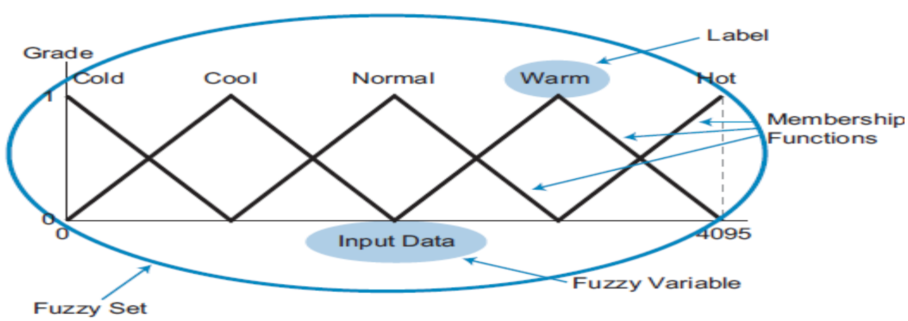


Figure 2. 7 Fuzzy set with five membership functions

FUZZY PROCESSING COMPONENTS

During fuzzy processing, the controller analyzes the input data, as defined by the membership functions, to arrive at a control output. During this stage, the processor performs two actions

1. **Rule Evaluation:** Fuzzy logic is based on the concept that most complicated problems are formed by a collection of simple problems and can, therefore, be easily solved. Fuzzy logic uses a reasoning process composed of IF...THEN rules, each providing a response or outcome. Basically, a rule is activated, or triggered, if an input condition satisfies the IF part of the rule statement. This results in a control output based on the THEN part of the rule statement. Sometimes, more than one rule is triggered at a time in a fuzzy control process. In this case, the controller evaluates all the rules to arrive at a single outcome value and then proceeds to the defuzzification process. Fuzzy logic rules with two inputs are often represented in matrix form to represent AND conditions. For example, Figure 2.8 illustrates a 3 × 3 matrix (9 rules) that uses two inputs, X1 and X2, and one output Y1. One advantage of this matrix representation is that it makes it easy to represent all the rules for a system. A five-label system translates into a 5 × 5 matrix with 25 rules, while a seven-label system produces a 7 × 7 matrix with 49 rules. An even membership function combination (e.g., a system with 6 labels for one input and 4 labels for another) will have a 24-rule matrix.

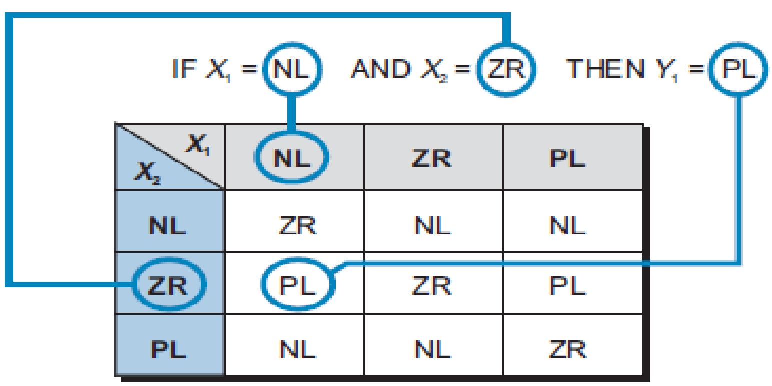


Figure 2. 8 Fuzzy logic rule matrix.

2. **Fuzzy Outcome Calculations**: Once a rule is triggered, meaning that the input data belongs to a membership function that satisfies the rule's IF statement, the rule will generate an output outcome. This fuzzy output is composed of one or more membership functions (with labels), which have grades associated with them. The outcome's membership function grade is affected by the grade level of the input data in its input membership function. In Figure 2.9a, the fuzzy input FI of 60% belongs to two membership functions, ZR and PS, corresponding to the grades of 0.6 and 0.4,

respectively. These two grades will have an impact on the amount of the output (see Figure 2.9b) by intersecting the output membership functions at the same grade levels (0.6 and 0.4). However, the output membership function that is selected for the final output value depends on the user's programming of the IF...THEN rules [17][18].

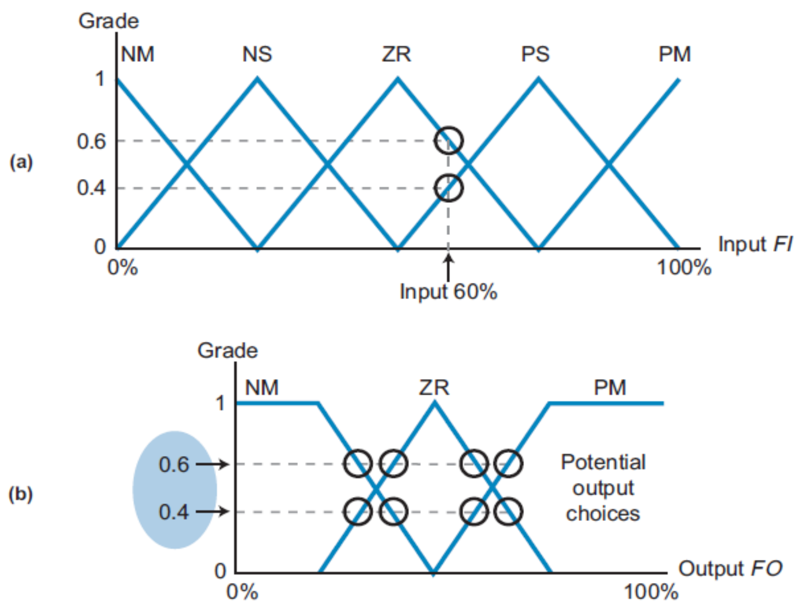


Figure 2. 9 (a) Fuzzy input grades and (b) the resulting output grades.

The Inference Mechanism provides the mechanism for referring to the rule base such that the appropriate rules are fired. The two most commonly used inference procedures in FLC are Mamdani's Max-Min and Max-Algebraic Product (or Max-Dot) composition. The inference or firing with this fuzzy relation is performed via the operations between the fuzzified crisp input and the fuzzy relation representing the meaning of the overall set of rules. As a result of the composition, one obtains the fuzzy set describing the fuzzy value of the overall control output. In this thesis a Mamdani's Max-min composition inference method is used [17][18].

Max-Min composition:

Consider a simple system where each rule comprises two antecedents and one consequent. A fuzzy system with two non-interactive inputs x1 and x2 (antecedents) and a single output y (consequent) are described by a collection of n linguistic IF-THEN rules [17][18].

$$IF\ X_1\ is\ {A_1}^{(k)}\ and\ X_2\ is\ {A_2}^{(k)}\ THEN\ Y(K)\ is\ B^{(k)}, k = 1,2,3 \dots n$$

Where

${A_1}^{(k)}$ *and* ${A_2}^{(k)}$ are fuzzy sets representing the k^{th} antecedent pairs and B^k are the fuzzy sets representing the k^{th} consequent.

Based on the Mamdani's max-min composition method of inference, and for a set of disjunctive rules, the aggregated output for the n rules will be given by [17][18][19]:

$$\mu_B{}^{(k)} = MaxMin[\left(\mu_{A1}{}^{(k)} input(i)\right).\mu_{A2}{}^{(k)}\left(input(j)\right)] \text{--} 2.14$$

Where, i and j are are input fuzzy set variables and y is output fuzzy set variable. The equation in (2.14) has a simple graphical interpretation, as seen in Figure 2.10.

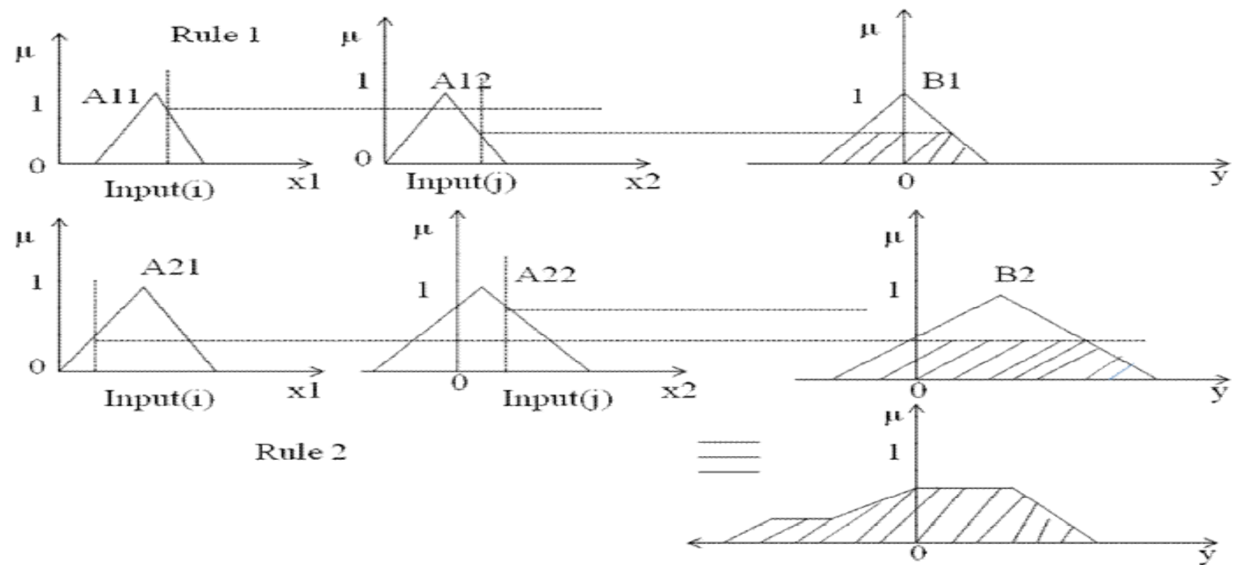


Figure 2. 10 Fuzzy inferencing using Mamdani's max-min compositional operation

The fuzzy IF-THEN rule in Figure 2.10 contains two rules. Both rules "IF x1 is A11 and x2 is A12 THEN y is B1" and "IF x1 is A21 and x2 is A22 THEN y is B2" are intersection fuzzy set operation and take the minimum membership values of the two inputs. Then the outputs of the two rules aggregated using the union fuzzy set operation that takes the maximum membership values of each fuzzy rule outputs. For this specific example Equation (2.14) can be simplified as

μB(y) = Max [Min(μA11(input(i)),μA12(input(j)), Min (μA21(input(i)), μA22(input(j))].

DEFUZZIFICATION COMPONENTS

The final output value from the fuzzy controller depends on the defuzzification method used to compute the outcome values corresponding to each label. The defuzzification process examines all of the rule outcomes after they have been logically added and then computes a value that will be the final output of the fuzzy controller. The Fuzzy controller then sends this value to the output module. Thus, during defuzzification, the controller converts the fuzzy output into a real-life data value [17][18]. There are many defuzzification methods, but all are based on mathematical algorithms. The two most common defuzzification methods are

1. **Maximum Value Method:**. The maximum value method bases the final output value on the rule output with the highest membership function grade. This method is mainly used with discrete output membership functions.

2. **Center of Gravity Method:** The center of gravity method, also referred to as "calculating the centroid," mathematically obtains the center of mass of the triggered output membership functions. The center of gravity method is the most commonly used defuzzification method because it provides an accurate result based on the weighted values of several output membership functions. The center of gravity method applies to noncontinuous, or discrete, output membership functions, as well as continuous ones.

2.4 Related works

Author in reference [23] have developed "implementations of fuzzy logic controllers in temperature control of tempered glass furnace". This work presents a technique to maintain temperature of a furnace using Fuzzy Logic. The furnace temperature has to be controlled and maintained up to 1500 degree Celsius and the purpose of the furnace is to study the characteristics of welding materials at very high temperature. Since the range of temperature above ambient is fairly high and the furnace working area is also large, the phenomenon of maintaining the temperature at the desired value poses many challenges. In this work, Fuzzy Logic Control (FLC) system has been implemented by duly considering the working temperature range, as well as working area. The advent of the control system is, its ability to attain the set temperature at the set time and also to maintain it. The limitation of this work is it introduces steady state error.

Author in reference [24] have developed "Understanding PID Temperature Control as Applied to Vacuum Furnace Performance". This paper presents PID controller to control the temperature of vacuum furnace at desired temperature for the process, such as 1500°F. The problem with this controller is that the system is not robust because which don't have auto tuning feature and it is set-up manually.

Authors in reference [25] have developed "Sliding mode controllers for a tempered glass furnace". This paper investigates the design of two sliding mode controllers (SMCs) applied to a tempered glass furnace system. The main objective of the proposed controllers is to regulate the glass plate temperature, the upper-wall temperature and the lower Wall temperature in the furnace to a common desired temperature. The first controller is a conventional sliding mode controller. The key step in the design of this controller is the introduction of a nonlinear transformation that maps the dynamic model of the tempered glass furnace into the generalized controller canonical form; this step facilitates the design of the sliding mode controller. The second controller is based on a state-dependent coefficient (SDC) factorization of the tempered glass furnace dynamic model. Using an SDC factorization, a simplified sliding mode controller is designed.

Author in reference [26] have developed "Feedback Linearization based control of a horizontal radioactive furnace". This paper describes the application of the feedback linearization technique for control of a horizontal radioactive furnace which generally used for producing the tempered glass.

Author in reference [27] have developed "Temperature Control System Using Fuzzy Logic Technique" This paper presents a fuzzy logic based-temperature control system, which consists of a microcontroller, temperature sensor, and operational amplifier, Analogue to Digital Converter, display interface circuit and output interface circuit. It contains a design approach that uses fuzzy logic technique to achieve a controlled temperature output function

Form the above review it is clear that different literatures are reviewed about glass tempering process, glass properties, electric heating furnace and temperature controllers that have been used in this thesis. Beside that different related works are reviewed related to glass tempering. As far as I now fuzzy supervised PID controller for temperature control of glass tempering process is not modeled. There for in this thesis a fault tolerant fuzzy-PID temperature controller is modeled and simulated.

CHAPTER THREE

3 MATERIALS AND METHODS

This chapter deals with the materials and the methods used in accomplishing the thesis. The materials used are digital computer and MATLAB/SIMULINK software. Modeling, designing and analyzing are the methods used. The following sections present each of these in detail.

3.1 Modeling of glass tempering process

The first step in the analysis and design of the control system is mathematical modeling of the different components. The transfer function method is widely used in designing control systems. After proper assumptions and approximations are made to linearize the mathematical equations describing the components, transfer functions are obtained. Thus, using these transfer functions, the tempering process is modeled for temperature control.

The block diagram in Figure 3.1 shows the main components of glass tempering process of heating section. Before designing the temperature control system, the appropriate model for each component should be obtained [28].

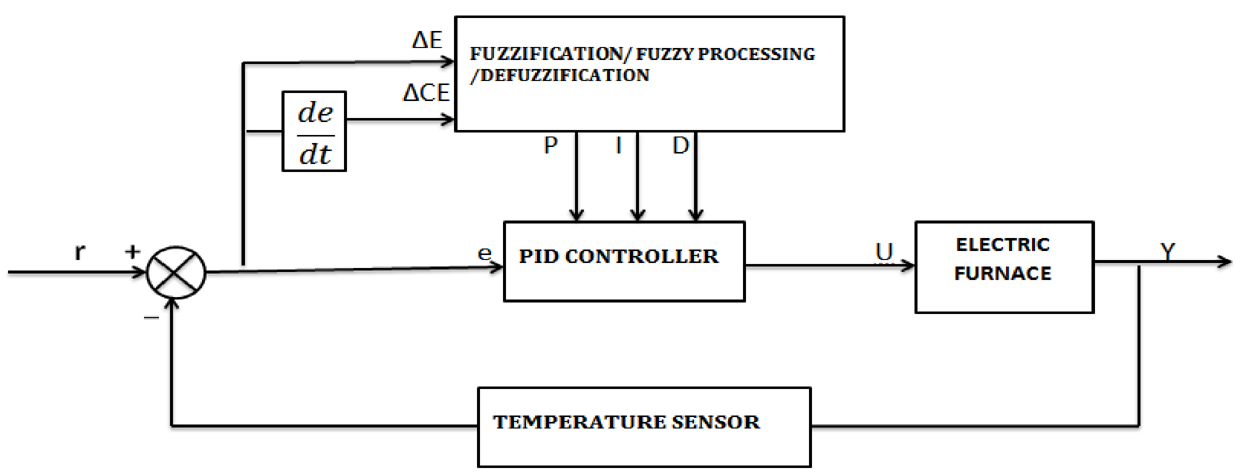


Figure 3. 1 General block diagram of fuzzy pid temperature control for glass tempering furnace

3.1.1 Modeling of electric furnace

In glass tempering process electric furnace is used to heat the glass to the desired value. Thus electric furnace converts electric power/energy to heat energy and heats the glass through conduction, convention and radiation. The temperature process of electric furnace is given by first order plus time delay equation [28][29][30][31].

$$G(S)\frac{k\,e^{-Ds}}{Ts+1} \text{---} 3.1$$

Where k is magnification value, T is time constant and D is pure lag time.

Assuming the walls of the glass tempering furnace has ideal heat insulation and there are no losses into surrounding. The equation of heat balance can be written as follows [32][33]:

$$\rho CV\,\frac{d\theta}{dt} = H_1 - H_2 \text{ --} 3.2$$

Where:

- ρ is the material density, ton/m3,
- C is the specific heat of the material, Mcal / deg .ton,
- V is the volume of the furnace, $m3$,
- θ is the temperature in zone, c^o ,
- $H_1 = Q_1\rho_1 c\theta_1$ is the amount heat flow with material $Q1$ Mcal/hour,
- $Q1$ is the flow rate input m3/hour,
- $H_2 = Q_2\rho_2 c\theta_2$ is the amount heat flow with material $Q2$ Mcal/hour, $Q2$ is the flow rate output m3 /hour.

Using the following assumptions:

I. $\theta(t) = \theta_2(t)$
II. $Q_1 = Q_2 = Q$

Substituting the assumptions in equation (3.2) and taking Laplace transform gives

$$\frac{\theta(s)}{\theta_1(s)} = \frac{1}{\frac{V}{Q}s+1} \cong \frac{1}{Ts+1} \text{ ---} 3.3$$

Where, T= $\frac{V}{Q}$, time constant

And considering H_1 as input the transfer function can be

$$\frac{\theta(s)}{H_1(s)} = \frac{K}{Ts+1} \quad \text{--} 3.4$$

Where $K = \frac{1}{Q\rho C_p}$, magnification value and

$T = \frac{V}{Q}$, time constant --3.5

To model the delay time:

The process involves the flow of ordinary glass at a constant velocity v, through a perfectly insulated furnace of length L. The glass is initially at a constant temperature T0, uniformly, throughout the entire furnace length. At time t = 0, the temperature of the glass coming in at the furnace inlet (z = 0) is changed to $T_i(t)$. we are now interested in investigating how the temperature at the furnace outlet (z = L) responds to such an input change [34].

It is easy to see that any temperature changes implemented at the furnace inlet will not be registered instantaneously 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 velocity is *v,* and it is assumed constant, the time for each glass element to traverse the required distance from the furnace inlet to the outlet is *L/v.* Since the furnace is also assumed to be perfectly insulated, there will be no heat losses, or any other changes for that matter, experienced by each glass element in the process. Thus, any changes implemented in the inlet will be preserved intact, to be observed at the outlet after the time required to traverse the entire pipe length has elapsed [34].

Mathematical Model and Analysis:

Consider an element of 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 [34]:

$$\rho A c_p \nabla Z \ \frac{\partial T}{\partial t} = \rho A v c_p (\mathrm{T} - T^*)/_Z - \ \rho A v c_p (\mathrm{T} - T^*)/_{Z+\nabla Z} \text{ ------------------------3.6}$$

Where:

ρ and c_p are, respectively, the glass density, and specific heat capacity
T* is the usual, reference temperature

By dividing through by ∇Z, and taking the limits as $\nabla Z \ \rightarrow 0$, Eq. (6) becomes:

$$\frac{\partial T(z,t)}{\partial t} + \mathrm{V}\frac{\partial T(z,t)}{\partial z} = 0 \text{--3.7}$$

The system whose dynamic behavior is represented by PDE's is known as a distributed parameter system. Thus the process currently under consideration is one of such systems.

Let us now define the deviation variable:

$$Y(z,t) = T(z,t) - T(0) \text{ --3.8}$$

Where we recall that T(O) = T0 is the initial glass temperature at t = 0 along the furnace length. In terms of this deviation variable, the process model in Eq. (3.7) is

$$\frac{\partial Y(z,t)}{\partial t} + \mathrm{V}\frac{\partial Y(z,t)}{\partial z} = 0 \text{---3.9}$$

To develop a transfer function model for this process, we must now take Laplace transforms of the expression in Eq. (3.9). From the definition of the Laplace transform, we have [34]:

$$sy(z,s) + v\int_0^\infty e^{-st}\,\frac{\partial y(z,t)}{\partial z}\,dt = 0 \text{ --3.10}$$

Since the first term is simply the Laplace transform (with respect to time) of the indicated time derivative. For the second term in Eq. (3.10), interchanging the order of integration and differentiation (an exercise which can be shown to be valid in this case) gives:

$$\int_0^\infty e^{-st}\,\frac{\partial y(z,t)}{\partial z}\,dt = \frac{d}{dz}\int_0^\infty e^{-st}\,y(z,t)dt = \ \frac{d}{dz}y(z,s) \text{------------------------3.11}$$

So that Eq. (3.7) becomes:

$$sy(z,s) + v\ \frac{d}{dz}y(z,s) = 0 \text{--3.12}$$

The solution of the ODE in Eq. (3.12) is easily obtained as:

$$y(z,s) = C_1 e^{\left(\frac{-z}{v}\right)s} \text{------} 3.13$$

We may evaluate the arbitrary constant C1 by using the entrance condition at z = 0, which is $y(z,s) = y(0,s)$.

So that Eq. (14) becomes

$$y(z,s) = y(0,s) e^{\left(\frac{-z}{v}\right)s} \text{------} 3.14$$

Eq. (3.14) represents the relationship between the Laplace transform of the inlet temperature and the Laplace transform of the temperature at any other point z along the length of the furnace. Observe further that the term z/v is the ratio of the distance traveled in arriving at this point, to the flow velocity, a direct measure of the time taken to traverse the indicated distance. We are particularly concerned with the situation at the furnace exit; i.e., at z = L. In this case, Eq. (3.14) becomes:

$$y(L,s) = y(0,s) e^{\left(\frac{-L}{v}\right)s} \text{------} 3.15$$

And if we define as D the time it takes for a glass element to traverse the entire length of the furnace, i.e.:

Then we have:

$$D = \frac{L}{V} \text{------} 3.16$$

$$y(L,s) = y(0,s) e^{-Ds}) \text{------} 3.17$$

Thus the transfer function relating the process input (changes in the inlet temperature, y(0,s)) to the process output (changes in the exit temperature, y(L,s)) is clearly seen from Eq. (13.17) to be given by:

$$g(s) = e^{-Ds} \text{------} 3.18$$

We may now use Eq. (3.18) to investigate how this process will respond to changes 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 Laplace transforms. We have, for any input u(t) that

$$y(L,s) = e^{-Ds}u(s) \text{---} 3.19$$

Which, when converted back to the time domain gives the result:

$$Y(L,t) = u(t - D) \text{---} 3.20$$

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

Heat flow rate:

The heat flow rate Q can be calculated as [35]

$$Q = \frac{H}{c_{pa}\rho_a(T_h - T_L)} \text{---} 3.21$$

Where

- ✓ H is the amount heat flow (input power)*in* material Mcal/hour,
- ✓ Q is the flow rate of heat through array of nozzles to glass surface by compressed air (radiation and convention)
- ✓ ρ_a is density of air
- ✓ c_{pa} is the specific heat of air, Mcal / deg .ton,
- ✓ T_h is heating temperature
- ✓ T_L is lower temp/room temperature

The required amount of heat flow (input power) H can be calculated as

1. By conduction [36]

   Rate of heat conduction $\infty \frac{(Area)(Temperature\ difference)}{\text{Thickness}}$

$$H_{cond} = kA\frac{\nabla T}{\nabla X} \text{---} 3.22$$

Where,

- ✓ ($k = 0.78\frac{\text{w}}{\text{m,°C}}$)is is the thermal conductivity of the glass
- ✓ ($\nabla X = 0.031m$) Thickness of the glass and furnace chamber that has physical contact to the heater

  Thus $H_{cond} = kA\frac{\nabla T}{\nabla X} = 0.78\frac{\text{w}}{\text{m,°C}} * 4m^2 * \frac{590°\text{C}}{0.031m} = 59380.64watt = 59.38kw$

2. By convention and radiation

By radiation heat transfer to or from a surface surrounded by a gas such as air occurs parallel to conduction (or convection, if there is bulk gas motion) between the surface and the gas [36]. Thus the total heat transfer is determined 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 includes the effects of both convection and radiation

$$H_{conv\&radi} = h_{combined} A_s (T_s - T_\infty) \text{---} 3.23$$

Where,

- ✓ ($h_{combined}$ is the $convection\&radiation\ heat\ transfer\ coefficient$ in $\frac{w}{m^2,°C}$, $take\ it$ 25.685 $\frac{w}{m^2,°C}$)
- ✓ T_s is surface temperature and ($T_\infty = 30°C$) is temperature of the gas.
- ✓ A_s is surface area of the glass to be heated.

Thus H_{conv} =60,616.6 watt

Therefore total amount of heat transfer is

$$H_{total} = H_{cond} + H_{conv\&radi} \text{---} 3.24$$

$$= 119997.24 \approx 120kw$$

Assumptions

- volume of the furnace chamber is chosen to be V=4m^3
- velocity of the glass is chosen to be $v = \frac{0.1m}{s}$
- Length of furnace =2m

c_p=0.18 $cal/gram^oC$

ρ=2.5gram/cm^3

Electric Power =120kw=120kJ/s

Thus substituting in Eq.(3.21)

Q= 0.16m^3/s

Substituting the values of Q , V,ρ and C_p in Eq.(3.5) yields:

K=13.07 and T=25s

And the delay time $D = \frac{L}{v}$ =20s

Thus the dynamic model of glass tempering resistance furnace is

$$G(S) = \frac{\theta(s)}{H(s)} = \frac{k\ e^{-Ds}}{Ts+1} = \frac{13.07\ e^{-20s}}{25s+1} \text{--}3.25$$

3.1.2 Modeling of temperature sensor

Temperature may be measured with a variety of instruments that respond to temperature with an electrical signal, including thermocouples, thermistors, RTDs (resistance thermometry devices), etc. In this section, we address both the static (calibration) and dynamic (time response) characteristics of temperature sensors [37][38]..

Calibration of the sensor is determining the relationship between the actual quantity of interest (the temperature at some location in the glass) and the output given by the sensor (which can be a voltage, a current in a circuit, a digital representation, etc., depending on the instrument). When we speak of 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 component that produces a linear relationship between temperature and sensor output, even though the behavior of the sensing element itself may be nonlinear [37][38].

Thus we relate the physical temperature T_o and its sensor reading T_s given by [38]:

$$T_s = K_s T_o + b_s \text{---}3.26$$

In a handheld digital thermometer, the electronics are adjusted so that gain K_s is unity and bias b_s is zero: 26ºC produces a reading of 26ºC. In a control loop, however, we are more likely to have T_o 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 a good way to transmit signals over the sorts of distances that separate operating processes from their control rooms.

The sensor range is adjusted by varying K_s and b_s. Suppose that we wish to follow T_O over the range 622ºC to 638ºC. Then

$$4mA = K_s 612°\text{C} + b_s \text{ -----} 3.27$$

$$20mA = K_s 628°\text{C} + b_s \text{-----} 3.28$$

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

We express the sensor calibration in deviation variables by subtracting the reference state from (1). Suppose we wish to use 620ºC as a reference operating condition. At the reference, the sensor output will be 12 mA [38].

Sensor response is first-order. Hence, we may write the sensor transfer function as

$$T_s(s) = \frac{K_s}{\tau_s s+1} T_o(s) \text{-----} 3.29$$

Where

- ✓ $T_s(s)$ is sensor reading
- ✓ $T_o(s)$ is physical temperature the system
- ✓ K_s is the sensor gain that was determined by calibration in (3.27) and (3.28)and most of the time it is unity.
- ✓ τ_s is a time constant that depends on the mass of the sensor element and the rate of heat transfer to the sensor.

When the sensor is immersed in to the medium, the heat transferred to the sensor in the time interval dt

$$dQ = \alpha A(Q - Q_T)dt \text{-----} 3.30$$

The heat stored in the sensor:

$$dQ = \text{mcd}Q_T \text{-----} 3.31$$

Thus,

$$\alpha A(Q - Q_T)dt = \text{mcd}Q_T \text{-----} 3.32$$

$$\frac{mc}{\alpha A}\frac{dQ_T}{dt} + Q_T = 0 \text{-----} 3.33$$

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

Where,

- α is coefficient of heat transfer
- A is the surface area of the sensor
- C is the heat capacity
- m is mass of sensor

Taking assumptions that the values of is τ_s equal to 10s.

3.2 Design and analysis of control system

There is different type of controllers used for temperature control of tempering furnace among those in the following section we discussed on PID and fuzzy-PID controllers.

3.2.1 Design and analysis of PID

PID controller can be understood as a controller that takes the present, the past, and the future of the error into consideration. Tuning a control loop is the adjustment of its control parameters (proportional band/gain, integral gain/reset, derivative gain/rate) to the optimum values for the desired control response. Stability (bounded oscillation) is a basic requirement, but beyond that, different systems have different behavior, different applications have different requirements, and requirements may conflict one with another. The transfer function Gc(s) of the PID controller is [10-13]:

$$\mathrm{Gc(s)} = K_p(1 + \frac{1}{\tau_i s} + \tau_d s) \quad \text{--} 3.34$$

$$\mathrm{Gc(s)} = K_p + \frac{K_i}{s} + K_d s \quad \text{---} 3.35$$

3.2.1.1 Loop tuning mechanism

Tuning a control loop is the adjustment of its control parameters (proportional band/gain, integral gain/reset, derivative gain/rate) to the optimum values for the desired control response. Stability (no unbounded oscillation) is a basic requirement, but beyond that, different systems have different behavior, different applications have different requirements, and requirements may conflict with one another [9] [15][16].

PID tuning is a difficult problem, even though there are only three parameters and in principle is simple to describe, because it must satisfy complex criteria within the limitations of PID control. There are accordingly various methods for loop tuning.

I. Manual tuning

For manual tuning in online system one of the methods is first setting all gains to zero. Increase the K_p value until the constant oscillation of the output is obtained 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 too much K_i can be a cause of instability. Finally, increaseK_d, if required, until the system becomes acceptably quick to reach the Set Point and decrease the oscillation. However, too much KD will lead to overshoot in the system. A quick PID controller loop normally has a little overshoots to achieve the desired value very fast; but in some systems cannot receive overshoot. In these systems a KP should be set remarkably less than half of the KP setting which provoked oscillation, to make over-damped in the closed-loop system [9] [15].

Parameter	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

Table 3. 1 Effects of independent P, I and D tuning on closed-loop response

II. CONVENTIONAL PID TUNINGS TECHNIQUE

There are different tuning correlations used in conventional PID tunings technique among those some of the methods are discussed below.

A. Ziegler-Nichols Method

The Ziegler-Nichols design methods are the most popular methods used in process control to determine the parameters of a PID controller. Although these methods were presented in the 1940s, they are still widely used [9][13][14]. The step response method is based on an open-loop step response test of the process. Hence requiring the process to be stable, the unit step response of the process is characterized by two parameters L and T. These are determined by drawing a tangent line at the inflexion point, where the slope of the step response has its maximum value.

The intersections of the tangent and the coordinate axes give the process parameters as shown in Figure 3.2, and these are used in calculating the controller parameters. Since our process model is first order plus dead time as indicated in equation (3.25) we can use the values of delay time (D/L), time constant (T) and DC gain (K) to calculate the values of K_p, K_i *and* K_d through different tuning correlations. The parameters for PID controllers obtained from the Ziegler-Nichols step response method are shown in Table 3.2.

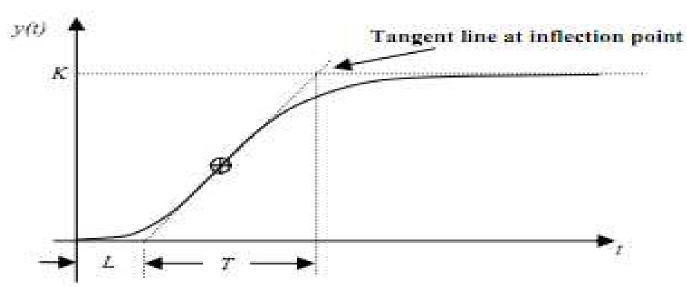


Figure 3. 2 Response curve for Z-N method

Controller	K_c	τ_i	τ_d
P	$(1/K)(T/D)$		
PI	$(0.9/K)(T/D$	3.3D	
PID	$(1.2/K)(T/D$	2D	0.5D

Table 3. 2 ZN PID controller parameter for first order plus dead time model

For PID controller

- K_c=0.1148
- K_d= 1.148(τ_d=10)
- K_i=0.00269(τ_i=40)

Thus, the controller transfer function is

$$Gc(s) = 0.1148 + \frac{0.00269}{s} + 1.148s \text{-----------------------------------} 3.36$$

B. Fine-tuned PID controller

The individual effect of K_P, K_i and K_d summarized in Table 3.2 can be very useful in fine tuning of PID controller. Beginning with the values of K_P, K_i and K_d obtained from Z-N step response method, unit step response for different combination of K_P, K_i and K_d were observed [9].

Closed loop Response	Overshoot	Settling time	Steady-state Error
Increase K_p	Increase	Small Increase	Decrease
Increase K_i	Increase	Increase	Large Decrease
Increase k_d	Increase	Decrease	Minor change

Table 3. 3 Effect of Kp ,Ki and Kd in . Fine-tuning technique

C. Cohen Coon Method

The Cohen-Coon method is a more complex version of the Ziegler-Nichols method [9] .This method is more sensitive than the Ziegler-Nichols method. The controller parameters with this method are given in table3.4.

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

Table 3. 4 Cohen Coon PID controller parameter for first order plus dead time model

For PID controller

For PID controller

- K_c=0.1466
- k_i= 0.00383(τ_i=37.9381)
- K_d =0.9308(τ_d=6.3492)

Thus, the controller transfer function is:

$$\mathrm{Gc(s)} = 0.1466 + \frac{0.00383}{s} + 0.9308s \text{------------------------------------} 3.37$$

D. ITAE Tuning Method

Another useful method for tuning of PID is ITAE method is described in [13]. Integral time multiplied by absolute error (ITAE) criterion is given by:

$$ITAE = \int_0^\infty t \mid e(t) \mid dt \text{---} 3.38$$

Where t is the time and e(t) is the error which is calculating the difference between the set point and the output. The controller parameters with this method are given by table 3.5. For a first order plus dead time model [13], solve for:

$$\frac{\partial ITAE}{\partial k_C} = 0, \ \frac{\partial ITAE}{\partial \tau_i} = 0 \ \ and \ \frac{\partial ITAE}{\partial \tau_d} = 0 \text{-------------------------------------} 3.39$$

Design for Load and Set-point changes yield different ITAE optimum.

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

Table 3. 5 ITAE PID controller parameter for first order plus dead time model

From table, we get

Load Settings:

$$A(D/T)^B = KK_c = T/\tau_i = \tau_d/T \text{--3.40}$$

Set point Settings:

$$A(D/T)^B = KK_c = \tau_d/T,\ T/\tau_i = A + B(D/T) \text{------------------------------------3.41}$$

i. Design load point change

For PID controller

- K_cK=1.357*(0.8)^(-0.947)
  K_c13.07=1.6763
  K_c=0.1282
- $T/\tau_i = 0.842 * (0.8)^{-0.738} = 0.9927,\ \ \tau_i = T/_{0.9927}$=25.1838
  K_i=0.0051
- $\tau_d/T = 0.381 * (0.8)^{-0.995} = 0.4757,\quad \tau_d = T * 1.6763 = 11.8925$
  K_d= 1.5246

Thus the controller transfer function is

$$\mathrm{Gc(s)} = 0.1282 + \frac{0.0051}{s} + 1.5246s \text{--------------------------------3.42}$$

ii. Design set point change

For PID controller

- K_cK=0.965*(0.8)^(-0.85)

  K_c13.07=1.665

  K_c=0.08925
- $T/\tau_i = 0.842 * (0.8)^{-0.738} = 0.9927,\ \ \tau_i = T/_{0.9927}$=36.8297

  K_i=0.0024
- $\tau_d/T = 0.308 * (0.8)^{-0.929} = 0.2503,\quad \tau_d = T * 1.6763 = 6.2575$

  K_d= 0.5584

Thus the controller transfer function is

$$\mathrm{Gc(s)} = 0.08925 + \frac{0.0024}{s} + 0.5584s \text{-------------------------------3.43}$$

E. AMIGO Tuning Method

In Astrom and Hagglund [9] an approximate method is proposed that accomplishes this goal in a simple way. The Method which is known as AMIGO (Approximate M-constrained integral gain Optimization), which consist in applying a set of equation to calculate the parameter of the Controller in a similar way to the procedure used in Ziegler- Nichols method. The suggested AMIGO Tuning Rule for PID Controller is:

$$Kp = \frac{1}{K}(0.2 + 0.45\frac{T}{L}) \text{-----} 3.44a$$

$$T_i = \frac{0.4L+0.8T}{L+0.1T}L \text{-----} 3.44b$$

$$T_d = \frac{0.5LT}{0.3L+T} \text{-----} 3.44c$$

From the above equation the values of those parameters Kp, Ki and Kd are obtained as 0.058339, 0.00187 and 0.47047 respectively.

Thus the controller transfer function is

$$\text{Gc(s)} = 0.058339 + \frac{0.00187}{s} + 0.47047s \text{-----} 3.45$$

3.2.2 Simulink model for pid controller

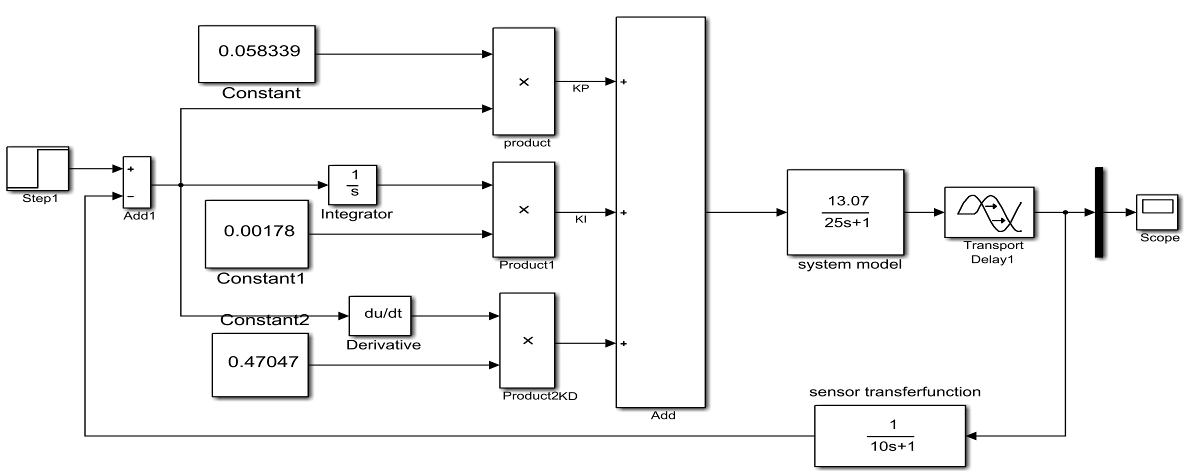


Figure 3. 3 Simulink model of PID controller

3.2.3 Modeling and Analysis of fuzzy PID controller

The fuzzy PID control is developed from tradition PID. Based on the fuzzy control theory, the fuzzy relationship between three PID parameters KP, KI, KD and the error E and error change rate EC can be established. According to different E and CE, the parameters KP, KI, KD can be self-adjusted online in order to make the controlled object have a good dynamic and static performance, which can meet different control requirement. The basic idea behind fuzzy logic control is to incorporate the experience of a human operator in the design of a controller in controlling a process whose input-output relationship is described by a collection of fuzzy control rules (e.g. IF-THEN rules) involving linguistic variables. When the parameters of PID are adjusted by the fuzzy control, the classical PID controller becomes the fuzzy PID controller.

A number of assumptions are implicit in a fuzzy control system design. Six basic assumptions are commonly made whenever a fuzzy rule-based control policy is selected [40][28].

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

The fuzzy controller used in this thesis can be depicted by a block diagram shown in Figure 3.4 A Fuzzy Logic Controller usually consists of Fuzzification, Knowledge base, Fuzzy reasoning mechanism (inference engine) and defuzzification :

- A **Fuzzification** unit which maps measured inputs of crisp value into fuzzy linguistic values to be used by a fuzzy reasoning mechanism.

- A **Knowledge base** (**KB**) which is the collection of expert control knowledge required to achieve the control objective. It contains knowledge about all the input and output fuzzy partitions. It includes the membership functions defining the input variables to the fuzzy rule base
- A **Fuzzy reasoning mechanism (inference engine)** that performs various fuzzy logic operations to infer the control action for the given fuzzy inputs. It contains rules in an antecedent-consequent form that sets the foundation for approximate (imprecise) reasoning.
- A **Defuzzification** unit which converts the inferred fuzzy control action into the required crisp control values to be entered into the system process.

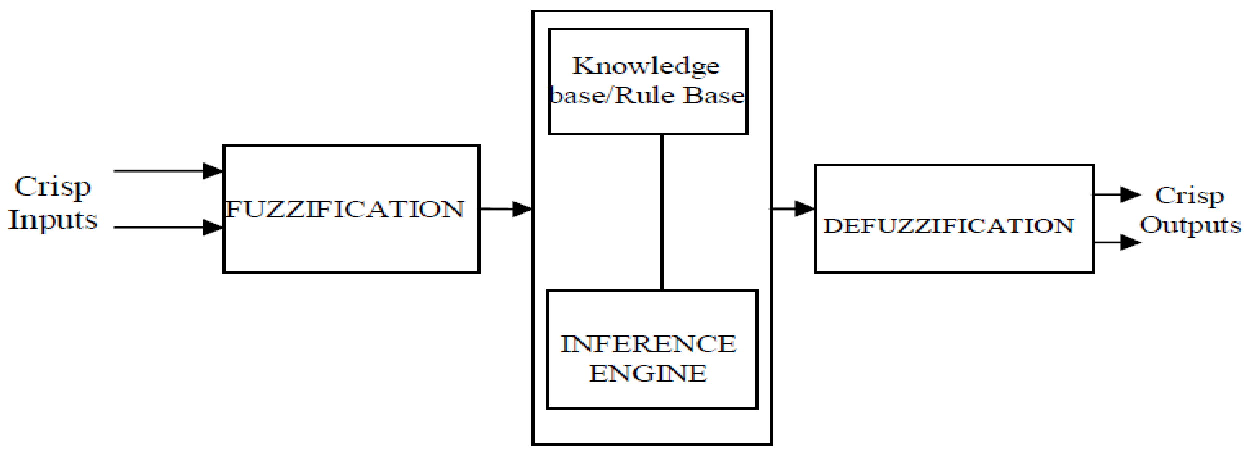


Figure 3. 4 General structure of fuzzy logic controller

The steps in designing a simple fuzzy control system are as follows [40]:

i. Identify the variables (inputs, states, and outputs) of the plant.
ii. Partition the universe of discourse or the interval spanned by each variable into a number of fuzzy subsets, assigning each a linguistic label.
iii. Assign or determine a membership function for each fuzzy subset

iv. Assign the fuzzy relationships between the inputs 'or states' fuzzy subsets on the one hand and the outputs 'fuzzy subsets on the other hand, thus forming the rule-base.

v. Choose appropriate scaling factors for the input and output variables in order to normalize the variables.

vi. Fuzzify the inputs to the controller.

vii. Use fuzzy approximate reasoning to infer the output contributed from each rule.

viii. Aggregate the fuzzy outputs recommended by each rule.

ix. Apply defuzzification to form a crisp output

3.2.3.1 Fuzzy inference system

First we defined on E in the domain of e linguistic variables and defined on the EC in the domain of ec linguistic variables [41]. Set the deviation e basic domain {el, eh} = {-30, 30}, error change rate ec basic domain {ecl, ech}= {-12,12}, △kP and △kD basic domain {-0.3,0.3}, △kI basic domain {-0.03,0.03}.

The typical FIS inputs are the signals of error E and change of error CE in temperature .The FIS output is the control action inferred from the fuzzy rules. The configuration of FIS for our fuzzy controller is given in the following figure.

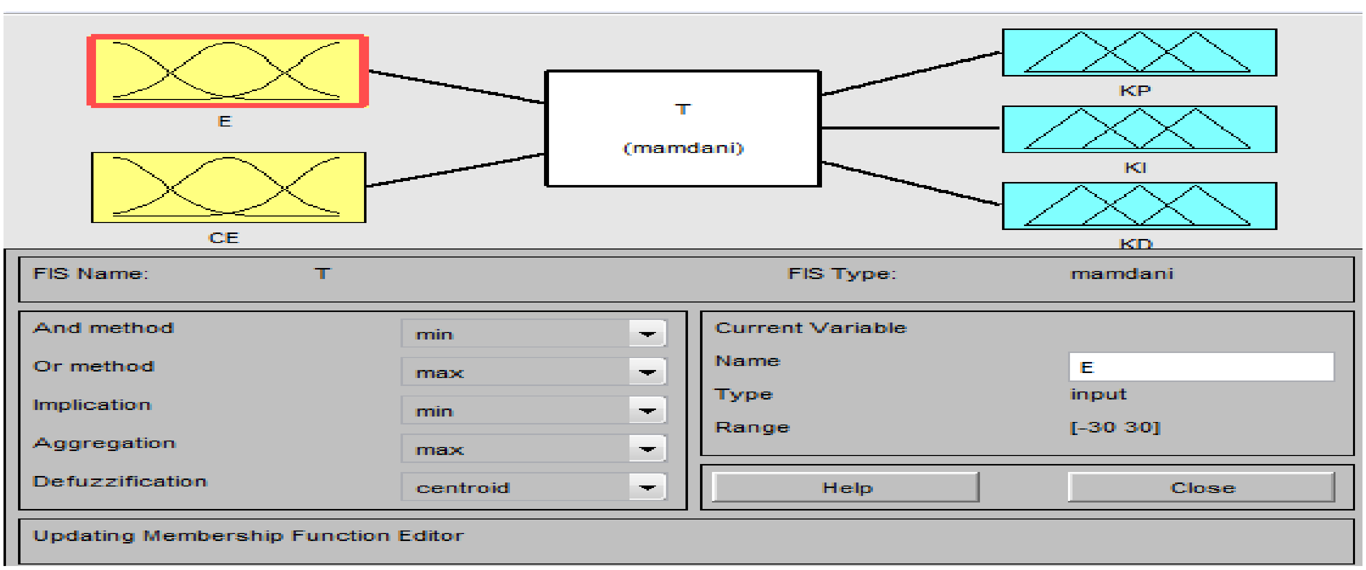


Figure 3. 5 Fuzzy interface block

Figure 3.6 (a), (b) and (c) shows the surface viewer of KP, KI and KD with E and CE respectively and figure shows rule viewer of fuzzy interface system. Where, the rule viewer of our fuzzy interface system is given in appendix A.

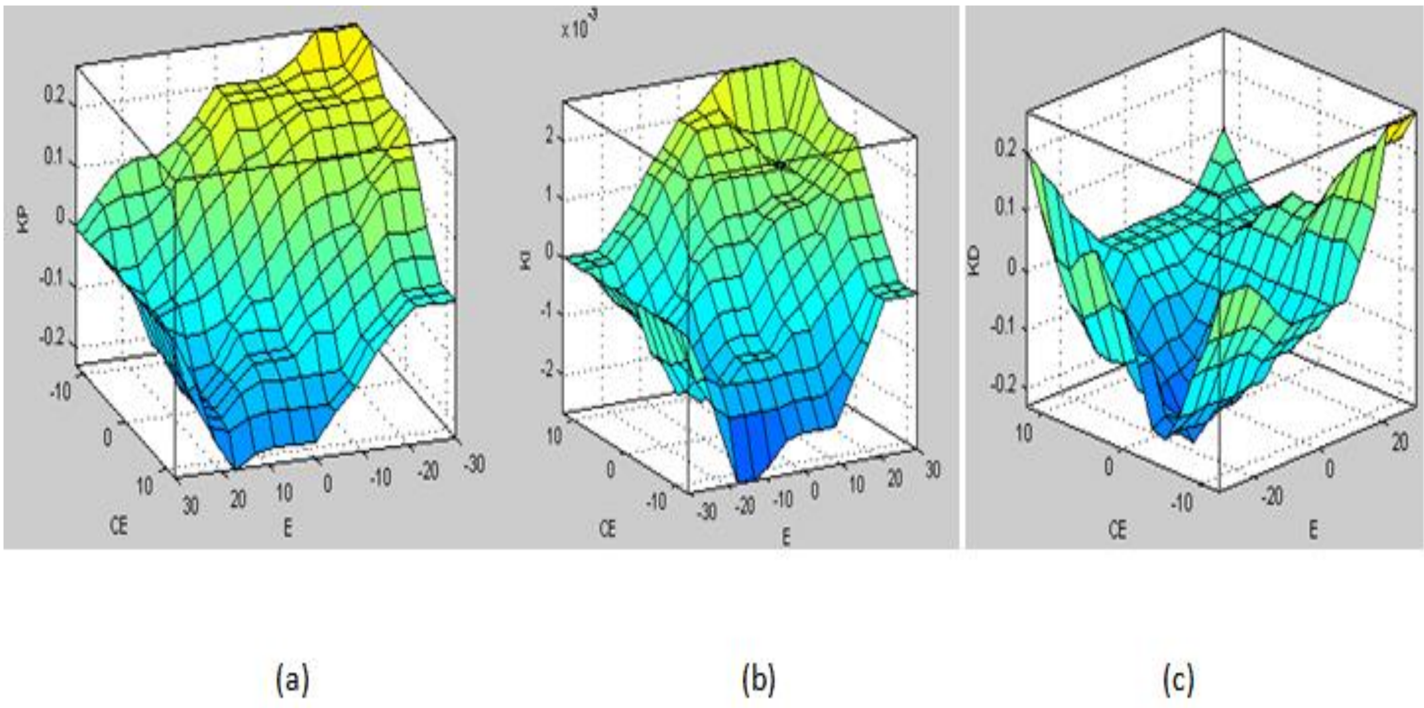


Figure 3. 6 Surface viewers of KP, KI and KD

3.2.3.2 Fuzzier Design

Figure 3.7 is a graph of the temperature error and change in temperature error fuzzy membership functions. It assigns a crisp value's degree of membership to seven fuzzy linguistic values -- NB, NM, NS, Z, PS, PM, and PB. The fuzzy input membership functions used for error in temperature and change of error in temperature are triangular type. The universe of discourse of each membership function is shown in Table 3.6 with degree of membership.

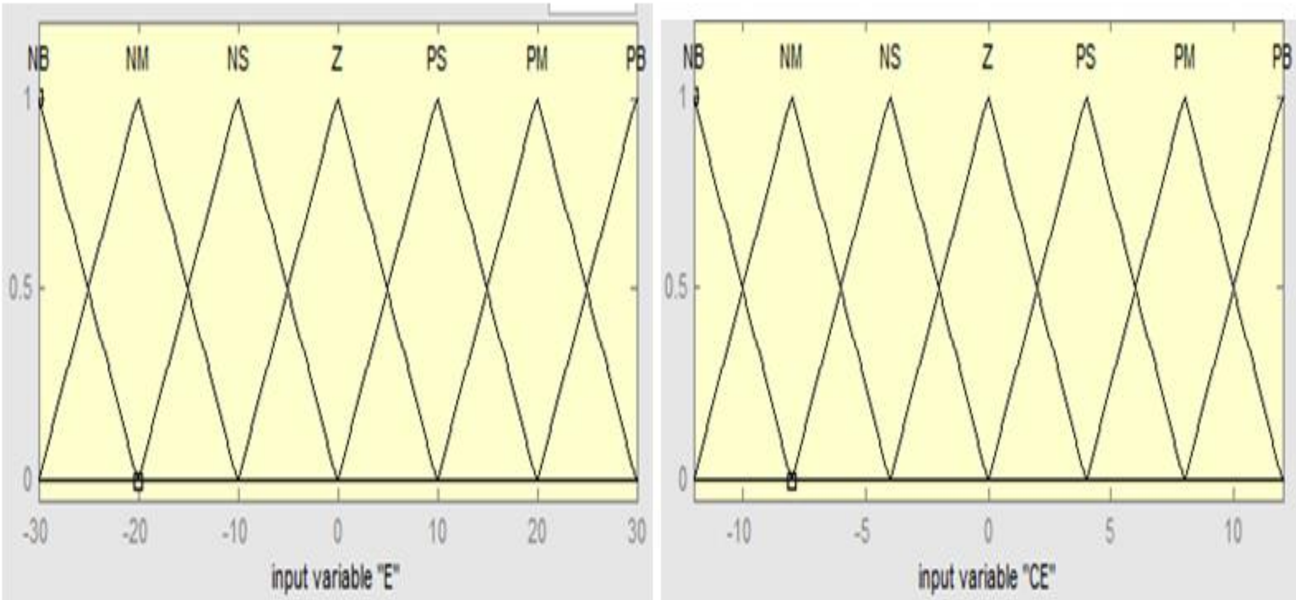


Table 3. 6 Member ship function for error and error rate in temperature

E	EC	Degree of membership						
		NB	NM	NS	Z	PS	PM	PB
-30 to -20	-12 to -8	1 to 0	0 to 1	0	0	0	0	0
-20 to -10	-8 to -4	0	1 to 0	0 to 1	0	0	0	0
-10 to 0	-4 to 0	0	0	1 to 0	0 to 1	0	0	0
0 to 10	0 to 4	0	0	0	1 to 0	0 to 1	0	0
10 to 20	4 to 8	0	0	0	0	1 to 0	0 to 1	0
20 to 30	8 to 12	0	0	0	0	0	1 to 0	0 to 1

Table 3. 7 Membership function universe of discourse of error and change of error in temperature

3.2.3.3 Rule base construction

The rules of fuzzy PID control are based on the control of people with long-term understanding of the theory lessons and learned technically artificial. Fuzzy self-tuning PID parameters rule [41]:

- ✓ When | e | is large, should take larger △kP and smaller △kD in order to speed up the response of the system, and take a smaller △kI in order to avoid a greater overshoot restrict the integral action
- ✓ When | e | is medium, in order to make the system has a smaller overshoot, △kP should be made smaller. At this point, △ kD has greater impact on the system, should get smaller, △kI values should be appropriate;
- ✓ When | e | is small, to make the system has good stability, should take a larger △kP and △kI, △kD should be appropriate in order to avoid oscillation occurs near the equilibrium point.

According to the above rules may draw the following 49 fuzzy relations. The rule tables for Δkp, Δki and Δkd are shown in Table 3.7, 3.8 and 3.9 respectively.

E	EC						
	NB	NM	NS	Z	PS	PM	PB
NB	PB	PB	PM	PM	PS	Z	Z
NM	PB	PM	PM	PS	PS	Z	Z
NS	PM	PM	PM	PS	Z	NS	NS
Z	PM	PM	PS	Z	NS	NS	NM
PS	PS	PS	Z	NS	NS	NS	NM
PM	PS	Z	NS	NS	NM	NS	NB
PB	Z	Z	Z	Z	Z	NM	Z

Table 3. 8 Rule table for Δkp

E	EC						
	NB	NM	NS	Z	PS	PM	PB
NB	Z	Z	Z	Z	Z	Z	Z
NM	NB	NS	NM	NS	NS	Z	Z
NS	NB	NS	NS	Z	Z	PS	PS
Z	NM	NS	NS	PS	PS	PM	PM
PS	NM	NS	Z	PS	PS	PM	PB
PM	Z	Z	PS	PS	PS	PSPS	PB
PB	Z	Z	PS	PM	PM	PM	PB

Table 3. 9 Rule table for Δki

E	EC						
	NB	NM	NS	Z	PS	PM	PB
NB	PS	NS	NB	Z	PS	Z	PM
NM	PS	NS	NB	NB	NB	NM	PS
NS	Z	NS	NM	NM	NS	Z	Z
Z	Z	NS	NS	NS	NS	Z	Z
PS	Z	Z	Z	Z	Z	Z	Z
PM	PB	PB	PS	PS	Z	Z	Z
PB	PB	PM	PS	Z	Z	Z	PS

Table 3. 10 Rule table for Δkd

3.2.3.4 Defuzzification Design

We used the Centroid de-fuzzification method to convert from the inference mechanism into the crisp values applied to the actual system. The output defuzzification membership functions for Δkp , Δki and Δkd are shown in Figures 3.8 and 3.9.

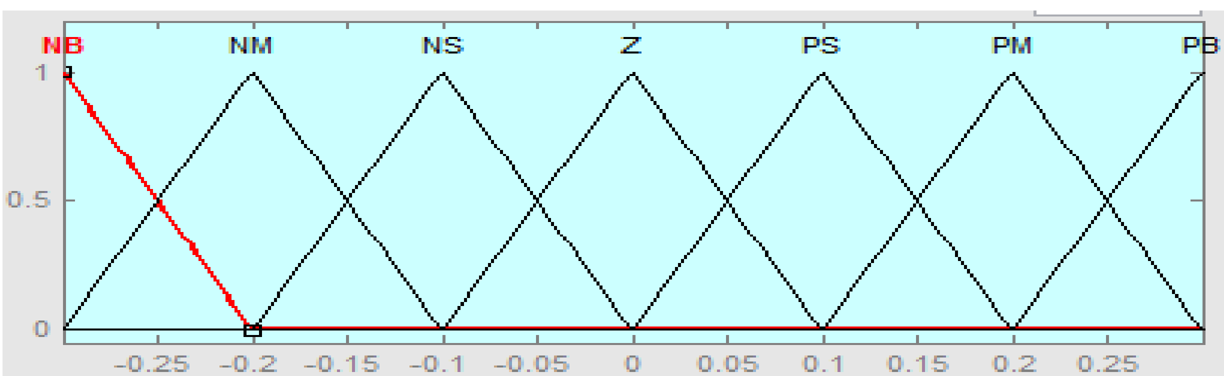


Figure 3. 7 Output membership function for Δkp and Δkd

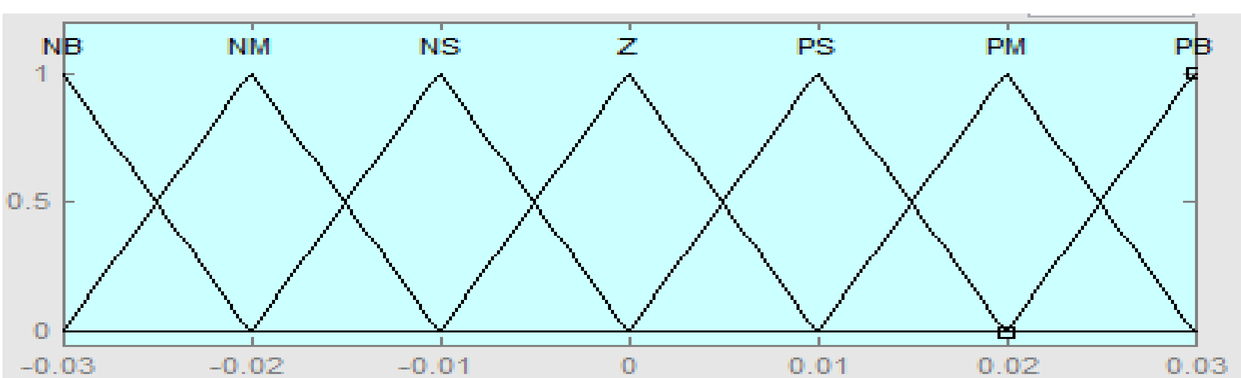


Figure 3. 8 Output membership function for Δki

Table 3.10 and Table 3.11 show universe of discourse for the fuzzy output variables Δkp &Δkd and Δki respectively.

Δkd	Degree of membership						
	NB	NM	NS	Z	PS	PM	PB
-0.03 to -0.02	1 to 0	0 to1	0	0	0	0	0
-0.02 to -0.01	0	1 to 0	0 to1	0	0	0	0
-0.01 to 0	0	0	1 to 0	0 to1	0	0	0
0 to 0.01	0	0	0	1 to 0	0 to1	0	0
0.01 to 0.02	0	0	0	0	1 to 0	0 to1	0
0.02 to 0.03	0	0	0	0	0	1 to 0	0 to1

Table 3. 11 Membership function universe of discourse for Δkp and Δkd

Δkp	Degree of membership						
	NB	NM	NS	Z	PS	PM	PB
-0.3 to -0.2	1 to 0	0 to1	0	0	0	0	0
-0.2 to -0.1	0	1 to 0	0 to1	0	0	0	0
-0.1 to 0	0	0	1 to 0	0 to1	0	0	0
0 to 0.1	0	0	0	1 to 0	0 to1	0	0
0.1 to 0.2	0	0	0	0	1 to 0	0 to1	0
0.2 to 0.3	0	0	0	0	0	1 to 0	0 to1

Table 3. 12 Membership function universe of discourse for Δki

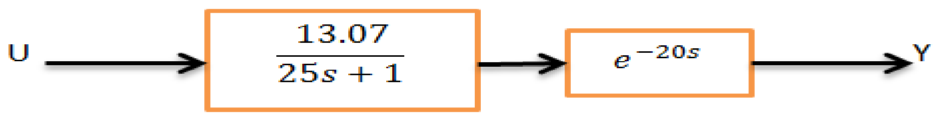
3.2.4 Digital controller

As a digital temperature controller is being used, the process model and senor transfer should be transformed into their discrete form for simulation studies. Since the process model has time delay many control design algorithms cannot handle directly. For example, techniques such as root locus, LQG (Linear-Quadratic-Gaussian), and pole placement do not work properly if time delays are present. A common technique is to replace delays with all-pass filters that approximate the delays.

To approximate time delays in continuous-time models, using pade command to compute a Padé approximation. The Padé approximation is valid only at low frequencies, and provides better frequency-domain approximation than time-domain approximation. It is therefore important to compare the true and approximate responses to choose the right approximation order and check the approximation validity.

Steps we use in Padé approximation

1. Create sample open-loop system with an output delay.



2. Compute the first-order Padé approximation of G(s) using pade MATLAB command.

```
s = tf('s');

G = exp(-20*s)/(25*s+1);

G1 = pade(P,1)

G1 =

   -13.07 s + 1.307

   ------------------,

   25 s^2 + 3.5 s + 0.1


Continuous-time transfer function.
```

This command replaces all time delays in G with a first-order approximation. Therefore, G1 is a second-order transfer function with no delays.

3. Compare the Temperature response of the original and approximate models using bodeplot

```
h = bodeoptions;

h.PhaseMatching = 'on';

bodeplot(G,'-b',G1,'-.r',{0.1,10},h)

legend('Exact delay','First-Order Pade','Location','SouthWest')
```

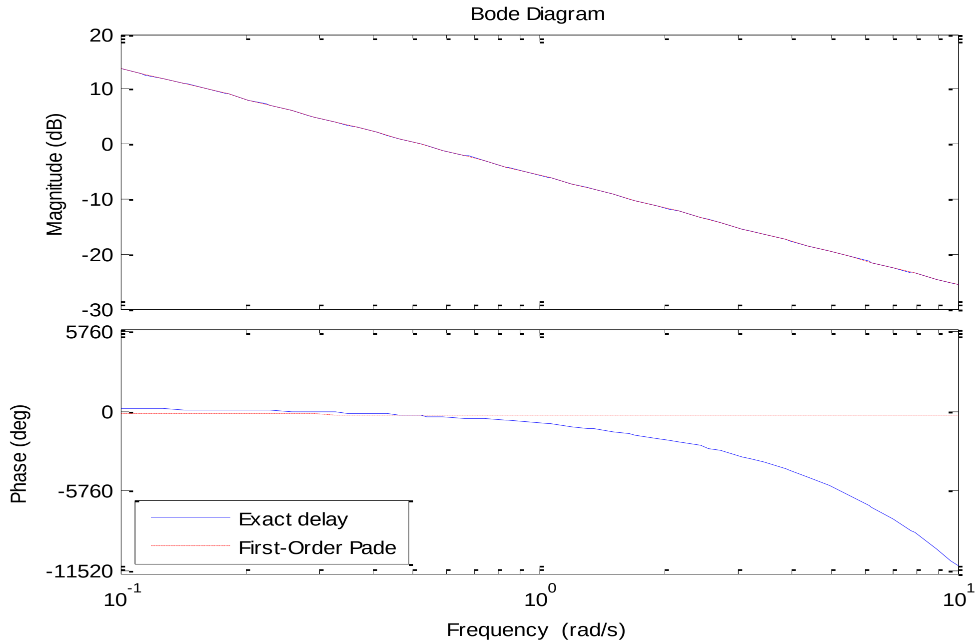


Figure 3. 9 Bode diagram for first order pade approximation

The magnitude of G and PG1 match exactly. However, the phase of G1 deviates from the phase of G

4. Increase the Padé approximation order to extend the Temperature band in which the phase approximation is good.

G3=pade(G,3)

G3 =

```
  -13.07 s^3 + 7.842 s^2 - 1.96 s + 0.1961
 -------------------------------------------
 25 s^4 + 16 s^3 + 4.35 s^2 + 0.525 s + 0.015
```

continuous-time transfer function.

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

bodeplot(G,'-b',G3,'-.r',G1,':g',{0.1 10},h)

legend('Exact delay','Third-Order Pade','First-Order Pade',... 'Location','SouthWest')

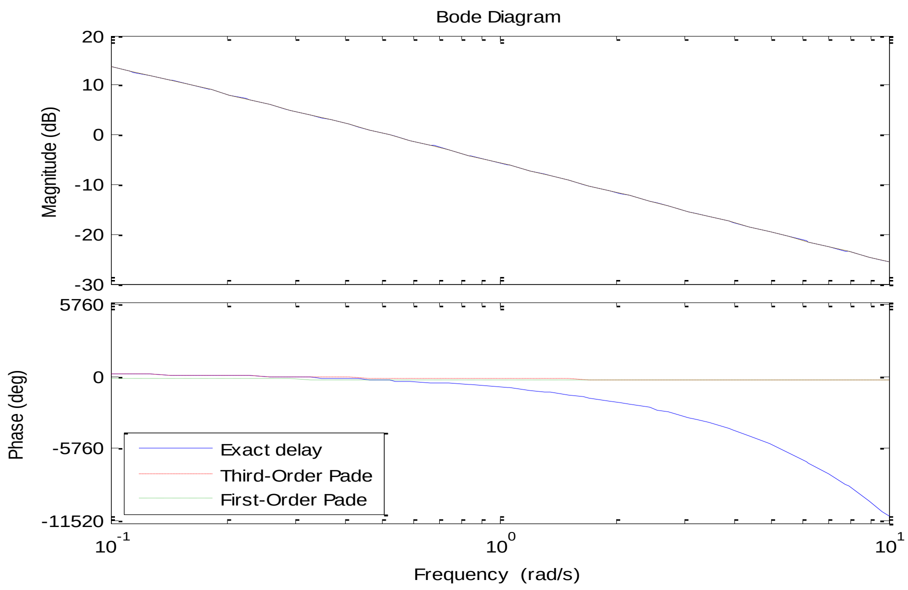


Figure 3. 10 Bode diagram for third order pade approximation

The phase approximation error is not changed by using a third-order Padé approximation. There is slight improvement using a ninth-order Padé approximation but using too high an approximation order may result in numerical issues and possibly unstable poles.

6. Compare the time domain responses of the original and approximated systems using step

```
step(P,'-b',Pnd3,'-.r',Pnd1,':k')
legend('Exact delay','Third-Order Pade','First-Order Pade',...
    'Location','Southeast')
```

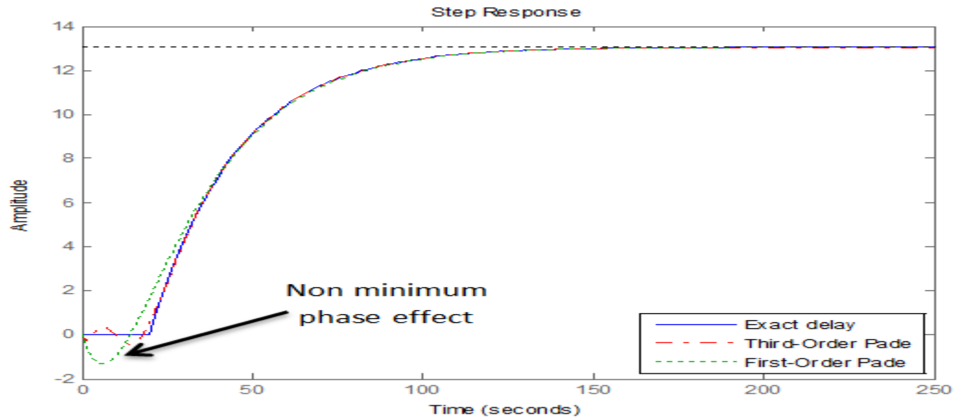


Figure 3. 11 Step response of the original and pade approximate transfer function

Using the Padé approximation introduces a non-minimum phase artifact ("wrong way" effect) in the initial transient response. The effect is reduced in the higher-order approximation.

Increasing the Padé approximation order extends the frequency band where the approximation is good. However, too high an approximation order may result in numerical issues and possibly unstable poles Thus for our Simulink model we use first-order pade approximation of the the process model

$$G(S) = \frac{-13.07s^2+1.307}{25s^2++3.5s+0.1} \text{--} 3.46$$

The continuous time transfer function is converted to its discrete time equivalent for simulation studies. The sampling frequency is taken 0.1 second and the microprocessor used in the controller is general purpose processor.

$$G(Z) = \frac{-0.4626Z+0.5113}{Z^2-1.866Z+0.8694} \text{--} 3.47$$

Also the PID controller is transformed into its discrete form. The discrete PID controllers are formulated directly from their corresponding Laplace transforms. For our system the control signal is given by equation 2.5 as:

$$\text{Gc(s)} = K_p + \frac{K_i}{s} + K_d s$$

Now, applying bilinear transformation to get an equivalent discrete PID controller, the expression in Equation (3.44) is obtained

$$\text{Gc(z)} = K_p + K_i * \frac{Ts\, z}{z-1} + K_d * \frac{Ts\, z-1}{z} \text{---------------------------------------} 3.48$$

Where, Ts is sampling time and taken as 0.1 secondes.

The PID parameter is tuned by using MATLAB/SIMULINK PID tuning extension tool box for fuzzy-pid and pid Simulink models [39]. The values of those parameters are obtained as $K_p = 0.03116, \quad K_i = 0.0016726 \;\; and \;\; K_d = 0.11165.$

It is assumed that kp, ki and kd are in prescribed ranges [$k_{p\,min}$ $k_{p\,max}$], [$k_{i\,min}$ $k_{i\,max}$] and [$k_{d\,min}$ $k_{d\,max}$] respectively. For convenience they are normalized in to the range between zero and one by the following linear transformation [42].

$$K_p{}' = \frac{kp - kp\ min}{kp\ max - kp\ min} \quad \text{------ 3.49}$$

$$K_i{}' = \frac{ki - ki\ min}{ki\ max - ki\ min} \quad \text{------ 3.50}$$

$$K_d{}' = \frac{kd - kd\ min}{kd\ max - kd\ min} \quad \text{------ 3.51}$$

Where, $k_p{}'$ $k_i{}'$ and $k_d{}'$ are the initial PID parameters, and kp, ki and kd are the adjusted PID parameters. The ranges of kp, ki amd kd are determined from the initial PID parameters as [0.03 0.04], [0.001 0.002] and [0.11 0.012] respectively.

3.2.5 Simulink model of overall system

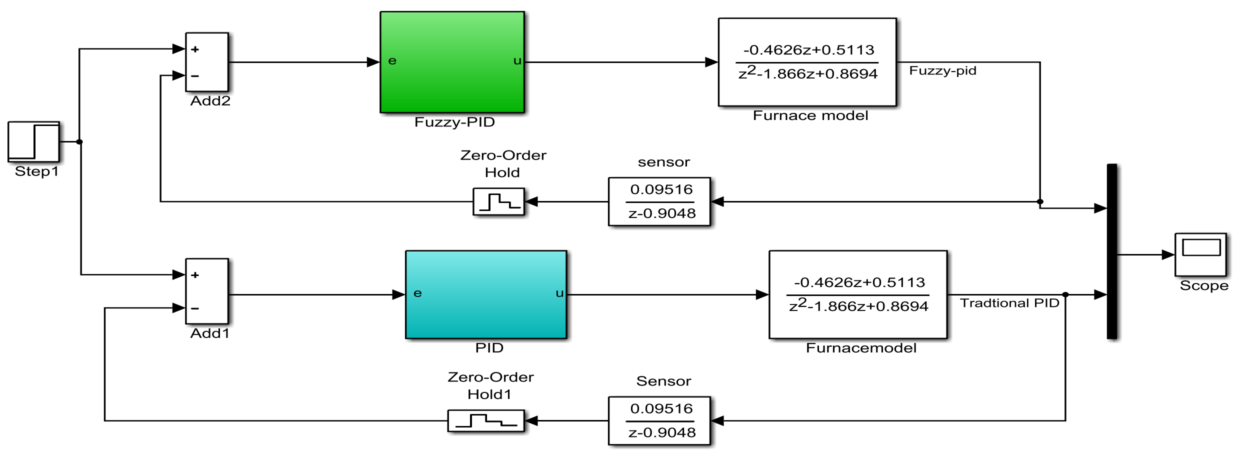


Figure 3. 12 Over all Simulink model of fuzzy pid and pid controller

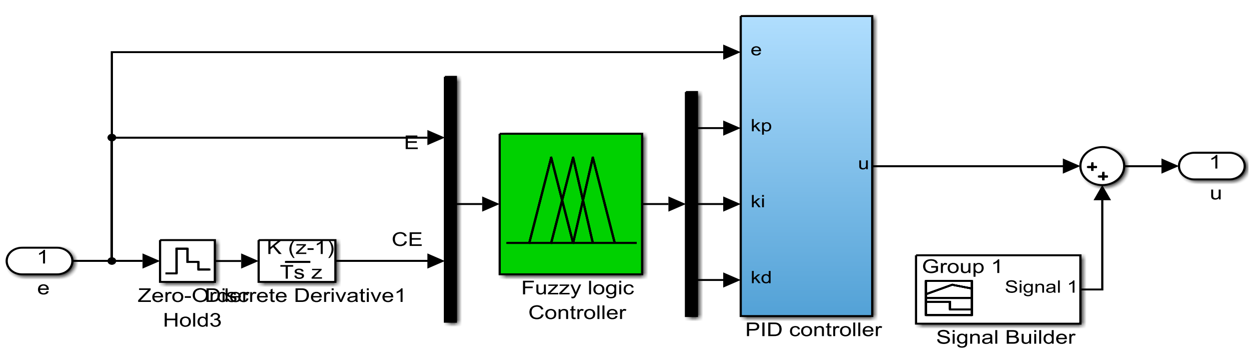


Figure 3. 13 Model components under fuzzy pid block

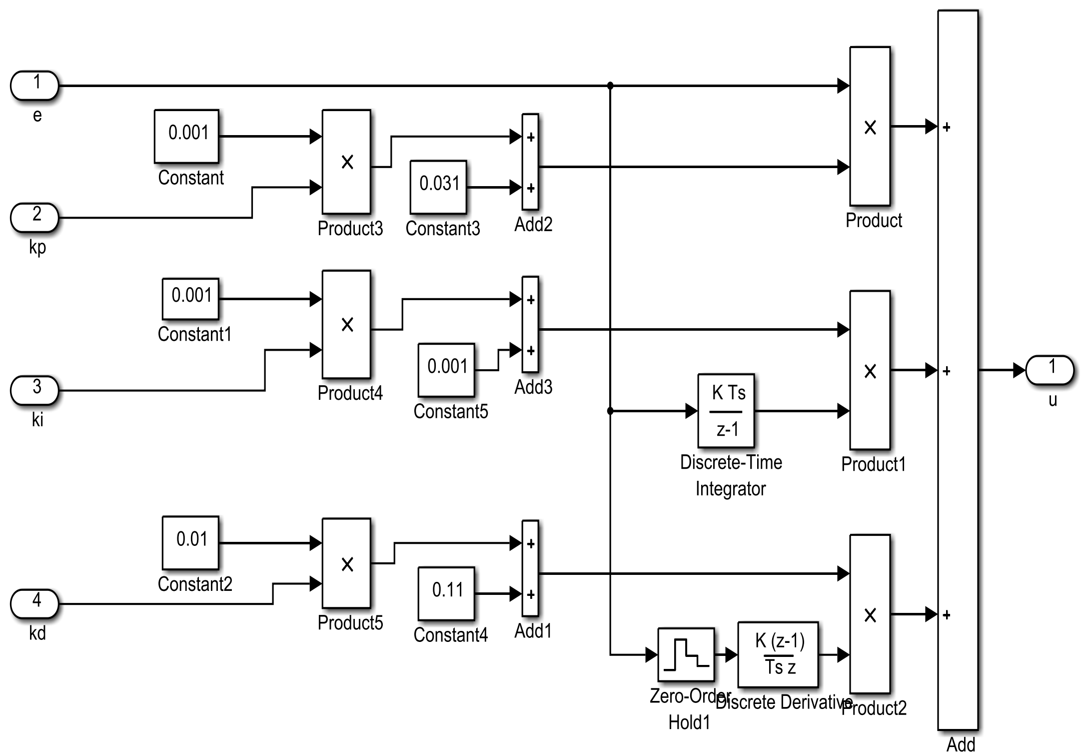


Figure 3. 14 PID controller block model components under fuzzy pid block

CHAPTER FOUR

4 SIMULATION RESULTS AND DISCUSSIONS

The electric furnace temperature control for glass tempering was modeled, deigned and analyzed in the materials and methods section of this thesis. This chapter deals with the results and discussion of MATLAB simulations. It is concerned with the simulation results of the closed loop control of the system using Fuzzy-PID and PID with MATLAB/SIMULINK. The simulation is analyzed by step response, disturbance rejection and parameter variation (robustness) of the closed loop system. SIMULINK® is a toolbox extension of the MATLAB program. It is a program for simulating dynamic systems. Simulink has the advantages of being capable of complex dynamic system simulations, graphical environment with visual real time programming and broad selection of tool boxes. Its graphical interface allows selection of functional blocks, their placement on a worksheet, selection of their functional parameters interactively, and description of signal flow by connecting their data lines. Simulink simulates analogue systems and discrete digital systems.

The Simulink model shown in Figure 3.13 is used to carry out simulation studies and analyze the performance of the controllers under different operating conditions.

4.1. Simulation of PID controller

The simulation result of the closed loop system step response of continuous and discrete time controllers is shown in Figure 4.1. For discrete time controller the process and sensor transfer function is changed to their equivalent discrete form for simulation study.

As it is observed from the figure 4.1 the discrete controller achieves better transient response than the continuous equivalent.

The discrete system transfer function model of a glass tempering furnace with PID controller is shown in Figure 3.3. This model was simulated in different PID tuning mechanisms using a MATLAB code presented in Appendix B.

The step response of the closed loop system with PID controller is shown in the Figure 4.2. Besides different tuning mechanism for PID controller like Ziegler-Nichols, Cohen-Coon, Integral of time-weighted absolute error (ITAE) both load point and set point changes and MATLAB/SIMULINK PID tuning is shown.

As it is observed from the figure, with the MATLAB/SIMULINK PID tuning extension tool box tuned PID controller the step response of the system gets decreased in percent overshoot & settling time, increased in rise time and the steady-state temperature error is zero. According to the design specifications, the allowable steady-state temperature error is ±5%. Therefore, the designed controller has met the desired specifications.

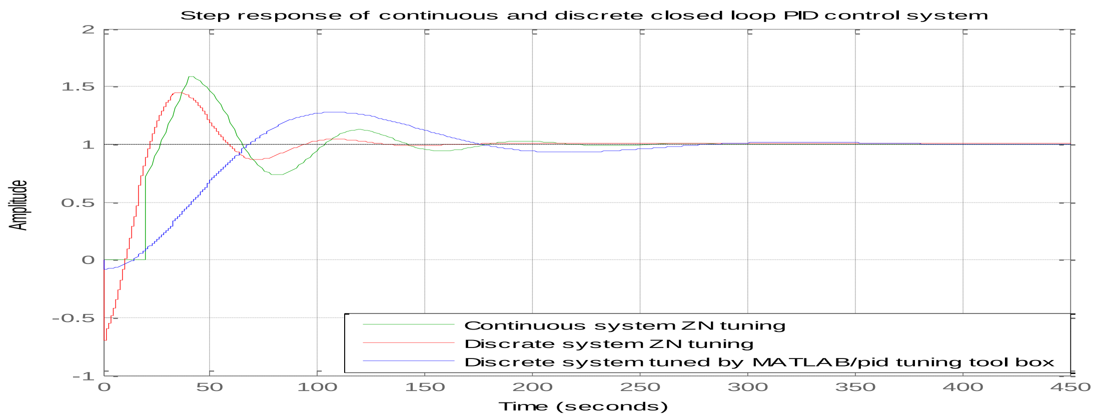


Figure 4. 1Step response of continuous and discrete closed loop PID control system

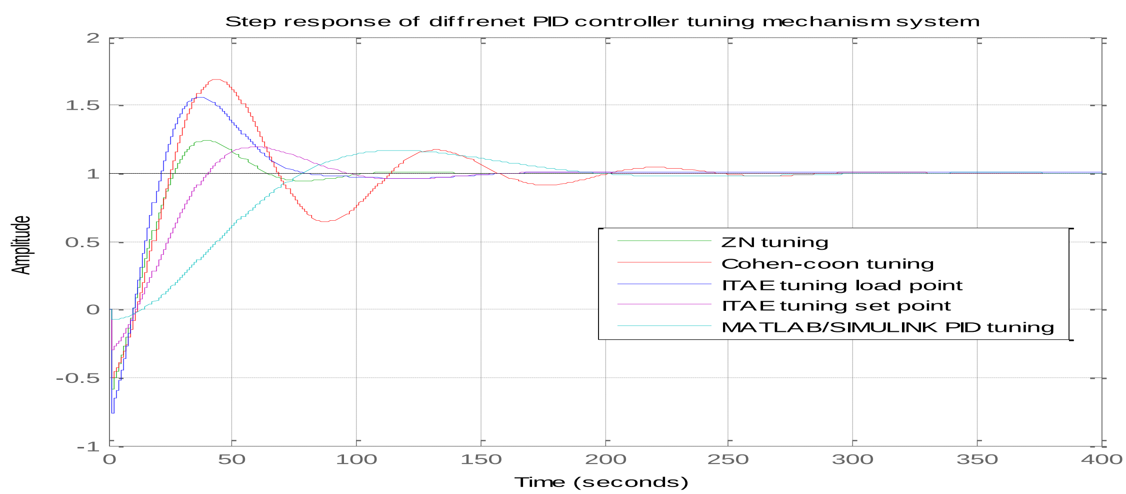


Figure 4. 2 Step response of closed loop system with different PID tuning mechanisms.

4.1.1 Comparison of different PID tuning mechanisms

Table 4.1 summarizes the continuous and discrete PID controllers tuned by Ziegler-Nichols tuning algorithm and MATLAB/ pid tuning extension tool box. , Table 4.1 summarizes the transient response comparison of PID controllers tuned by Ziegler-Nichols, Cohen-Coon, Integral of time-weighted absolute error (ITAE) both load point and set point changes and MATLAB/ pid tuning extension tool box.

	Rise time	Pick time	Settling time	Overshoot
Continuous Ziegler-Nichols	22s	42.1s	220s	58%
Discrete Ziegler-Nichols	21s	35.5s	150s	44%
MATLAB pid tuning extension tool	56s	112s	210s	16.129%

Table 4. 1 summarizes of continuous and discrete PID controller step response

Mechanism	Rise time	Pick time	Settling time	Overshoot
Ziegler-Nichols	23s	37.3s	110s	24%
Cohen-Coon	26s	50.2s	270s	69%
ITAE set point change	32s	61s	150s	19%
ITAE load point change	23.1s	37.3s	150s	56%
MATLAB pid tuning extension tool	56s	112s	210s	16.129%

Table 4. 2 Summarizes of PID controller step response with different tuning mechanism

As shown in figure 4.2 and table 4.2 the discrete controller is better in transient performance analysis. Because of glass tempering process needs the controller that has minimum overshoot, we use parameters of PID tuned by MATLAB pid tuning extension tool for our MATLAB/SIMULINK model and farther simulation studies.

4.2 Simulation result of PID and fuzzy PID Simulink model

The overall MATLAB/SIMULINK model of both fuzzy-PID and PID controllers for glass tempering furnace is shown in Figure 3.12. The figure shows the temperature response of the system for 620 degrees Celsius temperature input for PID controller and FPID controller.

In both case the system is stable, as it is observed from the figure FPID controller achieves better transient response than that of traditional PID controller

The simulation result is also similar to that of temperature control of furnace for glass tempering process presented in different literatures [25][28][31] Thus, the result is supported by different related works.

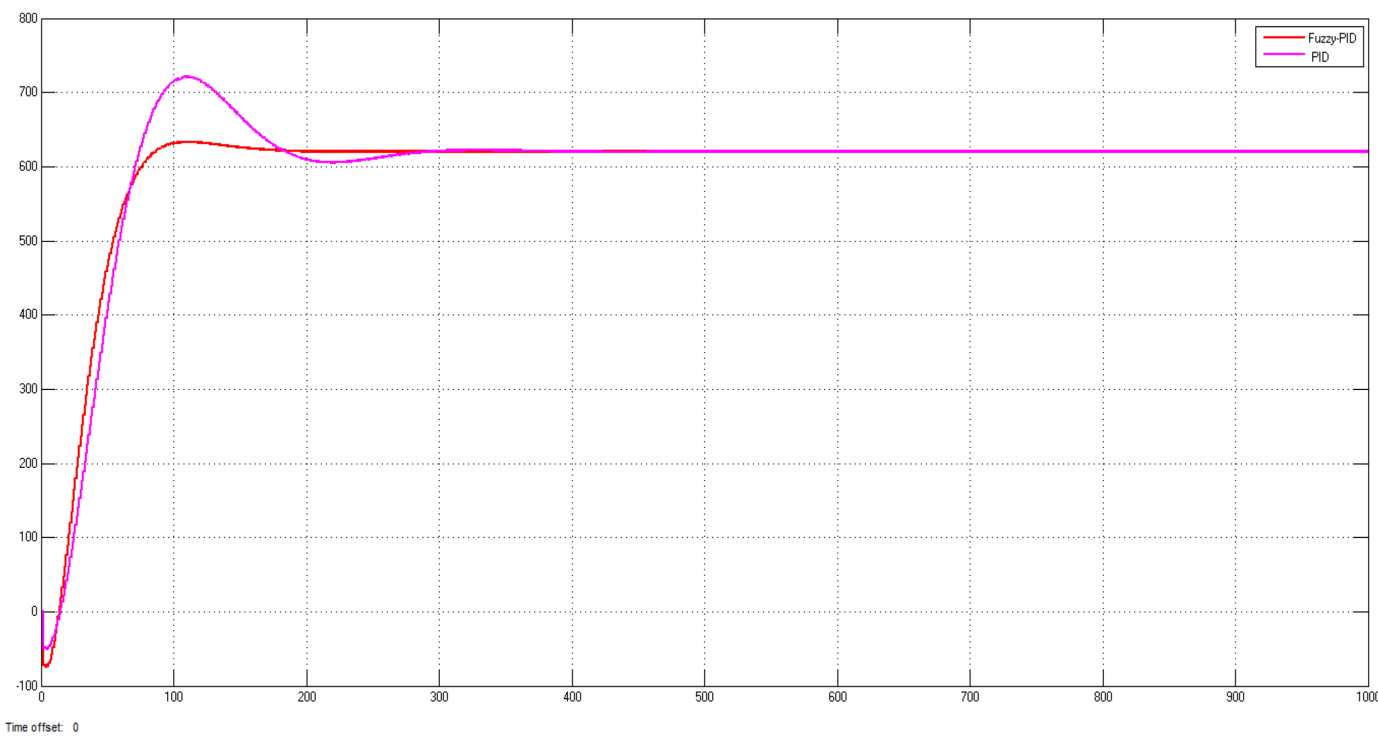


Figure 4. 3 Simulation results of FPID and PID controllers

4.2.1 Comparison of PID and FPID controller mechanisms

Table 4.3 Summarizes the comparison of PID controller with FPID controller

Transient parameters	Methods	
	PID	FPID
Pick time	110s	110.4s
Rise time	44s	45s
Settling time	290s	180s
Pick overshot	16.13%	2.129%

Table 4. 3 Step response of FPID and PID

In order to test the robustness, stability and effectiveness of the proposed fuzzy logic controller, different operating conditions are checked by parameter variation like delay time, time constant and DC gain. Figure 4.4 shows the simulation results of ±50 % delay time. As shown in the figure the pick overshoot, settling time and rise time is increased from the original system in both controllers. The proposed FPID controller has better transient performance than PID controller when the delay time is ±50 %.

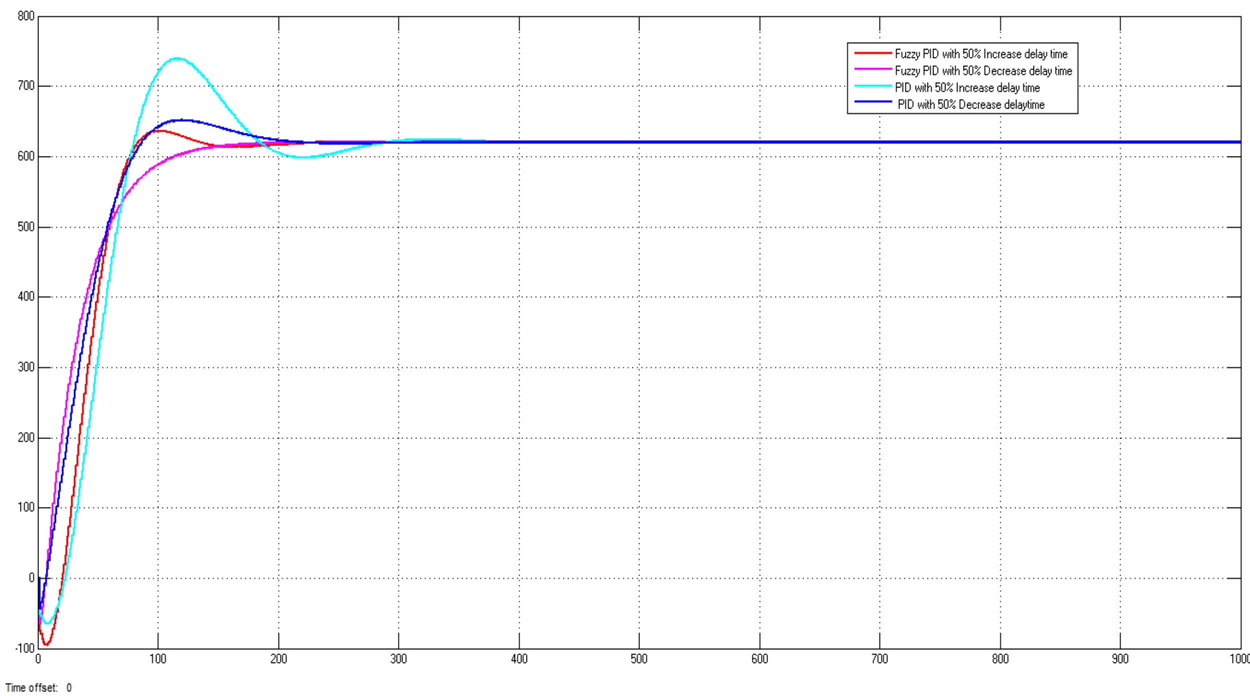


Figure 4. 4 Response of the PID and FPID controllers with ±50% delay time

Table 4. 4 Transient performance for ±50% delay time

	Rise time	Pick time	Settling time	Overshoot
Fuzzy PID with 50% increasing delay time	42s	101.2s	220s	2.6387%
Fuzzy PID with 50% decreased delay time	69s	-	180s	No overshoot
PID with 50% increasing delay time	39s	116s	360s	19.35%
PID with 50% decreased delay time	60s	119.4s	220s	5.083%

Figure 4.5 shows the results of simulation for ±50 % time constant and Figure 4.6 shows the results of simulation for ±50 % DC gain.

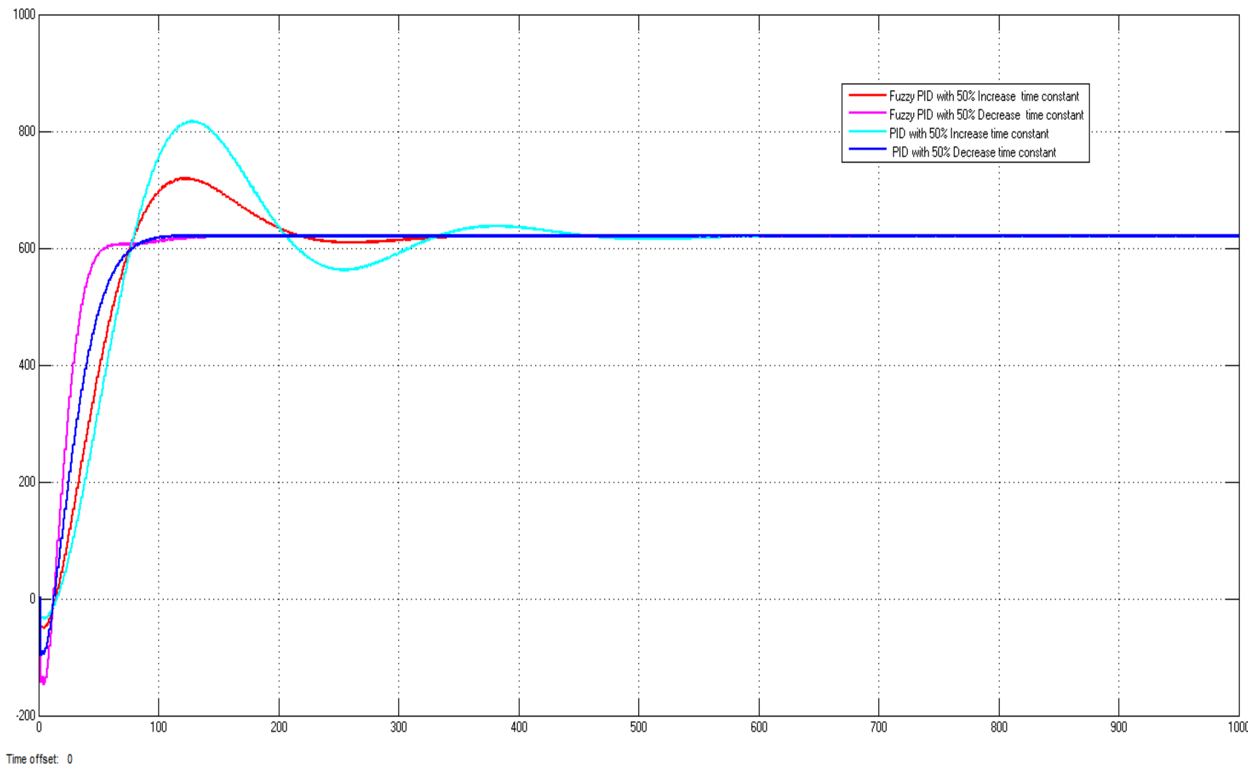


Figure 4. 5 Response of the PID and FPID controllers with ±50% time constant

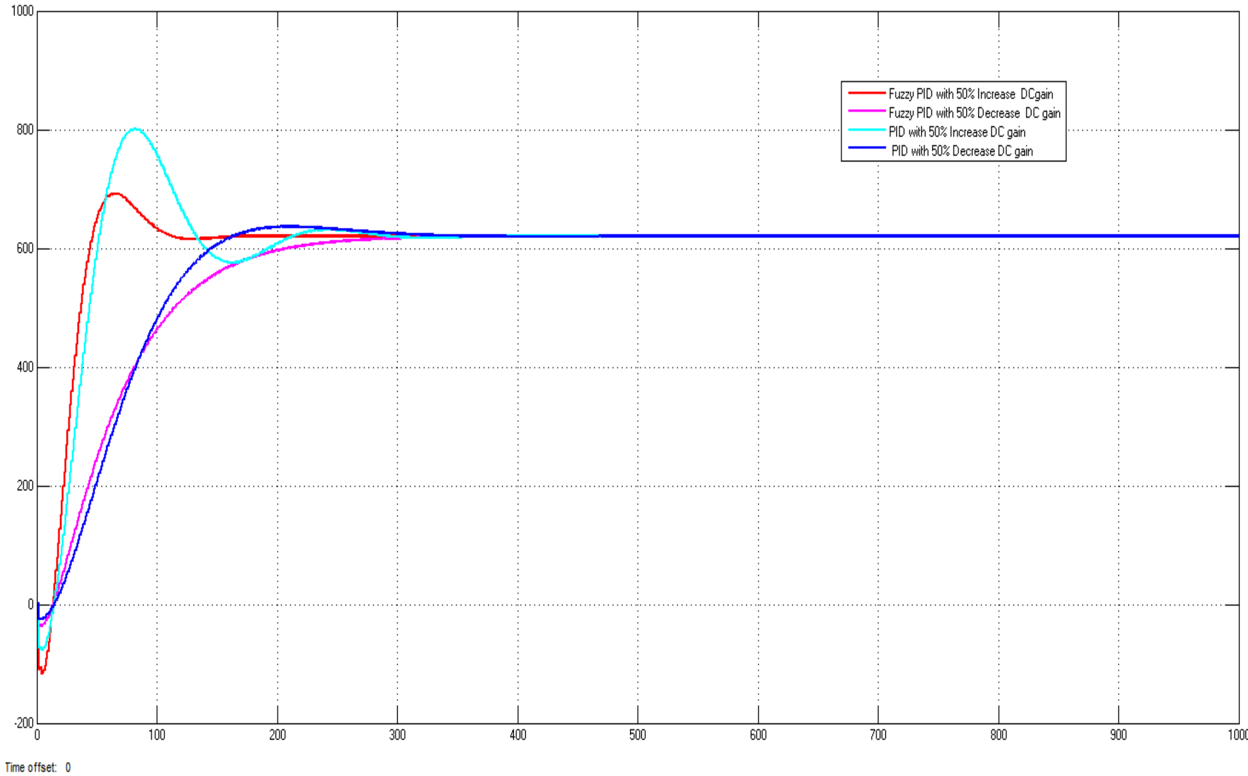


Figure 4. 6 Response of the PID and FPID controllers with ±50% DC gain

Table 4. 5 Transient performance for ±50% Time constant

	Rise time	Pick time	Settling time	Overshoot
Fuzzy PID with 50% increasing Time constant	45s	122s	340s	16.13%
Fuzzy PID with 50% decreased Time constant	28s	-	100s	No overshoot
PID with 50% increasing Time constant	52s	128s	450s	31.613%
PID with 50% decreased Time constant	46s	-	100s	No overshoot

	Rise time	Pick time	Settling time	Overshoot
Fuzzy PID with 50% increasing DC gain	24s	65s	160s	11.613%
Fuzzy PID with 50% decreased DC gain	127s	-	100s	No overshoot
PID with 50% increasing DC gain	29s	82s	290s	29.03%
PID with 50% decreased DC gain	97s	210.1s	320s	2.63%

Table 4. 6 Transient performance for ±50% DC gain

The variation of delay time, Time constant and DC gain has a significant effect on the transient Performances of both controllers. As shown in the figures the system is robust, effective and stable in ±50% parameter variations this shows that the capability of both PID and FPID controller to perform satisfactorily over a wide range of operating conditions. Hence the proposed FPID controller has better acceptance of parameter variation when compared to traditional PID controller.

The capability of the controllers in disturbance rejection is checked by adding step disturbance signal from signal builder that has magnitude of two as shown in figure 4.7 and control signal that is input to the plant as shown in figure 3.13 after settling time. As shown in Figure 4.6 fuzzy-PID controller has better disturbance rejection than PID.

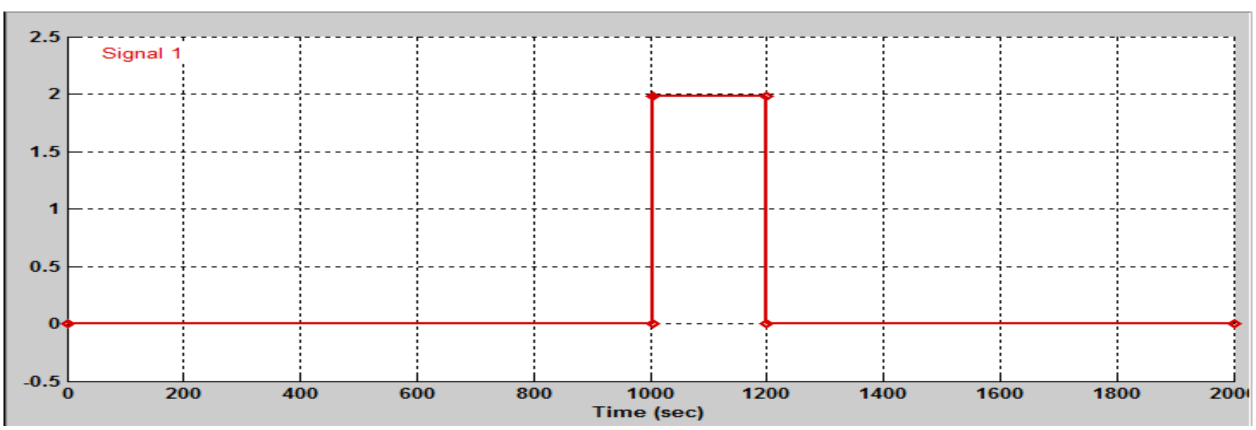


Figure 4. 7 Disturbance signal added to control signal after settling time

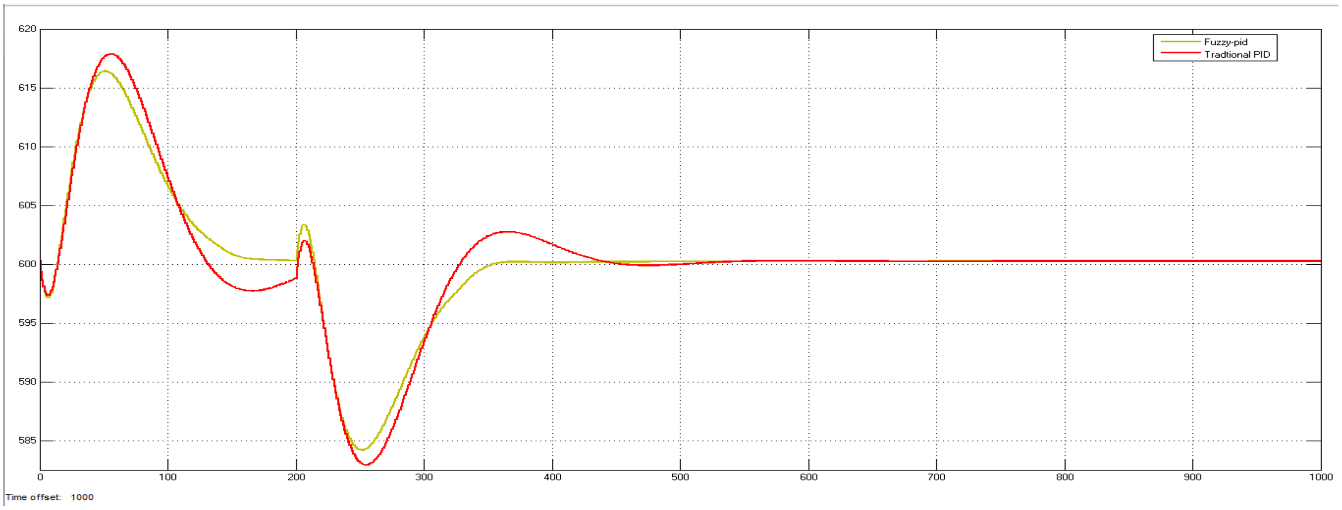


Figure 4. 8 Disturbance rejections of PID and FPID controllers

In general, the simulation results show that the proposed controller works efficiently for the whole range of temperature control of glass tempering furnace. Thus, the Fuzzy PID (FPID) controller has already performed well in transient as well as steady states.

CHAPTER FIVE

5 CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusion

In this thesis, temperature control of glass tempering resistance furnace is presented. Furnace is a nonlinear and time-varying complex system. After proper assumptions and approximations are made to linearize the mathematical equations describing the furnace, transfer function are obtained in frequency domain. The continuous time transfer function of the controlled plant and the PID controller is changed to its digital time equivalent. The sampling time is taken as 0.1 second. The results of traditional PID temperature controller are not satisfactory to the higher degree of accuracy condition. Even it introduces steady state error to the system, fuzzy control not only has the prominent advantage in complex, time-varying and nonlinear system control but also don't need the mathematical model of controlled object.

Fuzzy logic controller is proposed to tune the parameters of PID controller for temperature control of glass tempering resistance furnace, which has the advantages of PID control and fuzzy control. Both the transient and steady state performances of the controller are improved by increasing the number of membership functions

A model that represents the glass tempering resistance furnace was developed. The performance of the proposed fuzzy PID and PID controller is tested through simulation studies using MATLAB/SIMULINK. It is observed from the simulation results that the average overshoot is 2.129%, rising time 45 seconds and the settling time is 180 seconds with the proposed fuzzy PID controller while overshoot is 16.13%, rising time is 44 seconds and settling time is 290 seconds with traditional PID controller.

The proposed fuzzy PID controller is able to maintain the temperature of a glass tempering furnace within tolerable limits in spite of the parameter variation like delay time, time constant and DC gain. Furthermore, the capability of disturbance rejection is cheeked by giving step signal after settling time and it founded better than traditional PID.

The shape and distribution of the membership functions used in the design of fuzzy controller should be chosen with care to achieve system stability and robustness.

5.2 Recommendation

For fuzzy interface system the rule base and scale factor can be optimized with genetic algorithm for further improvements in the performance of temperature control of glass tempering process. Moreover, the hardware of self-tuning fuzzy PID controller/ fuzzy supervisor PID controller can be developed for temperature control of glass tempering electric furnace.

6 REFERENCES

[1] Ford M. "How is tempered glass made? In Scientific American"; America, 2001.

[2] Jonathan Barr, "Understanding the Glass Tempering Process", Germany, 2003.

[3] Andersen Windows "glass making process, and review the tempering process." Stillwater, Minnesota, America.

[4] Rantala, M. "Heat Transfer Phenomena in Float Glass Heat Treatment Processes (Tampere University of Technology. Publication; Vol. 1355)". Tampere University ;2015

[5] Gilbert Rancoule., "Glass handling in tempering furnaces", glass worldwide magazine-july/august, 2011.

[6] Jeroen Kuenen, "Manufacture of glass and glass products", Guidebook, 2013

[7] Berkeley, Ajax Electric Company, "Improving Process Heating System Performance" Second Edition; Lawrence Berkeley National Laboratory , California, 2007.

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

[9] Diwakar T. Korsane[1], Vivek Yadav[2], Kiran H. Raut[3], "PID Tuning Rules for First Order plus Time Delay System", International Journal of Innovative Research In Electrical, Electronics, Instrumentation And Control Engineering; Vol.2, Issue 1, January 2014

[10] K. J. Astrom and T. Haglund, "Automatic Tuning of PID Controllers", 1st ed. Research Triangle Park, NC: Instrum. Soc. Amer., 1988.

[11] K. J. Astrom and T. Haglund, PID Controllers: Theory, Design and Tuning, 2nd ed. Research Triangle Park, NC: Instrum. Soc. Amer.,1995

[12] J. G. Ziegler and N. B. Nichols, "Optimal settings for automatic controllers," Trans. ASME, vol. 64, pp. 759–768, 1942.

[13] "Feedback Control of Linear SISO systems"; Process Dynamics and Control.

[14] J. C. Basilio and S. R. Matos; "Design of PI and PID Controllers With Transient Performance Specification"; IEEE Transactions on Education, Vol. 45, No. 4, November 2002

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

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

[17] L.A. Bryan and E.A. Bryan, "Programmable Controllers Theory and Implementation", Second Edition, An Industrial Text Company Publication , Atlanta ,Georgia, USA,1997.

[18] Walter Banks / Gordon Hayward, "Fuzzy Logic in Embedded Microcomputers and Control Systems", Burlingame, CA, July 1993.

[19] L.A. Zadeh, "Fuzzy Sets," Inform. Contr., vol. 8, pp. 338-353, 1965

[20] Ian S. Shaw (1998), Fuzzy Control of Industrial Systems: Theory and Applications, Kluwer Academic Publishers; ISBN: 0792382498

[21] M. Godoy Simões and M. Friedhofer, "An implementation methodology of a fuzzy based decision support algorithm," International Journal of Knowledge-Based Intelligent Engineering Systems, October 1997, vol.1 no. 4, pp. 267-275.

[22] M. Godoy Simoes, N. N. Franceschetti, "Fuzzy optimisation based control of a solar array system" IEE Proceedings - Electric Power Applications, September 1999, Vol. 146,Issue 5, pp. 552-558

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

[24] Real J. Fradette "Understanding PID Temperature Control as Applied to Vacuum Furnace Performance," Senior Technical Consultant, Solar Atmospheres, Inc, 1996.

[25] Naif B . Almutairi and Mohamed Zribi, Sliding mode controllers for a tempered glass, ISA Transactions, Nov 5, 2015.

[26] Peyman Nazarian, Feedback Linearization based control of a horizontal radiative furnace, Proceedings of the 9th WSEAS International Conference on Robotics, Control and Manufacturing Technology, 2010.

[27] 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, No. 3, 2012.

[28] Yu han, Jia Jinling ,Chen Guangjian and Chen Xizhen, "Temperature Control of Electric Furnace Based on Fuzzy PID", International Coriference on Electronics and Optoelectronics , 2011.

[29] Dr. K Babulu and D. Kranthi Kumar, "Fuzzy Self-Adaptive PID Controller Design for Electric Heating Furnace", International Journal of Engineering Inventions,ISSN: 2278-7461, www.ijeijournal.com,Volume 1, Issue 5 ,September2012.

[30] SHI Dequan*, GAO Guili, GAO Zhiwei and XIAO Peng, "Application of Expert Fuzzy PID Method for Temperature Control of Heating Furnace", International Workshop on Information and Electronics Engineering (IWIEE), 2012.

[31] Shoubin Wang , Na Li b and Fan Yang, "Resistance Furnace Temperature System on Fuzzy PID Controller" Journal of Information & Computational Science 9: 9 , 2012.

[32] G. Shabib and K. Hassan, "Fuzzy Logic Control of Three Phase Submerged Arc Ferrosilicon Furnace", Journal of Information & Computational Science 9: 9, 2012.

[33] Radhika .S .B and Prof. Nasar .A, "Internal Model Control based Preheating Zone Temperature Control for Varying Time Delay Uncertainty of the Reheating Furnace", International Journal of Science and Research (IJSR),ISSN (Online): 2319-7064 Index Copernicus Value ,2013.

[34] BABATUNDE A. OGUNNAIKE and W. HARMON RAY, "process dynamics, modeling, and control", New York Oxford OXFORD UNIVERSITY PRESS, 1994.

[35] http://www.engineeringtoolbox.com/air-heating-systems-d_1136.html

[36] McGraw, “Introduction to Thermodynamics and Heat Transfer”, chapter two heat equation and part two transfer, Second Edition, Hill 375 Companies,2008.

[37] YOON and Heenam, Medical Physics, “Dynamic Temperature Measurement”, 2012.

[38] Process Dynamics, Operations, and Control;: Heated Tank, Lesson 5, spring 2006.

[39] Mr. Tushar Upalanchiwar ,Prof.A.V.Sakhare, “Design and implementation of the fuzzy PID controller using MATLAB/SIMULINK model”, International Journal of Research in Computer and Communication Technology, Vol 3, Issue 3, March- 2014.

[40] Mahlet Legesse ,“Speed Control of Vector Controlled PMSM Drive using Fuzzy Logic-PI Controller” ,Addis Ababa Institute of Technology (AAiT), August, 201.

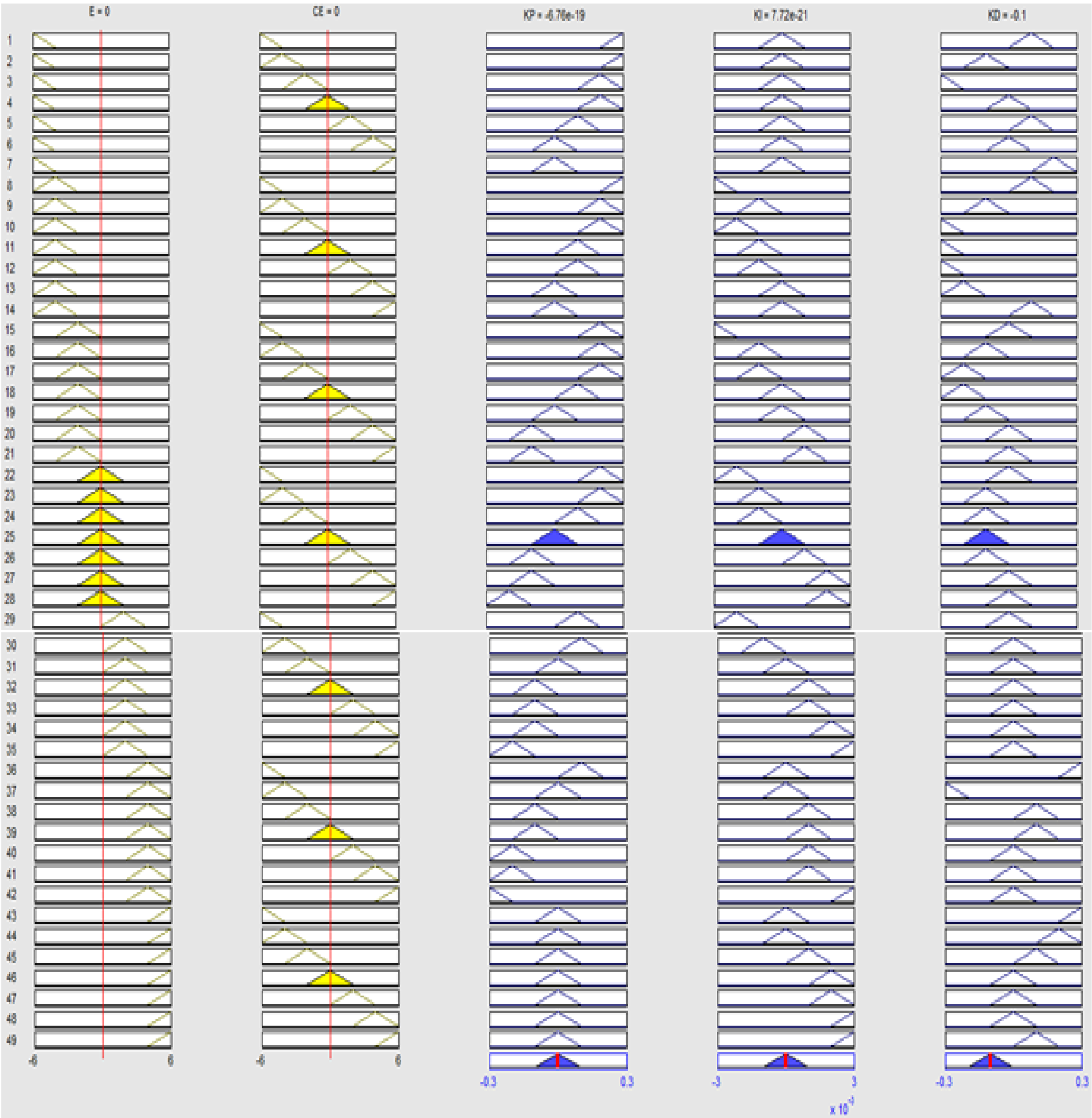
[41] Chengwu Lina, Miao Liu & Jianguang Zhu, “Research on Fuzzy Adaptive PID Control Algorithm Based on Siemens PLCSIM”, 3rd International Conference on Electric and Electronics (EEIC), Shenyang University of Technology,Shenyang, China ,2013.

[42] Zhen-Yu Zhao, Malayoshi Tomikzuka & Satoru Isaka, “Fuzzy Gain Scheduling of PID Controllers”, Article in IEEE Transactions on Systems Man and Cybernetics, October 1993.

7 APPENDICES

Appendix A

Rule viewer of fuzzy interface system



Appendix B

MATLAB code for PID controller

```
%%MATLAB code for different tuning system PID controller
s=tf('s');
TF=exp(-20*s)*13.07/(25*s+1); % model of the furnace to be controlled
H=1/(10*s+1);%transfer function of sensor (feedback)
T=25;
D=20;
K=13.07;
z=tf('z');
p=pade(TF,1);
pd=c2d(p,1);
hd=c2d(H,1);
%% System step response using PID controller 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')
hold on;
%% System step response using 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));
KIc=KCc/TIc;
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;
%% System step response using PID controller ITAE load setting tuning
kcl=0.1282;
til=25.1838;
KIl=kcl/til;
tdl=11.8925;
kdl=kcl*tdl;
CITAEL=kcl+KIl*(z/(z-1))+kdl*((z-1)/z);
SYSITAEl=feedback(CITAEL*pd,hd);
step(SYSITAEl,'B')
hold on;
%% System step response using continuous time PID controller ITAE set point
tuning
kcs=0.08925;
tis=36.8297;
KIs=kcs/tis;
tds=6.2575;
kds=kcs*tds;
CITAEs=kcs+KIs*(z/(z-1))+kds*((z-1)/z);
SYSITAEs=feedback(CITAEs*pd,hd);
step(SYSITAEs,'m')
hold on;
```

```
%% System step response using PID controller matlab tuning
kcm=0.03426;
KIm=0.0016726;
kdm=0.11415;
Cm=kcm+KIm*(z/(z-1))+kdm*((z-1)/z);
SYSITAEm=feedback(Cm*pd,hd);
step(SYSITAEm,'c');
hold on;
legend('ZN tuning ','Cohen-coon tuning','ITAE tuning load point','ITAE tuning
set point','MATLAB/SIMULINK PID tuning');
title('Step response of different PID controller tuning mechanism system');
grid on;
```