UNIVERSITI
TEKNOLOGI
PETRONAS

Modeling and simulation of PI-Fuzzy Control on Gas Pilot Plant via Matlab/Simulink and Industrial PC

By

Naveed Ahmed Memon

7161

Submitted to the Electrical & Electronics Engineering Programme
in Partial Fulfillment of the Requirements
for the Degree
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Universiti Teknologi Petronas
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Modeling and simulation of PI-Fuzzy Control on Gas Pilot Plant via Matlab/Simulink and Industrial PC

by

Naveed Ahmed

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved:

Dr Norddin saad
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

April 2008

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

Naveed Ahmed Memon

ABSTRACT

PID controllers are most commonly used in the industrial plants because of their good performance and easiness of use. Due to growth technology, new advanced techniques have come into existence either to produce new things or to improve existing ones. Several techniques can be used but having advanced control system, we are looking forward to design system which behaves in the same way as to what human do, and this is known as artificial intelligence. Artificial intelligence techniques have been used to convert human experience into a form understandable by computers. Intelligent systems are usually describes by analogies with biological systems, for example looking at human beings perform control tasks, recognize patterns, or make decisions. Notably fuzzy logic emerged as a tool to deal with uncertain, imprecise or qualitative decision making problem, controllers that combine intelligent and conventional techniques are commonly used in the intelligent control of complex dynamic systems. In this work the pilot fuzzy control used to improve existing traditional control systems by adding and an extra layer of intelligence to current control method to make it more efficient, the focus of this work is to design, analysis and implementation of PI-fuzzy control on a gas pilot plant using industrial PC.

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my deepest gratitude to Allah the Almighty for the courage to put my plucky efforts in the Final Year Project. I would like to thank to following individuals or groups for giving me moral support through out the semester.

- Dr. Norddin saad (Final Year Project Supervisor)
- Final Year project Committee
- Resource Center of University Technology PETRONAS
- Mr. Azhar (Laboratory Assistant)
- University Technology PETRONAS lecturers.

Lastly I would like to thank all those who were involved directly or indirectly throughout this Final Year Project and in completing this final report. Thanks for your kindness and may Allah bless you all.

TABLE OF CONTENTS

LIST OF TABLES vii

LIST OF FIGURES viii

LIST OF ABBREVIATIONS viii

CHAPTER 1 INTRODUCTION 1

- 1.1 Background of Research 1
- 1.2 Problem Statement 4
- 1.3 Objectives 4
- 1.4 Scope of Study 4

CHAPTER 2 LITERATURE REVIEW 6

- 2.2 Basic elements in process control loop 8
- 2.3 Pressure control loop 9
- 2.4 Emprical Modelling 10
- 2.5 Process reacation curve 11
- 2.6 Open loop test 12
- 2.7 Closed loop test 13

CHAPTER 3 METHODOLOGY 14

- 3.1 Flow of Project 14
  - 3.1.1 Experimental Design 14
  - 3.1.2 Plant Experiment 15
  - 3.1.3 Determining Model Structures 15
  - 3.1.4 Parameter Estimation 15
  - 3.1.5 Diagnostic Evaluation 15
  - 3.1.6 Verification 16
  - 3.17 Plant investigation 16
  - 3.18 Plant Data 17
- 3.2 Fuzzy Controller 18
  - 3.2.1 Fuzzy control design 20
  - 3.2.2 Fuzzifcation/Defuzzification 21
  - 3.2.2 Shape of membership function 22
  - 3.2.3 Interference and compostion 22
  - 3.2.4 FIS Editor 23

3.2.5 Membership functions..23

3.2.6 Rule viewer and Editor..24

3.2.7 Surface viewer.... 25

3.2.8 Fuzzy logic PI controller 27

3.3 Tools and software..27

3.3.1 Matlab and Simulink..... 27

3.3.2 Matlab Simulink Xp Target...28

CHAPTER 4 RESULTS.. 30

4.1 Process Reaction Curve (Open Loop Method)..............................30

CHAPTER 5 DISCUSSIONS AND FINDINGS.. 33

CHAPTER 6 CONCLUSIONS.. 34

REFERENCES...35

APPENDICES..37

Appendix I ACRONYMS..38

Appendix II Figures..42

LIST OF TABLES

Table 2.1 Fuzzy rulesfor controller 6
Table 2.2 Conversion Table......6
Table 3.1 Ziegler Nicholas closed loop tuning constants......10

LIST OF FIGURES

Figure 1.1 A Block diagram of PID Controller 3

Figure 2.1 Fuzzy Structure 3

Figure 2.2 Block diagram showing the elements of a process control loop 3

Figure 2.3 Simplified Pressure Diagram for Gaseous Pressure Plant 3

Figure 2.4 Block Diagram of the pressure plant 3

Figure 3.1 Plant Block diagram 3

Figure 3.2 Bode plot 3

Figure 3.3 PID Controller 10

Figure 3.4 Block diagram of Fuzzy controller 10

Figure 3.5 Membership Curves 10

Figure 3.6 Fuzzy Algorithm 11

Figure 3.7 Fuzzy FIS Editor 12

Figure 3.8 Fuzzy Membership Function 12

Figure 3.9 Fuzzy Rule Editor 13

Figure 3.10 Fuzzy Rule Viewer 13

Figure 3.11 Fuzzy Surface Viewer 14

Figure 3.12 Fuzzy Logic Controller 14

Figure 3.13 Simulink with Fuzzy controller 14

Figure 3.14 Block Diagram using Simulink Xpc Target 14

Figure 3.15 Block diagram of xPC Target box 14

Figure 3.16: Default ideal PID using difference error approach. 14

Figure 4.1: A matlab/simulink model of the plant 14

Figure 4.2: Output graph of PI(D) controller 14

Figure 4.2: A matlab/Simlink model of the plant with fuzzy controller 14

Figure 4.2: Output graph of PI-Fuzzy controller 14

LIST OF ABBREVIATIONS

ABV	Actuator Ball/Block Valve
ASD	Alarm and Shutdown Equipment
ANSI	American National Standards Institute
API	American Petroleum Institute
barg	Bar gauge (pressure)
CCR	Central Control Room
DOSH	Department of Occupational Safety and Health. Formerly known as Factory and Machinery Department (FMD)
ESD	Emergency Shutdown with Venting
FCV	Flow Control Valve
FI	Flow Indicator
FLC	Fuzzy Logic Controller
FR	Flow Recorder
FRC	Flow Recorder Controller
FT	Flow Transmitter
HP	High Pressure
LP	Low Pressure
K	Compressor
LC	Level Controller
LCV	Level Control Valve
LCP	Local Control Panel
LI	Level Indicator
LIC	Level Indicating Controller
LR	Level Recorder

LT	Level Transmitter
M	Motor
MV	Manipulated Variable
NC	Normally Closed
NO	Normally Open
P	Pumps for Equipment, Hydrocarbon for Piping
PC	Pressure Controller
PC	Personal Computer
PCV	Pressure Control Valve
PV	Process Variable
PR	Pressure Recorder
psig	Pound per square inch gauge (pressure)
RV	Relief Valve
SP	Set Point
V	Vessel
UTP	University Technology PETRONAS

CHAPTER 1
INTRODUCTION

1.1 Background of Research

Plants consist of number of control loops. Each of these control loops is designed to keep some important process variable such as pressure, flow, level, temperature etc. within a required operating range to ensure the quality of the end product. In this project the author is going to control the pressure of process by combing some attributes of PID controller and Fuzzy logic to modify it.

A proportional-integral-derivative controller (PID controller) is a generic control loop feedback mechanism widely used in industrial control systems. A PID controller attempts to correct the error between a measured process variable and a desired setpoint by calculating and then outputting a corrective action that can adjust the process accordingly.

The PID controller calculation (algorithm) involves three separate parameters; the Proportional, the Integral and Derivative values. The Proportional value determines the reaction to the current error, the Integral determines the reaction based on the sum of recent errors and the Derivative determines the reaction to the rate at which the error has been changing. The weighted sum of these three actions is used to adjust the process via a control element such as the position of a control valve or the power supply of a heating element. By "tuning" the three constants in the PID controller algorithm the PID can provide control action designed for specific process requirements. The response of the controller can be described in terms of the responsiveness of the controller to an error, the degree to which the controller overshoots the setpoint and the degree of system oscillation. Note that the use of the PID algorithm for control does not guarantee optimal control of the system. Some applications may require using only one or two modes to provide the appropriate system control. This is achieved by setting the gain of undesired control outputs 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 particularly common, since derivative action is very sensitive to measurement noise, and the absence of an integral value prevents the system from reaching its target value due to the control action. The PID control scheme is named after its three correcting terms, whose sum constitutes the manipulated variable (MV). Hence:

$$\mathrm{MV(t)} = P_{\mathrm{out}} + I_{\mathrm{out}} + D_{\mathrm{out}}$$

where P_{out}, I_{out}, and D_{out} are the contributions to the output from the PID controller from each of the three terms, as defined below.

Proportional term

The proportional term makes a change to the output that is proportional to the current error value. The proportional response can be adjusted by multiplying the error by a constant K_p, called the proportional gain.

The proportional term is given by:

$$P_{\mathrm{out}} = K_p e(t)$$

Integral term

The contribution from the integral term is proportional to both the magnitude of the error and the duration of the error. Summing the instantaneous error over time (integrating the error) 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 magnitude of the contribution of the integral term to the overall control action is determined by the integral gain, Ki.

The integral term is given by:

$$I_{\mathrm{out}} = K_i \int_0^t e(\tau)\, d\tau$$

Derivative term

The rate of change of the process error is calculated by determining the slope of the error over time (i.e. its first derivative with respect to time) and multiplying this rate of change by the derivative gain Kd. The magnitude of the contribution of the derivative term to the overall control action is determined the derivative gain, Kd.

The derivative term is given by:

$$D_{\text{out}} = K_d \frac{de}{dt}$$

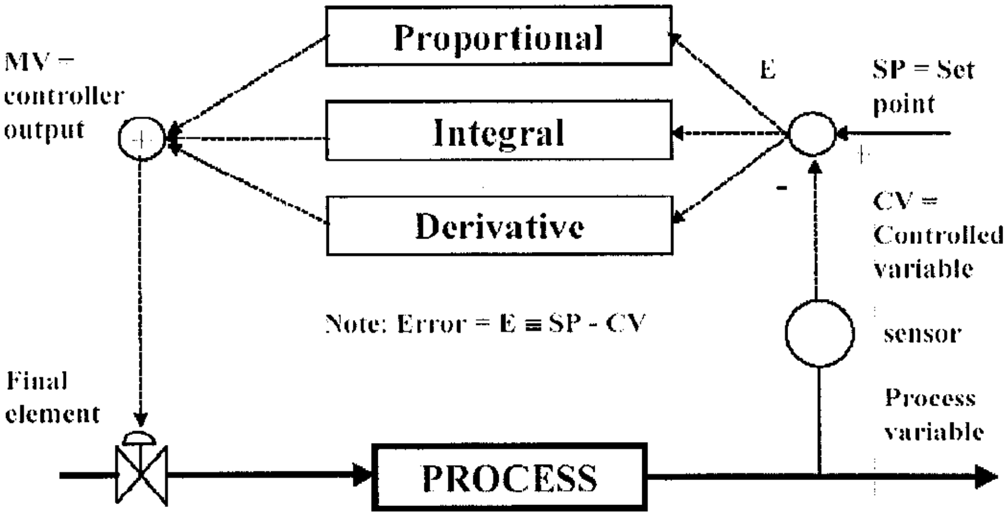


Fig 1.1: A block diagram of a PID controller

But in the last few years the application of artificial intelligence techniques have been used to convert human experience techniques into a form understandable by computers, e.g. fuzzy control system is a control system based on fuzzy logic - a mathematical system that analyzes analog input values in terms of logical variables that take on continuous values between 0 and 1, in contrast to classical or digital logic, which operates on discrete values of either 0 and 1.

The input variables in a fuzzy control system are in general mapped into by sets of membership functions similar to this known as fuzzy sets, the process of converting a crisp input value to fuzzy value is called fuzzification.

The most common shape of membership functions is triangular, although trapezoid and bell curves are also used, but shape is generally is less important than the number of curves and their placement. From three to seven curves are generally appropriate to cover required range of value.

1.2 Problem Statement

- The purpose of this work is to develop a model of process with Matlab/Simulink tool that would be used for investigation of the effectiveness of several control strategies towards effective control
- The development of model will allow observation of the controller's action and changes in variables with a view that the industrial process can directly be manipulated and controlled.
- Particularly in this work, a PI-fuzzy controller is developed and implemented in real time, so our main focus on development of new controller that can be used for further analysis particularly optimizing the pressure process. The purpose of this model it to investigate the system and based upon that we can design Fuzzy controller.

1.3 Objectives

- To model and simulate the pressure Gas pilot plant via matlab/simulink and Industrial PC.
- To design the controller to improve the plant operation by Fuzzy logic.

1.4 Scope of Study

The modeling and simulation will be done on UTP's pressure pilot plant. The study will be based on input and output of the pressure together with its control action. A fuzzy logic system will be incorporated together with a PI-control to make it more efficient, an analysis of the behavior of the plant has been done by fuzzy logic curves for decision making process.

To validate the model, MATLAB Simulink is used to design Fuzzy controller then this controller output will be compared with already existing PID controller, after doing comparison, we'll try to optimize our controller to improve its working, after improving we'll implement on real time plant by excluding PID controller instead of Fuzzy controller.

The model accuracy is verified using the MATLAB/SIMULINK. Comparison is made with the real time data. The controller performance is tested and compared. The PI controller was used which very common in industry.

CHAPTER 02

LITERATURE REVIEW

Fuzzy logic imitates the logic of human thought, which is much less rigid than the calculations computers generally perform. Consider the task of driving a car. As you drive along, you notice that the stoplight ahead is red and the car in front of you is braking. Your (very rapid) thought process might be something like this: "I see that I need to stop. The road is wet because it's raining. The car is only a short distance in front of me. Therefore, I need to apply significant pressure to the brake pedal immediately." This reasoning takes place subconsciously, of course, but that's the way our brains work—in fuzzy terms.

Human brains do not base such decisions on the precise distance to the car ahead or the exact coefficient of friction between the tires and the road, as an embedded computer might. Likewise, our brains do not use a filter to derive the optimal pressure that should be applied to the brakes at a given moment. Our brains use common-sense rules, which seem to work pretty well.

Let's think about how a fuzzy cruise control system might work. The cruise controller maintains a constant vehicle speed in spite of neverending changes in road grade, wind resistance, and other variables. The controller does this by comparing the commanded speed with the actual speed. We can call the difference between commanded and actual *speed current error*. The *error change* is the difference in error from one sample period to the next.

If the current error is a small positive number—vehicle speed is slower than commanded—the controller needs to slightly increase the throttle angle in order to speed up the vehicle appropriately.

If both current error and error change are positive, the vehicle is going too slowly and decelerating. In this case, the controller needs to increase the throttle angle by a larger amount to achieve the desired speed.

Δe e	NB	NS	Z	PS	PB
NB	NB	NB	NS	Z	Z
NS	NB	NS	NS	Z	Z
Z	NB	NS	Z	PS	PB
PS	Z	Z	PS	PS	PB
PB	Z	Z	PB	PB	PB

NB=Negative big NS=Negative small Z=Zero
PS=Positive small PB=Positive big

	LN	MN	SN	Z	SP	MP	LP
LN	*LP*	*LP*	*LP*	*MP*	*MP*	*SP*	*Z*
MN	*LP*	*MP*	*MP*	*MP*	*SP*	*Z*	*SN*
SN	*LP*	*MP*	*SP*	*SP*	*Z*	*SN*	*MN*
Z	*MP*	*MP*	*SP*	*Z*	*SN*	*MN*	*MN*
SP	*MP*	*SP*	*Z*	*SN*	*SN*	*MN*	*LN*
MP	*SP*	*Z*	*SN*	*MN*	*MN*	*MN*	*LN*
LP	*Z*	*SN*	*MN*	*MN*	*LN*	*LN*	*LN*

Table 2.1: Fuzzy rules for the controller

Such specifications are called *fuzzy rules.* Table 2.1 shows a set of fuzzy rules that might be used for a controller. The controller inputs are the variables current error and error change. The output represents a fuzzy specification of how much to change the throttle, so these rules vary and depend upon system to system, now to implement them it is highly important for us to understand its behaviour survey of its functioning.

Fig 2.1: Fuzzy structure

2.2 Basic Elements in the Process Control Loop

The selected process control systems consist of a control loop having four main components:

- A Pressure Transmitter measures the pressure of the gas.
- A controller calculating an action based on measured value against a preset or desired value (set point).
- An output signal resulting from the controller calculation, which is used to manipulate the process action through some form of actuator.
- The final action or corrective action is by the Pressure Control valve (PCV)

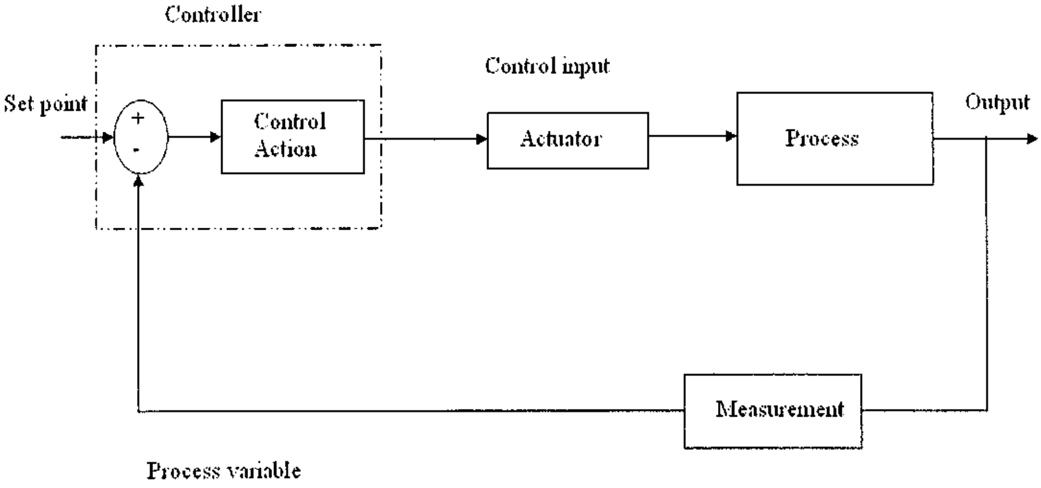


Figure 2.2: Block Diagram showing the elements of a process control loop

2.3 Pressure Control Loop

The study is carried out on the gas plant which is located in University Technology TRONAS Process laboratory. The pressure control loop is shown below:

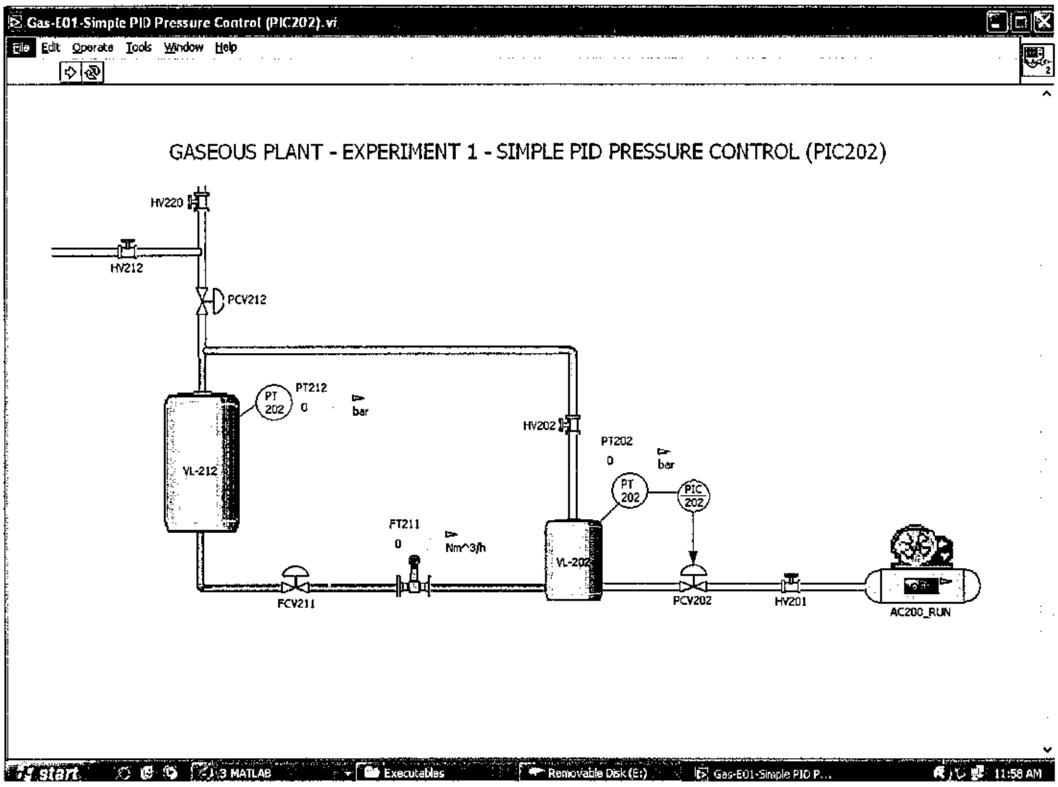


Figure 2.3: Simplified Pressure Diagram for Gaseous Pressure Plant

This pressure plant consists of gas vessel VL212, Pressure controller PIC 212, fail open diaphragm valve and pressure sensor.

The task is to regulate the flow from vessel VL212. The manipulated variable is the flow to the vessel VL212 that is to be adjusted to manipulate the desired pressure.

The pressure transmitter (PT212) sends the measured pressure signals to the pressure controller (PIC 212) which compares the measured pressure with the desired pressure set-point and then sends the signal to the valve (PCV 202). The PCV 202 will step up or down to any change of the flow rate that affects the pressure in the vessel.

Table 2.2: Conversion Table

Current (mA)	**Pressure (psig)**	**Percentage (%)**
4	3	0
8	6	25
12	9	50
16	12	75
20	15	100

The control block diagram is developed for the feedback control loop. The elements in control loop are combined and the closed transfer function is obtained using the figure 2.3 below

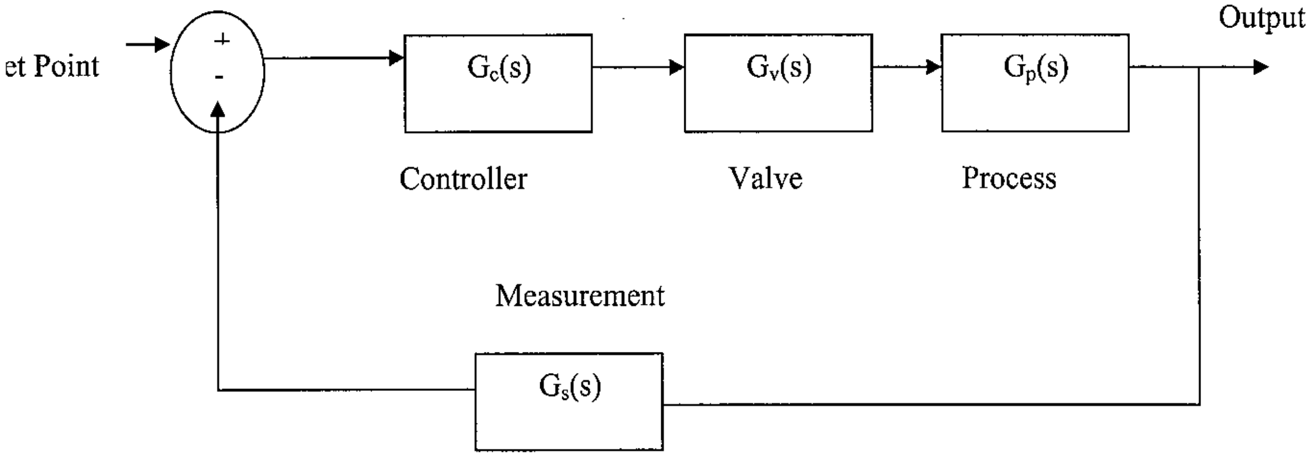


Figure 2.4: Block Diagram of the pressure plant

2.4 Empirical Modeling

To succeed in process control the designer must first establish a good understanding of the process to be controlled. Since it is not advisable to become too deeply involved in chemical or process engineering, we need to find a way of simplifying the representation of the process that we wish to control. This is done by adopting a technique modeling and the idea is to use empirical modeling.

Empirical modeling is the alternative to the mathematical modeling for the process control system. Empirical modeling provides the dynamics of the system (i.e. dynamic behavior of the system between the selected inputs and outputs).

In empirical model building two common methods used to design the experiments. The perturbation to the system is the step response in this project to generate the dynamic response. [3].

2.5 Process Reaction Curve

The process reaction curve method involves the following four actions:

- Allow the process to reach the steady state.
- Introduce the perturbation which is the perfect step input.
- Collect the process data (input and output response data) until the response reaches the steady state.
- Perform the calculations on the process reaction curve.

The graphical calculations determine the parameters for the first order with dead time. The process reaction curve is restricted to the first order and the transfer function from the process data is easily achieved. [3].

$$(t)\delta[1-e^{-(t-\theta)/t}]$$

The differential equation then transformed into transfer function given below.

$$\frac{X(s)}{Y(s)} = \frac{K_p\, e^{-\theta s}}{\tau s+1}$$

Based on the Ziegler and Nichols) method, there are two graphical techniques that can be used [3].

2.5.1 Method I for Process Reaction Curve

The values determined from the graph are

- The magnitude of change in input,δ
- The magnitude of change in output,Δ
- The maximum slope of output versus time, S
- The intercept of maximum slope with initial values, also called Dead timeθ.

The values from the plot are used to get the transfer function for the plant.
The maximum slope occurs at $t=\theta$ and $s= \Delta/\tau$. The model parameters can be calculated as follows: [3].

$K_p = \Delta / \delta$

$\tau = \Delta / s$

θ = intercept of maximum slope with initial value

2.5.2 Method II for Process Reaction Curve

Method II involves the calculation of model parameters when the output reaches the 28 and 63 percent of its final value. Any two values are picked where the transient response is changing rapidly so that the model parameters can be calculated accurately.
The process gain is the same as Method I but the other parameters calculated as follows:

$\tau = 1.5\,(t_{63\%} - t_{28\%})$ $\theta = t63_{\%} - \tau$

2.6 Open Loop Test

The open loop test is also called the manual test or manual control. The measured value of process variable is displayed to the operator and he has the ability to manipulate the control valve while sitting in the control room and sending signal to control valve. The purpose of open loop is to evaluate the process reaction curve for the model. For the process reaction it involves the following actions:

- First allow the reaction to get steady state
- Then introduce the single step change to the input
- Let the process again reach the steady state
- Collect the input and output values
- Perform the graphical calculations on the process reaction curve.
-

2.7 Closed Loop test

In the closed loop test the control is transferred to the controller rather than the operator. The closed loop control system provides a form of feedback to the process. The process is measured and then it is compared with the provided set point and final control element (control valve) is used to adjust the process. The process load will determine whether corrective action in the controller algorithm is needed. The PI controller was used for plant to control the overall system behavior.

CHAPTER 3
METHODOLOGY

3.1 Methodology

To accomplish the objective the project is divided into two parts, PI (D) controller and Fuzzy controller, firstly we have to work on only implementing PI controller once that is successful, we have to add fuzzy control to make it more efficient. As our work is on gas plant pressure, so usually derivative mode is not used because it is very sensitive to noise and pressure since it is fast process. To implement PI controller the best way is to do through empirical modeling, through which we can analyze process reaction curve by graphical method, however we have another more general method known as statistical.

Empirical model is build after getting the plant data from the experiment. Empirical model building should be undertaken using the six step procedure shown below in flow chart. This procedure ensures that proper data is generated through careful experimental design and execution

3.1.1 Experimental Design

An important aspect of the empirical modeling is the need for proper experimental design. Determine the input perturbation and its shape and duration, base operating conditions for the process. Get the prior knowledge before the experiment and its dynamic responses.

3.1.2 Plant Experiment

Plant operation should be monitored during the experiment because variation in plant operation is inevitable and large disturbances during the experiment can invalidate the results. The experiment is designed to establish the relationship between one input and one output (Single Input Single Output), changes in other inputs during the experiment can make the data unusable for identifying a dynamic model.

3.1.3 Determining Model Structures

First step in determining the model is to know the order of the system (e.g. first order, second order), based on the data. The initial structure is selected based on the prior knowledge of the unit operation, perhaps based on the structure of the fundamental model, and based on the patterns in the experimental data. The goal is to develop the model that matches the experimental data and to know the input output behavior of the system.

3.1.4 Parameter Estimation

The process reaction curve (Plant dynamic response) is obtained in the experiment is used to approximate the process parameters. The process parameters are:

- Dead Time (time delay), θ
- Time constant, τ
- Steady state Gain (Process gain), K_p

3.1.5 Diagnostic Evaluation

Evaluation is required before the model is used for the control. The diagnostic evaluation determines how well the model fits the data used for parameter estimation. Usually the comparison of the model prediction with the measured data is done in diagnostic evaluation.

3.1.6 Verification

In verification comparing the model to data collected at another time to make sure that typical variation in the plant operation does not significantly degrade model accuracy.

3.1.7 Plant investigation

To implement fuzzy logic, it is very important to investigate plant inorder to understand its behavior, in this pressure gas pilot plant, our main aim to get stable and fast response, the value of MV is very important to take corrective action, so this value of MV actually turns the valve, so, to turn the valve we need pressure of 0-2 bar, and plant vessel can store up to 6 bar. Knowing all these details, we can design controller for this plant, making it sure that its setpoint does not go above the limitation and the valve opening and closing must be remain within range to assure safety.

Container= 0-6 bar

Valve= 0-2 bar

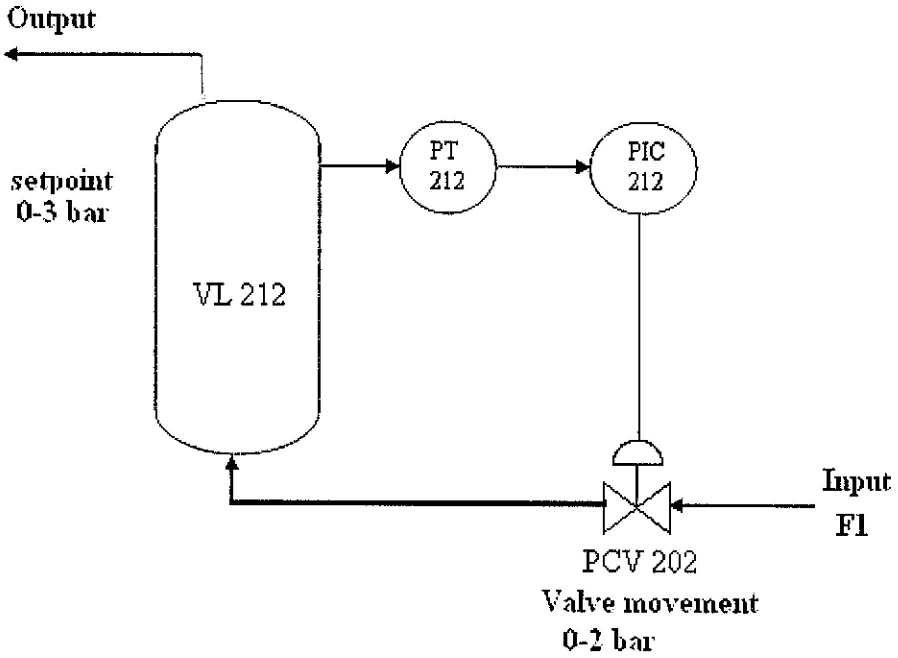


Fig 3.1 : Plant Block diagram

3.1. 8 Process reaction curve

3.1.8.1 Method 1

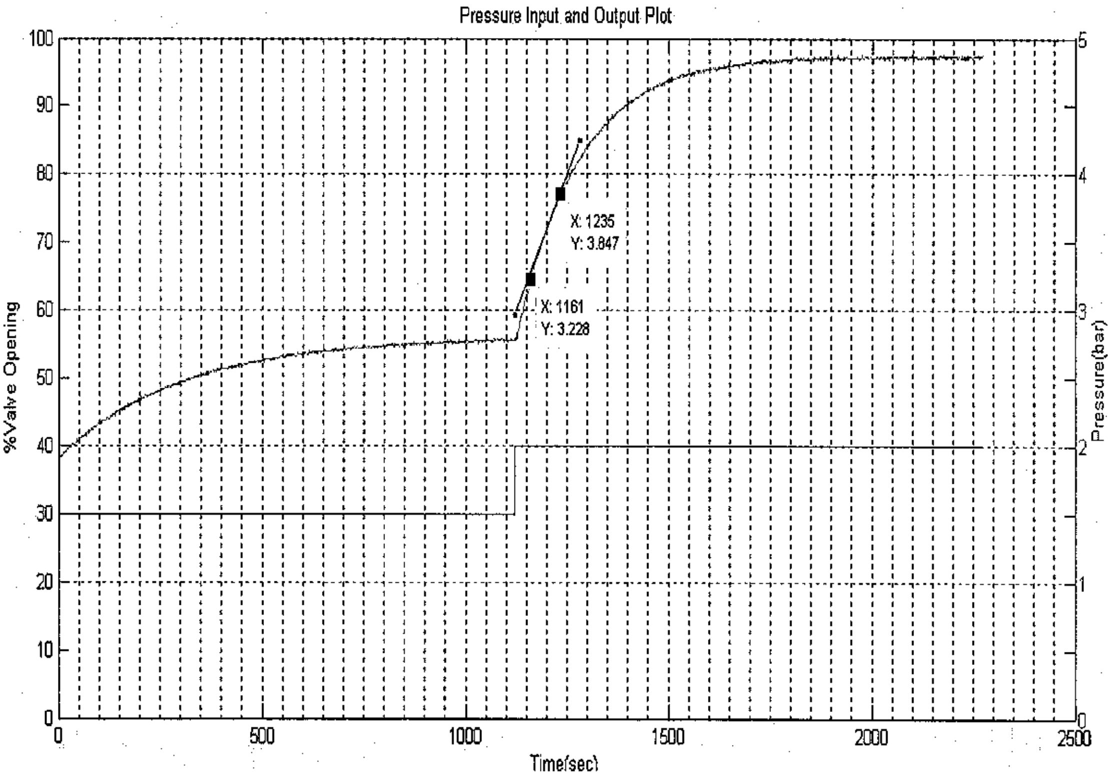


Figure 4.2: Method I Process Reaction Curve from Open Loop

Calculation:

Change in input $= \delta = 40\ \% - 30\% = 10\ \%$ Open

Change in output$= \Delta = 4.8650$ bar – 2.787 bar

$= 2.078$ bar

$K_p \quad = \quad \Delta / \delta$

$= \quad 2.078/\ 10\ \%$

$= \quad 0.2078$ bar / %

Slope $= \quad (3.847 - 3.228)/(1235 - 1161)$

Slope $= \quad 0.008365$

Time Constant $= \quad \tau \quad = \quad 2.078/0.008365$

$= \quad 248$ seconds

Dead Time $= \theta \quad = \quad 10$ seconds

The model transfer function:

$$G(s) = \frac{0.2078\, e^{-10s}}{248s+1}$$

3.1.8.2 Method II

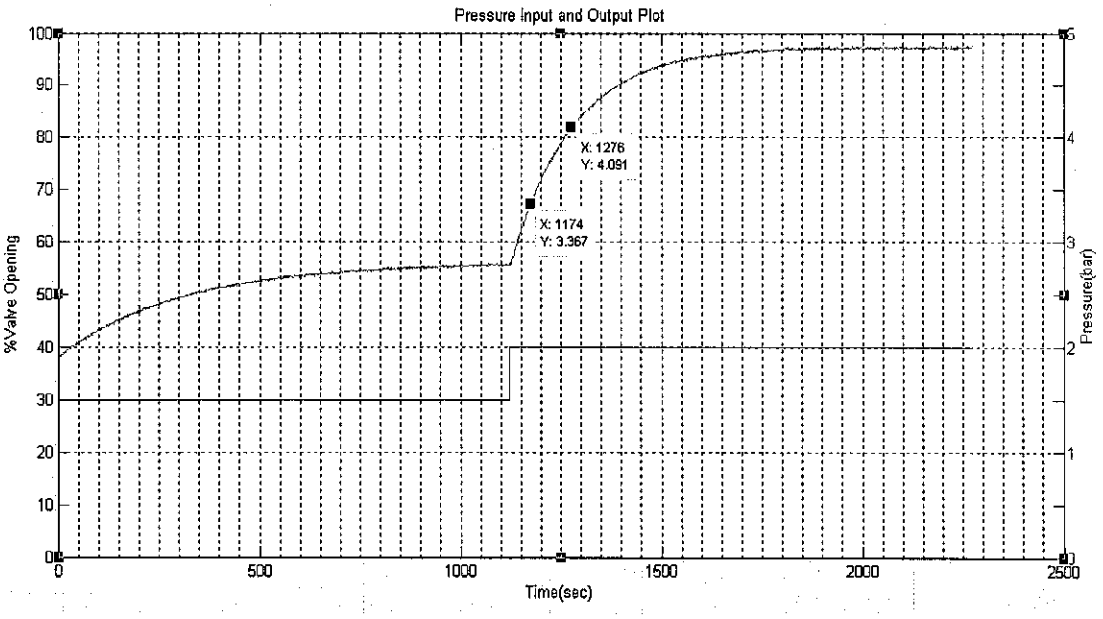


Figure 4.3: Method II Process Reaction Curve from Open Loop

Calculation:

Change in input $=\delta = 40\ \% - 30\% = 10\ \%$ Open
Change in output$= \Delta = 4.8650$ bar – 2.787 bar
$= 2.078$ bar

$K_p \quad = \quad \Delta / \delta$

$\quad\quad = \quad 2.078/\ 10\ \%$

$\quad\quad = \quad 0.2078$ bar / %

$0.63\ \Delta = \quad 1.30914+2.787$

$\quad\quad = \quad 4.09614$

$0.28\ \Delta = \quad 0.58184+2.787$

$0.28\,\Delta = 3.36884$

$T_{63\%} = 156$ seconds

$T_{28\%} = 54$ seconds

$\tau = 1.5\,(156 - 54)$

$= 153$ seconds

$\theta = t_{63\%} - \tau$

$= 156 - 153 = 3$ seconds

The model transfer function:

$$G(s) = \frac{0.2078\,e^{-3s}}{153s+1}$$

3.1.9 Plant Data

Working on UTP gas pilot plant and using the methods discussed in Marlin, to get the plant model, the following transfer function for the plant is obtained;

$$G(s) = \frac{0.2078\,e^{-3s}}{153s+1}$$

Using this transfer function in Matlab we obtained a Bode plot and from there we got value of closed loop transfer function.

```
s=tf('s');
G=[0.2078/(153*s+1)];
G.inputdelay=3;
bode(G);
grid
```

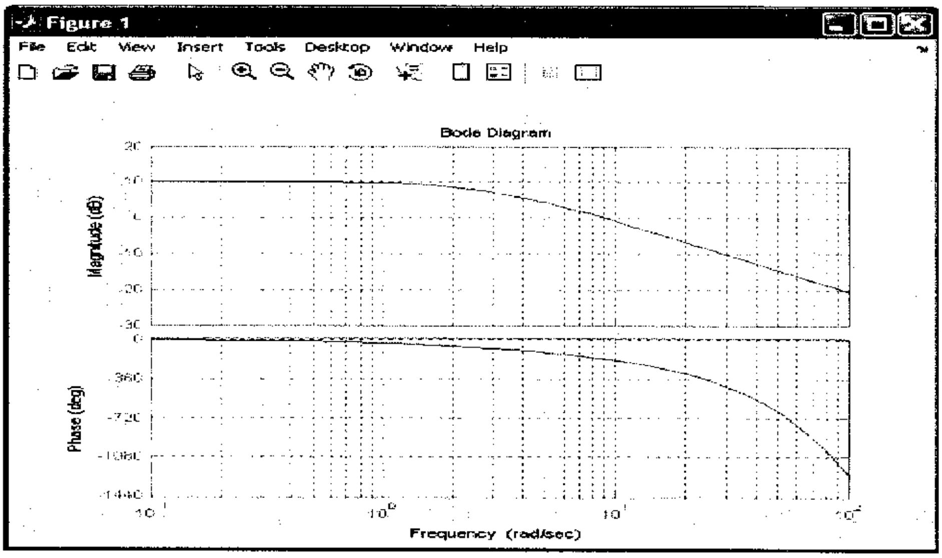


Fig 3.2: Bode plot

From this bode plot we can get values of tuning constant and we can use them in PID controller inorder to get desired response.

From above graph we get,

□c=10

AR=3dB

AR=1.41

Ku=0.71, Pu=0.628

Using Ziegler Nicholas closed loop tuning method we can get

controller	Kc	Ti	Td
PI	0.323	0.253	--------
PID	0.417	0.314	0.785

Table 3.1: Ziegler Nicholas closed loop tuning constants

Now we can put these tuning constants in PID controller to see its response.

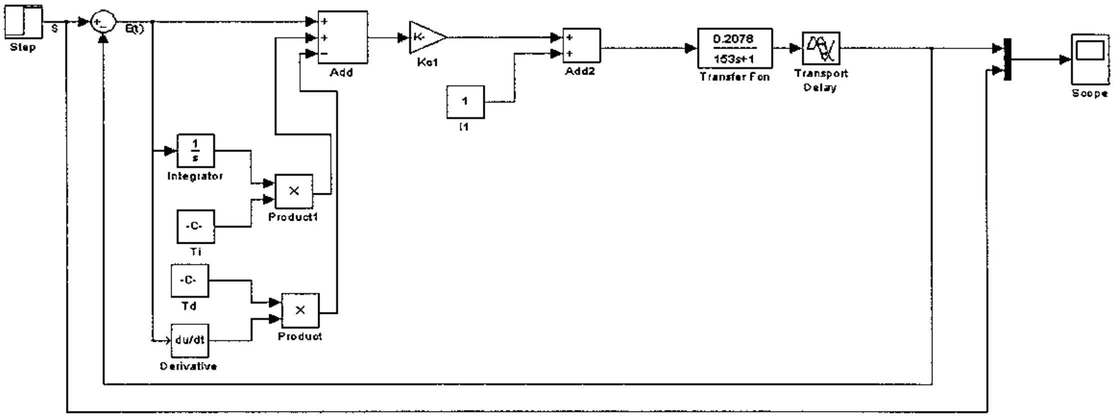


Fig 3.3: PID controller

3.2 Fuzzy controller

Fuzzy control is an appealing alternative to conventional control methods when systems follow some general operating characteristics and a detailed process understanding is unknown or traditional system models become overly complex, The capability to qualitatively capture the attributes of a control system based on observable phenomena is a main feature of fuzzy control.

Fuzzy logic is conceptually easy to understand, the mathematical concepts behind fuzzy reasoning are very simple. Fuzzy logic is a more intuitive approach without the far-reaching complexity.

- Fuzzy logic is flexible.

With any given system, it is easy to layer on more functionality without starting again from scratch. Everything is imprecise if you look closely enough, but more than that, most things are imprecise even on careful inspection, Fuzzy reasoning builds this understanding into the process rather than tacking it onto the end. Fuzzy logic can model nonlinear functions of arbitrary complexity. You can create a fuzzy system to match any set of input-output data. This process is made particularly easy by adaptive techniques like Adaptive Neuro-Fuzzy Inference Systems (ANFIS), which are available in Fuzzy Logic Toolbox.

- Fuzzy logic can be built on top of the experience of experts.

- Fuzzy logic can be blended with conventional control techniques; Fuzzy systems don't necessarily replace conventional control methods. In many cases fuzzy systems augment them and simplify their implementation.

Now to implement PI-Fuzzy controller, we have converted the equation of Conventional PI(D) controller into a form that is equivalent and understandable in simulink.

We have,

$$u = K_p e + K_I \int_0^t e \, dt,$$

Take derivative on both sides, we get,

$$\dot{u} = K_p \dot{e} + K_I e$$

$$\Delta e = e(k) - e(k-1) = (1 - z^{-1})e(k)$$

Since the input to the block is *e(k),* the transfer function we want is,

$$\frac{\Delta e(k)}{e(k)} = (1 - z^{-1})$$

Or

$$\frac{u}{\Delta u} = \frac{1}{(1 - z^{-1})}$$

3.2.1 Fuzzy Logic Control Design

The FLC developed here is a two-input single-output controller. The two inputs are the deviation from setpoint *error, e(k)*, and *error rate*, *□e(k)*. The FLC is implemented in a discrete-time form using a zero-order-hold as shown in Figure 3a. The operational structure of the Fuzzy controller is shown in Figure 3b.

3.2.1.1Fuzzification/Defuzzification

Fuzzification and defuzzification involve mapping the fuzzy variables of interest to "crisp" numbers used by the control system. Fuzzification translates a numeric value for the error, *e(k)*, or error rate, *De(k)*, into a linguistic value such as positive large with a membership grade. Defuzzification takes the fuzzy output of the rules and generates a "crisp" numeric value used as the control input to the plant.

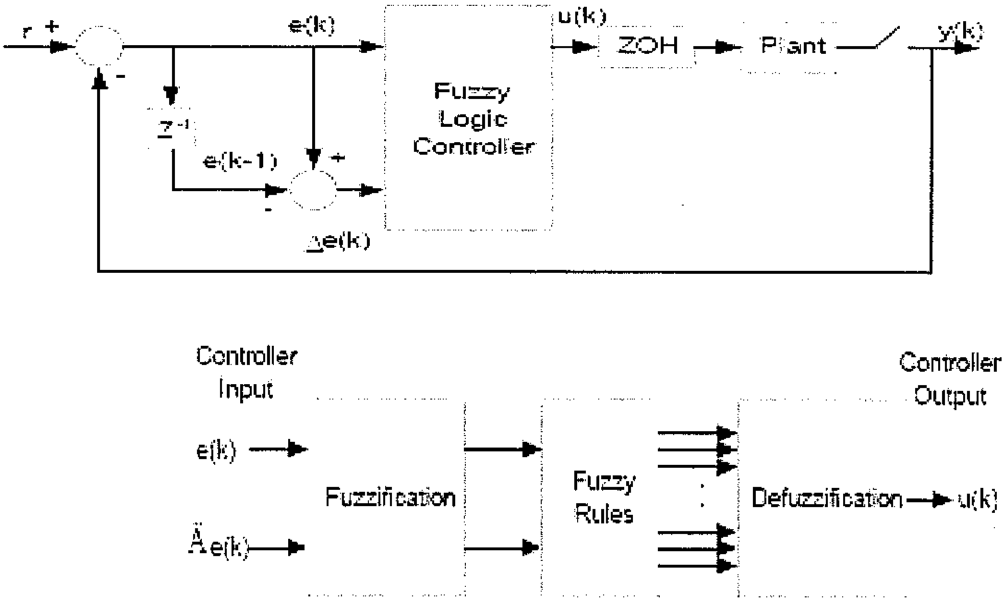


Fig 3.4: Block diagram of Fuzzy controller

The FLC membership functions are defined over the range of input and output variable values and linguistically describes the variable's universe of discourse.

3.2.2 Shape of membership functions

The shape of membership has great importance and influence on the "weighting" of the input e.g. A triangular shaped MF has only one input value where the degree of membership (DOM), it is necessary to have a range of input values where the DOM values should be 1 it is necessary to use trapezium shaped MFs. Gauß- shaped MFs are often used for input values. For control purposes just – Trim shaped MFs are used to map the input values to the fuzzy sets, other types of membership function also taken into consideration depending upon system and its need e.g rectangular and trapezium shaped ,Z-shaped and S-shaped MFs (figure 3.5). Other shapes for membership functions which are often used are Gauß- and cos-shaped MFs. The values between these points can be interpolated very easily.

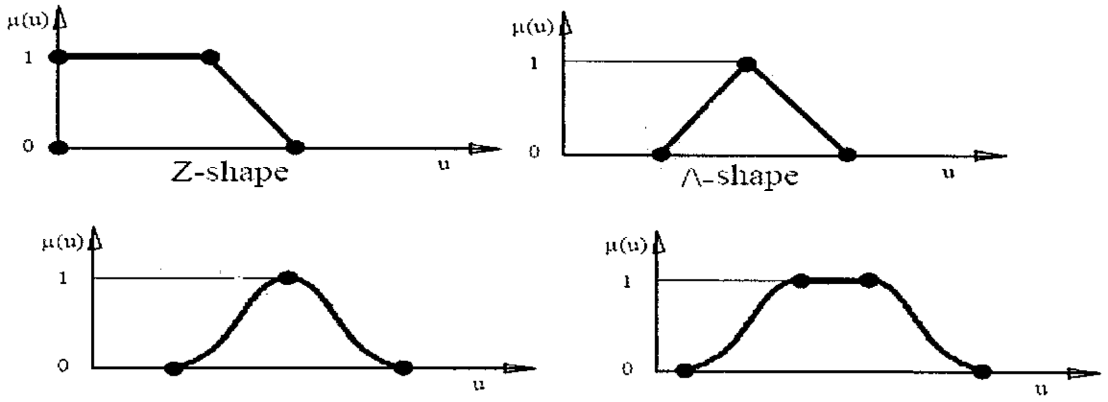


Fig 3.5: Membership Curves

3.2.3 Inference and Composition

The inference processing (rule evaluation) is the central part of the knowledge based decision making, and is expressed by linguistic rules. Rules are statements expressing a dependency relation among system inputs and system outputs. Individual rules represent parts of the solution to a problem. All rules considered together determine the final solution.

Rule evaluation takes the fuzzy inputs (degrees of membership) from the fuzzification step and the rules from the knowledge base and calculates fuzzy outputs. A typical rule base for a pressure controller below shows in principle the rule evaluation.

In the fist part of Fyp1, I have worked on this equation and used three membership functions, but for FYP (II), currently I am working on 5 to 7 function.

The whole fuzzy system consists of below algorithm.

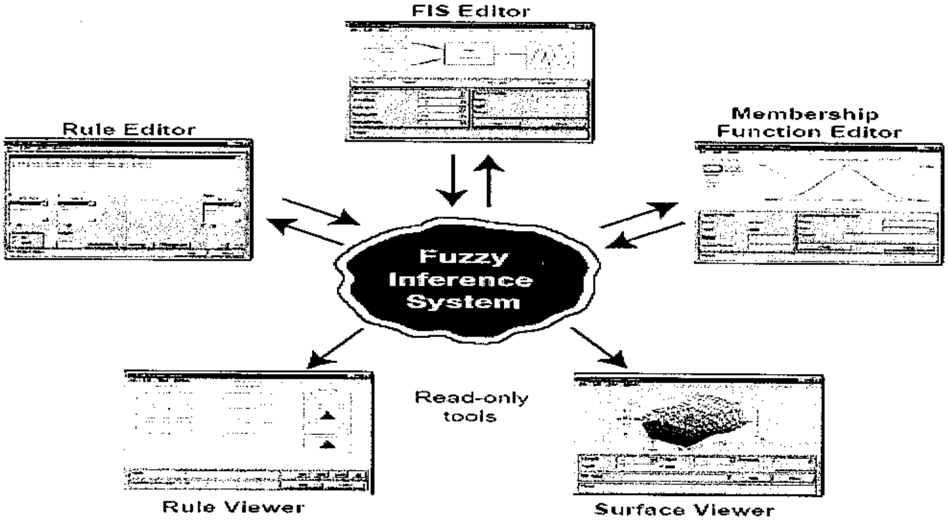
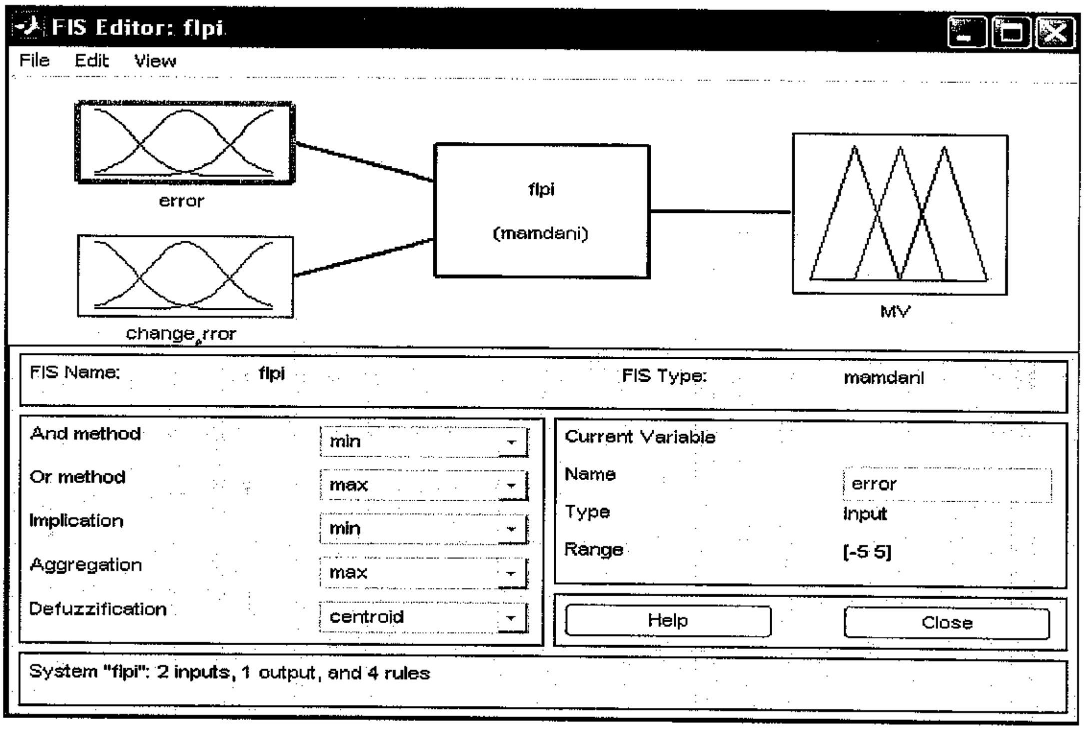


Fig 3.6: Fuzzy algorithm

3.2.4 FIS Editor

FIS stand for Fuzzy Inference System, in this FIS we select inputs and outputs, we can add input as much as we need, Here our inputs are error (e) and change of error (de) and output is mv. The range for our input is from 0-6.



FIS Name: flpi

FIS Type: mamdani

And method: min

Or method: max

Implication: min

Aggregation: max

Defuzzification: centroid

Current Variable

Name: error

Type: input

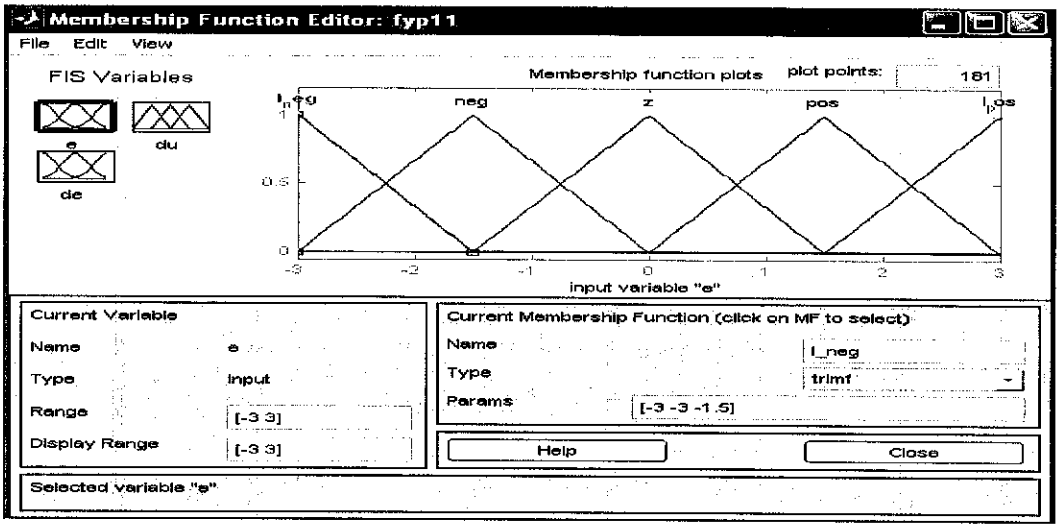
Range: [-5 5]

System "flpi": 2 inputs, 1 output, and 4 rules

Fig 3.7: Fuzzy FIS Editor

3.2.5 Membership Function

This is the most important and crucial part because all process depends upon its shape of its curve who are responsible to make decision, so one need to very careful and have sufficient knowledge about the plant then he can set these membership functions, it is very important for us to know what is the range of membership function, this can be obtained from real plant data, we must try to figure out how much would be setpoint and range of inputs and manipulated variable.

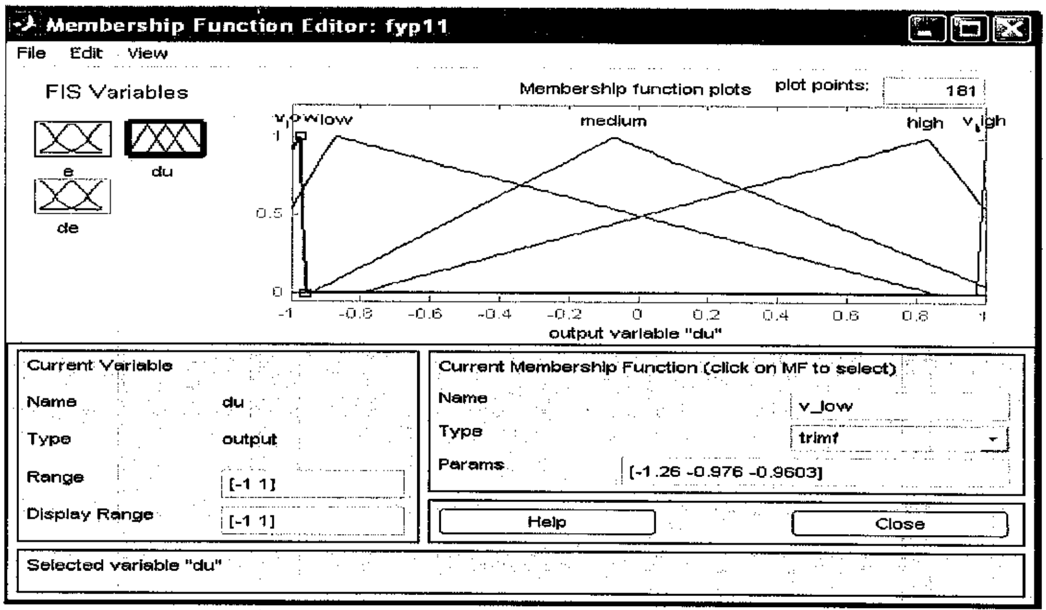


Current Variable
Name e
Type input
Range [-3 3]
Display Range [-3 3]

Current Membership Function (click on MF to select)
Name l_neg
Type trimf
Params [-3 -3 -1.5]

Selected variable "e"

Fig 3.8(a): Fuzzy Membership function for input



Current Variable
Name du
Type output
Range [-1 1]
Display Range [-1 1]

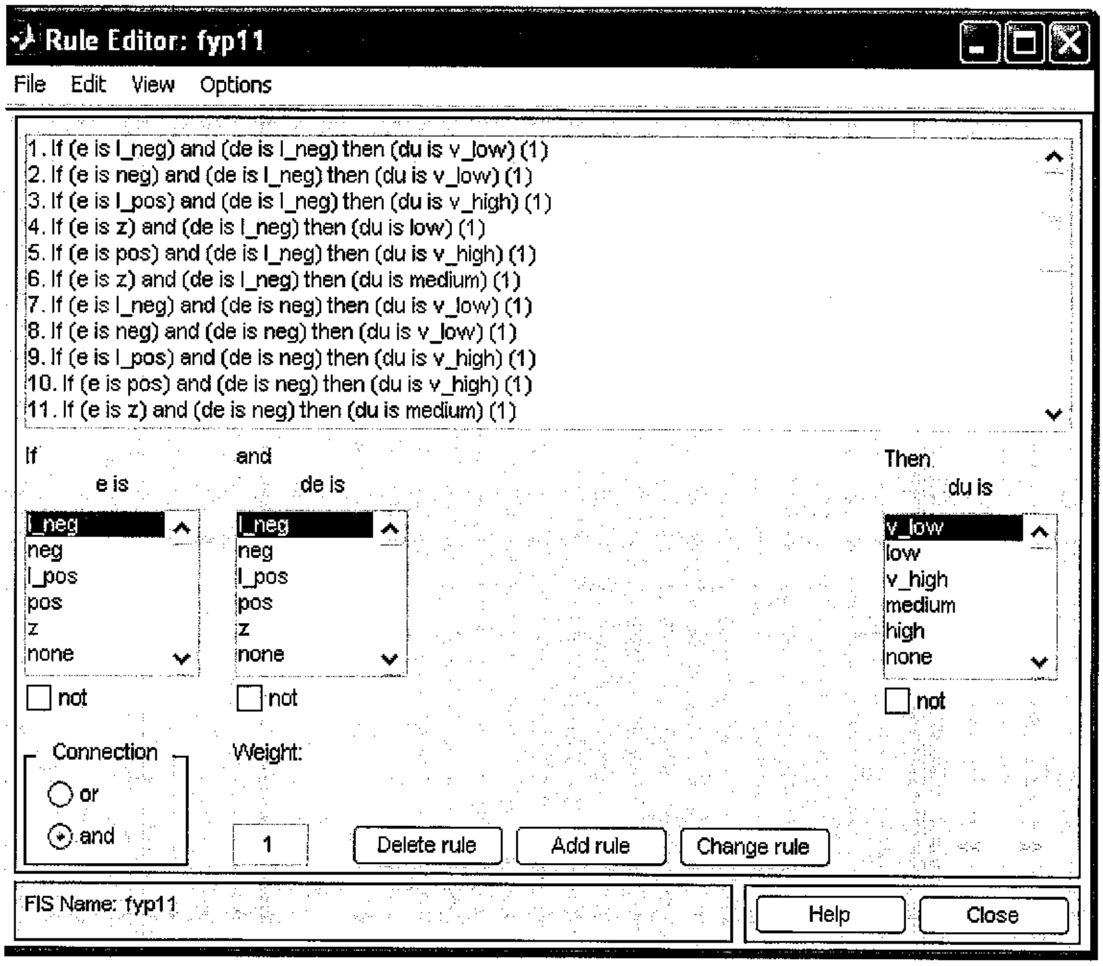
Current Membership Function (click on MF to select)
Name v_low
Type trimf
Params [-1.26 -0.976 -0.9603]

Selected variable "du"

Fig 3.8(b) Fuzzy Membership function for output

3.2.6 Rule editor and viewer

Rule Editor allows us to construct the rule statements automatically, From the GUI, we can set them on knowledge based. Now rules depend upon membership function, more the membership function we have, more will be the number of rules therefore these rules play key role in precise evaluation at every point.



1. If (e is l_neg) and (de is l_neg) then (du is v_low) (1)
2. If (e is neg) and (de is l_neg) then (du is v_low) (1)
3. If (e is l_pos) and (de is l_neg) then (du is v_high) (1)
4. If (e is z) and (de is l_neg) then (du is low) (1)
5. If (e is pos) and (de is l_neg) then (du is v_high) (1)
6. If (e is z) and (de is l_neg) then (du is medium) (1)
7. If (e is l_neg) and (de is neg) then (du is v_low) (1)
8. If (e is neg) and (de is neg) then (du is v_low) (1)
9. If (e is l_pos) and (de is neg) then (du is v_high) (1)
10. If (e is pos) and (de is neg) then (du is v_high) (1)
11. If (e is z) and (de is neg) then (du is medium) (1)

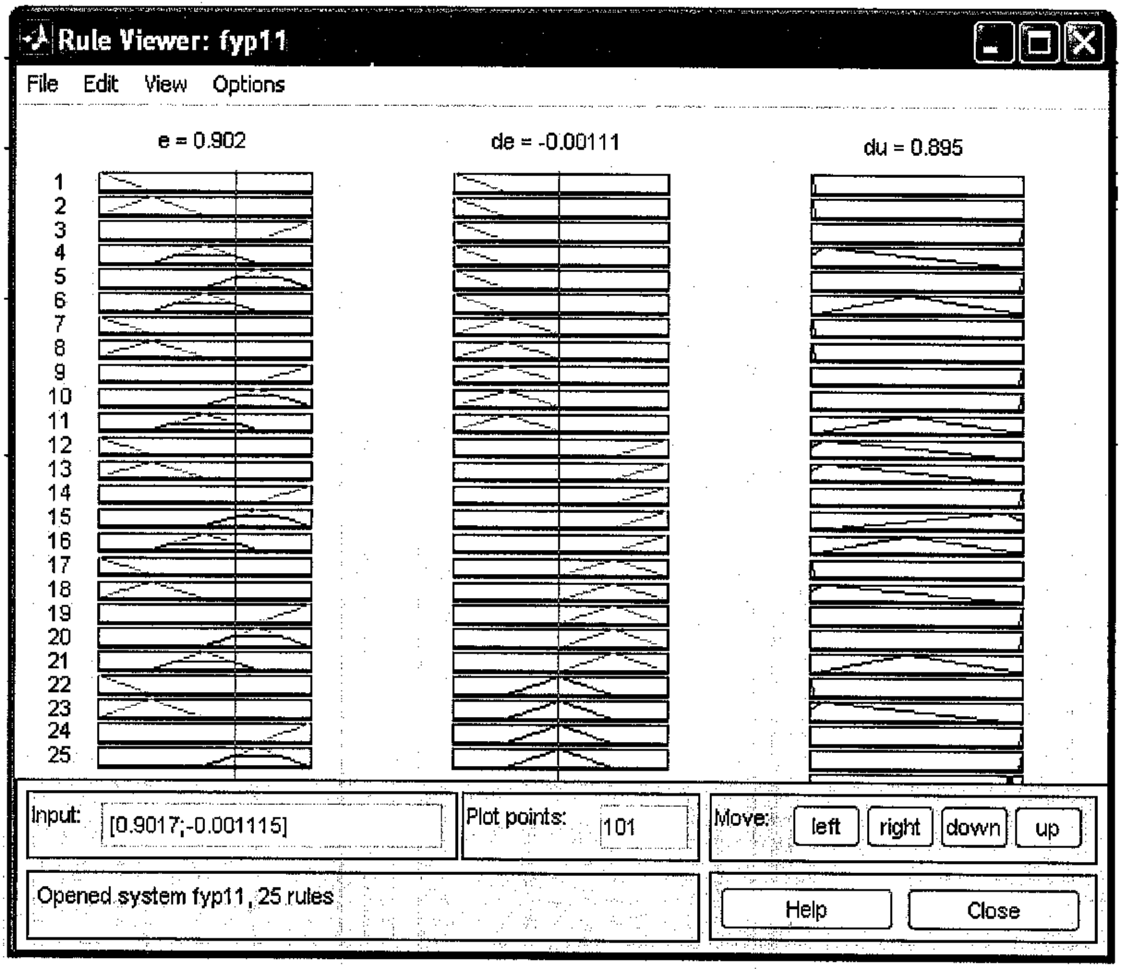
Connection: and

Weight: 1

FIS Name: fyp11

Fig 3.9: Fuzzy Rule Editor and viewer

This gives information about our change of error here we can see how our system is working, it tells us the rate of change of error and we can see how our system is really behaving and we can analyze its weaknesses.



e = 0.902

de = -0.00111

du = 0.895

Input: [0.9017;-0.001115]

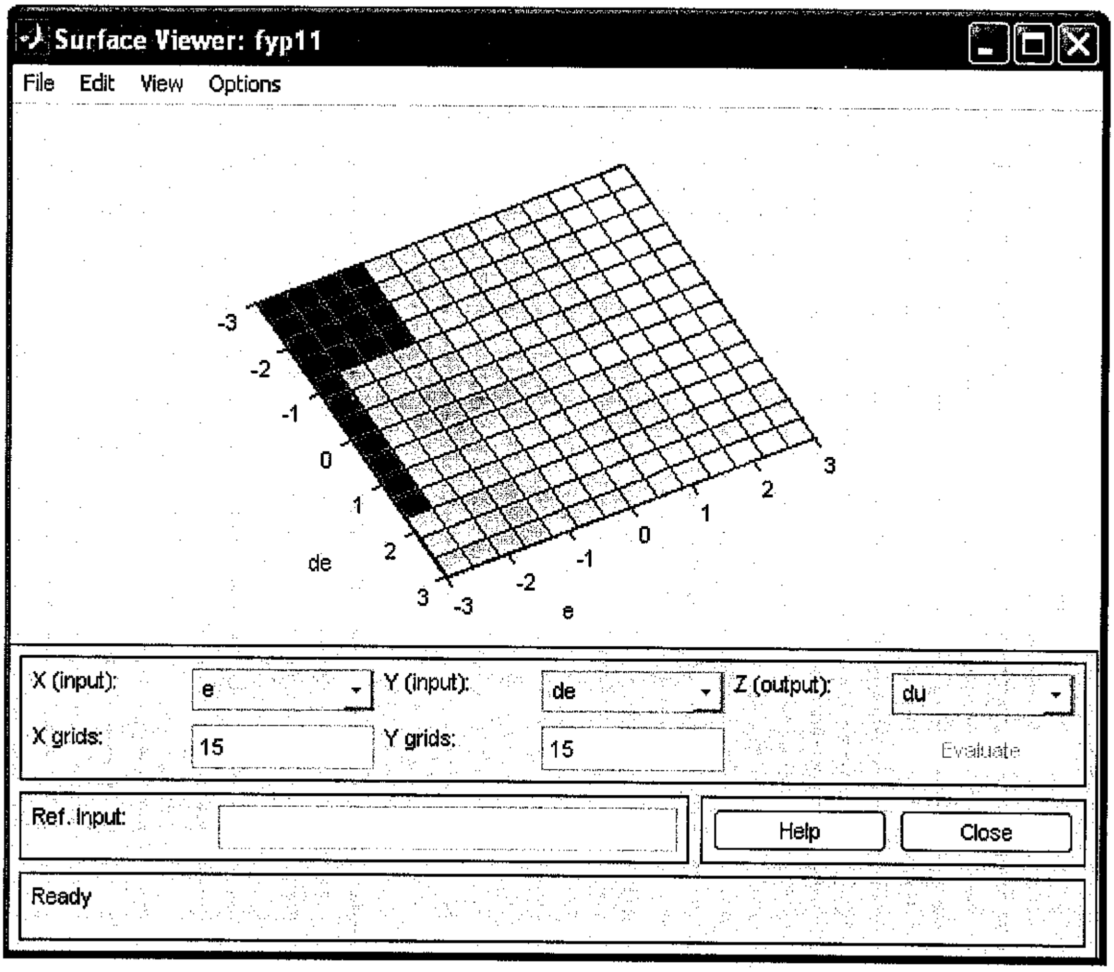
Plot points: 101

Opened system fyp11, 25 rules

Fig 3.10: Fuzzy viewer

3.2.7 Surface Viewer

A three-dimensional surface that represents the mapping of error and change of error at every point. Because this surface represents two-input and one-output case, you can see the entire mapping in one plot, each cube of on this surface is crisp value which gets its value from membership function and this crisp value represents the position of valve.



X (input): e

Y (input): de

Z (output): du

X grids: 15

Y grids: 15

Ref. Input:

Ready

Fig 3.11: Fuzzy Surface Viewer

Fuzzy logic PI controller

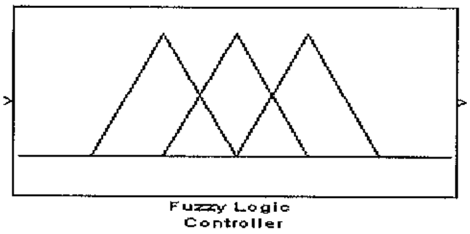


Fig 3.12: Fuzzy Logic Controller

all above, inputs and output, membership function and rules are copied into above fuzzy logic controller block.

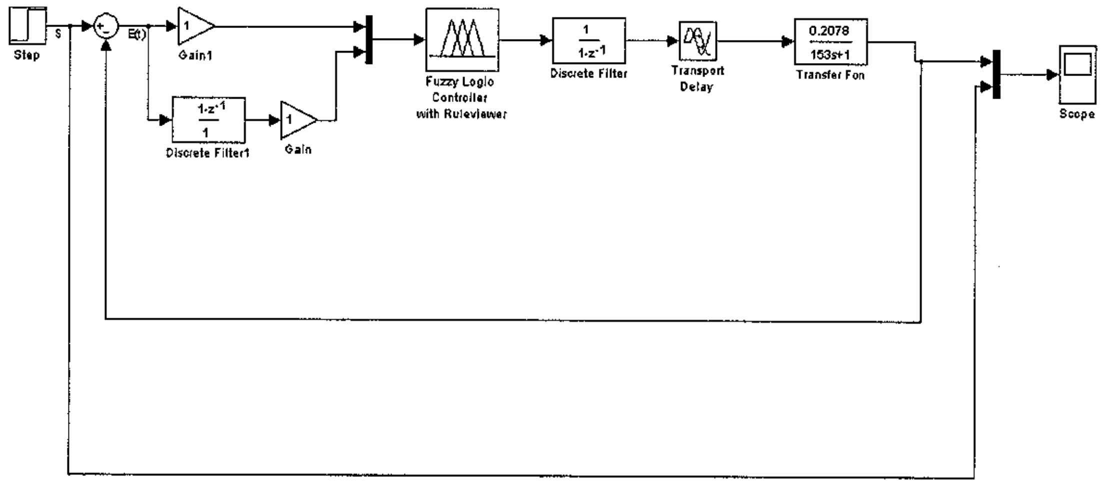


Fig 3.13: Simulink with Fuzzy controller

3.3 Tools and Software

3.3.1 MATLAB and Simulink

MATLAB is interactive software whose basic data element is an array that does not require dimensioning. This allows us to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time,MATLAB is offering array operations and since the data received is used in MATLAB to get the Process Reaction Curve (PRC). MATLAB also offers some programming features similar to other languages. The Graphical User Interface (GUI) in MATLAB allows us to use the Process Reaction Curve for our calculations of the transfer function. MATLAB features were really utilized in this project.

Simulink is an extension to MATLAB. It is a software package for modeling, simulating, and analyzing the dynamic systems. It allows building accurate computer model of dynamic systems using block diagram notation. Simulink model can produce the simulation and is as easy approach to know the dynamic behavior of the system.

3.3.2 MATLAB Simulink Xpc Target

The control block diagram is constructed in Simulink. The necessary blocks that are used to create model are step input, Xpc target scope, and other blocks are added in the application. The suitable parameters are entered for the scope block to view the result. The Simulink output block is added to log the output for analysis. The Xpc target is used to visualize signal while running the plant. The mode is selected as external because the model is connected to the Gaseous Pilot Plant. The block diagram which is connected to Xpc target is shown in figure 3.14 below.

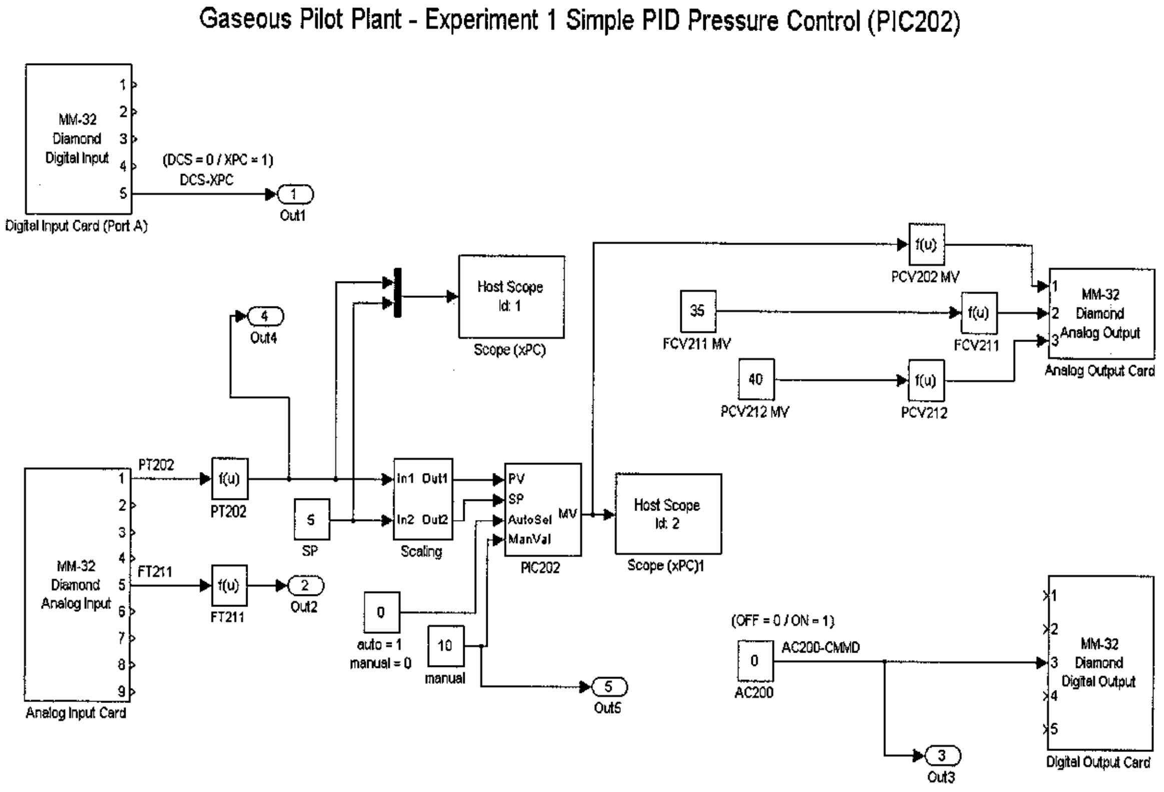


Figure 3.14: Block Diagram using Simulink Xpc Target

The Xpc Target industrial PC acts as server and is connected to the Gaseous Pilot Plant through the Simulink model. The signal from Simulink model is written to the server via Xpc target scope block. When the application is downloaded on the PC it creates the scope automatically.

The Xpc target validates the design in real time. In this project the controller is running in real time connected to the Pilot Plant. Figure 3.15 shows the block diagram of Xpc target box and Figure 3.16 shows the Default ideal PID using difference error approach.

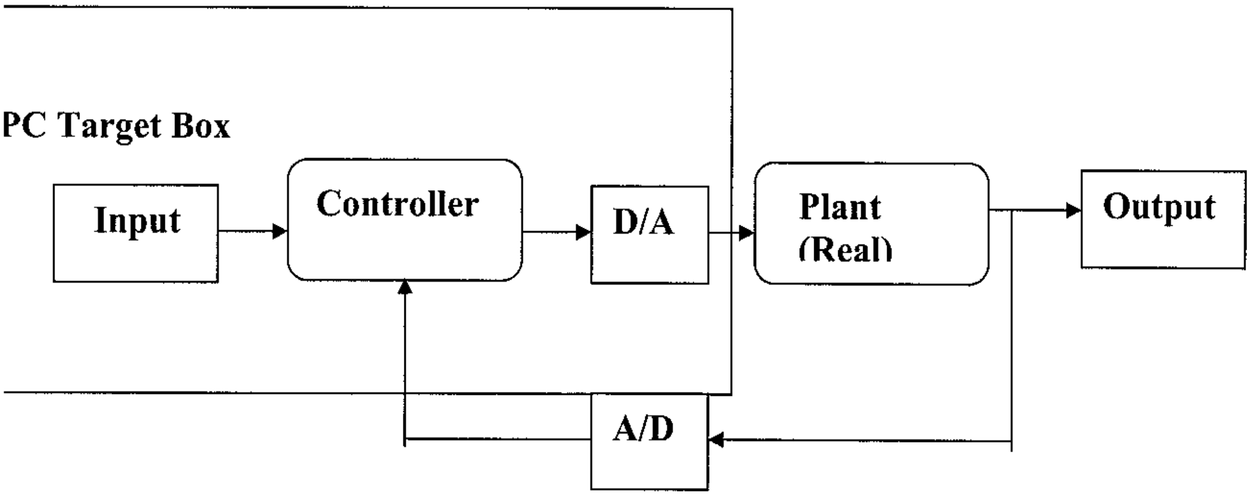


Figure 3.15: Block diagram of xPC Target box

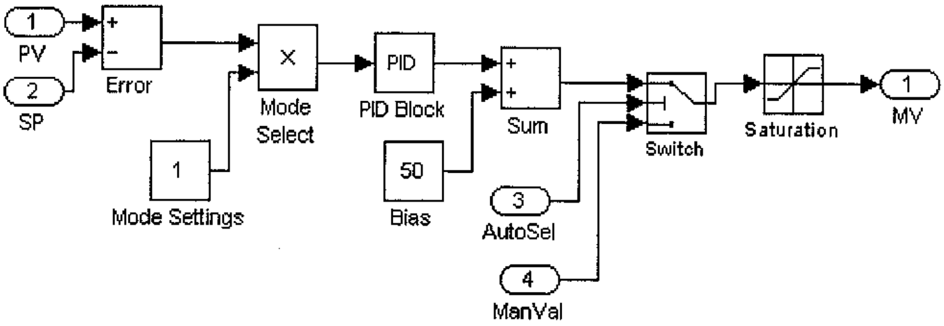


Figure 3.16: Default ideal PID using difference error approach.

CHAPTER 4

RESULTS

4.1 Process Reaction Curve (Closed Loop Method)

The experiment was conducted at UTP in the Plant Process laboratory. The parameters were estimated from the process reaction curve. First of all the variables are identified. Those variables include manipulated variable (MV), process variable (PV) and set point (SP). The PV and SP are flow rate and MV is percent valve opening for this experiment. A small change is applied to the manipulated variable (%valve opening) to generate the process reaction curve.

$$G(s) = \frac{0.2078\, e^{-3s}}{153s+1}$$

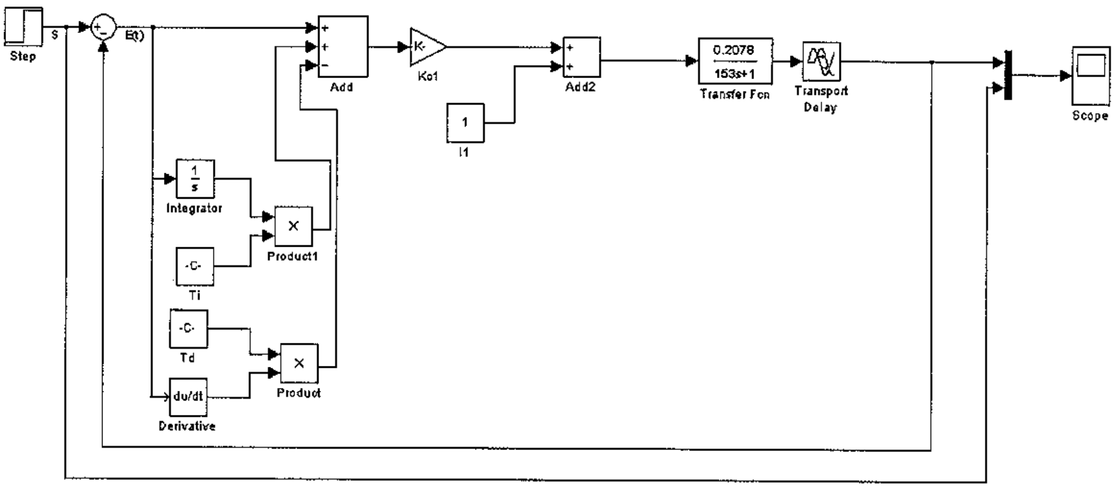


Fig 4.1: A matlab/simulink model of the plant

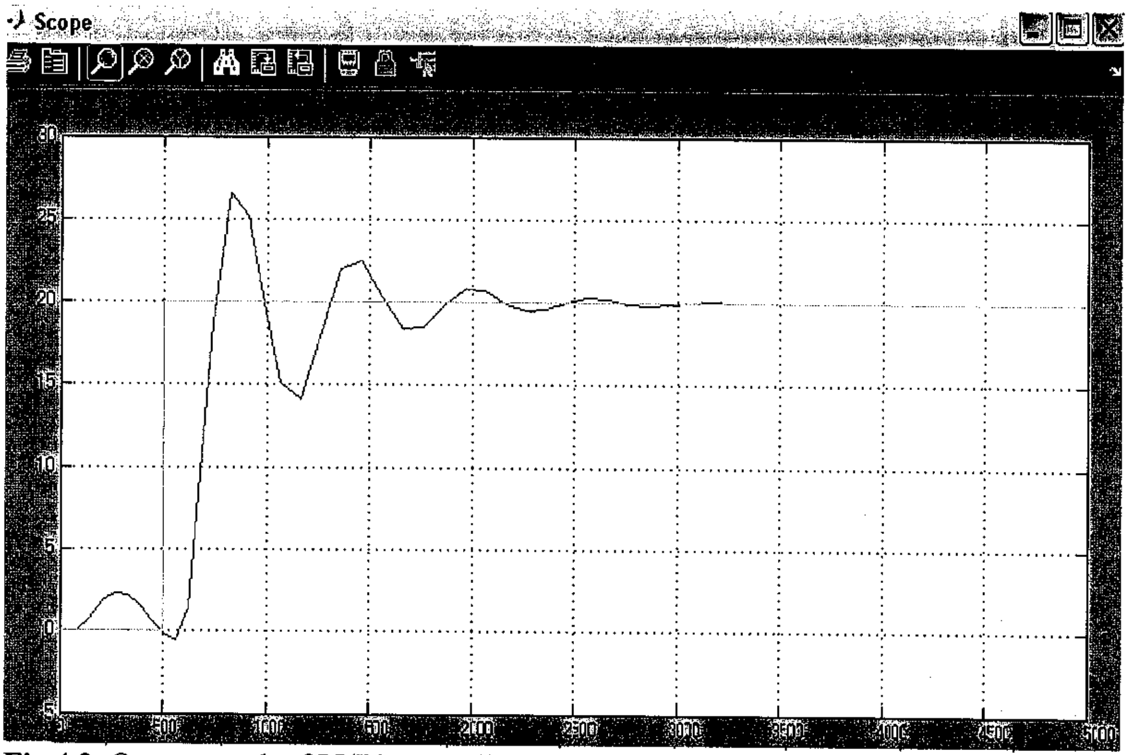


Fig 4.2: Output graph of PI(D) controller

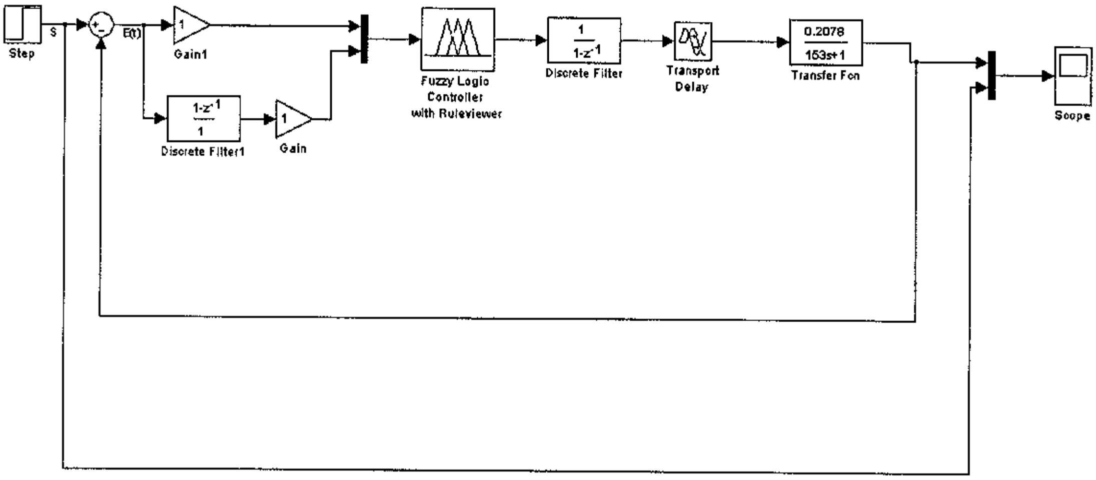


Fig 4.3: A matlab/Simlink model of the plant with fuzzy controller

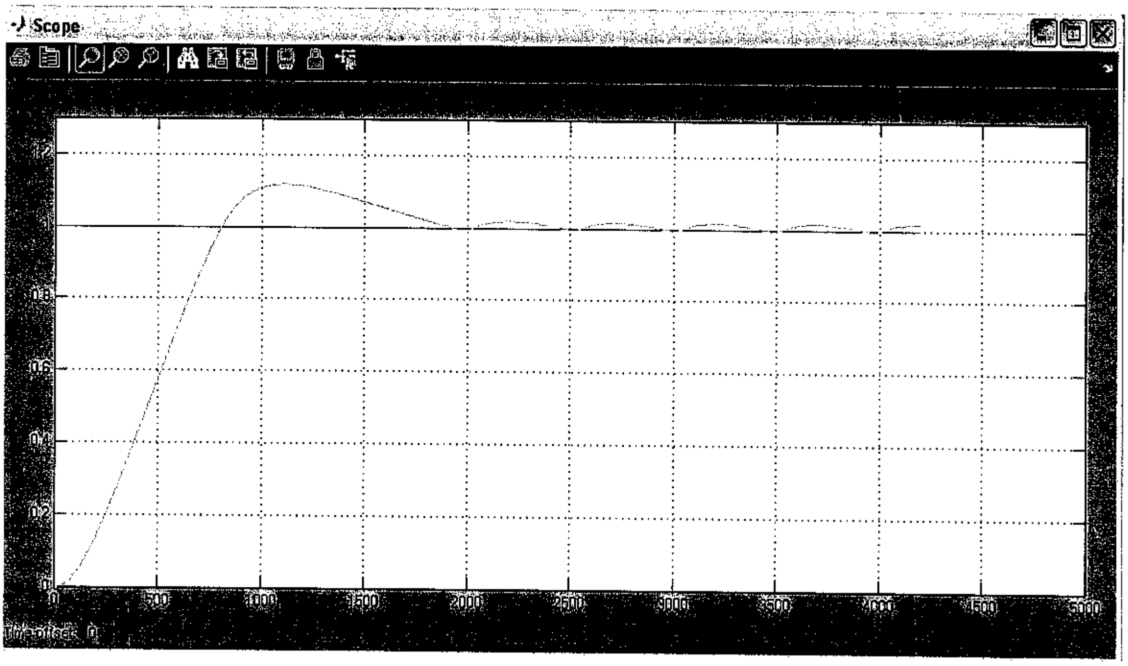


Fig 4.4: Output graph of PI-Fuzzy controller

CHAPTER 5

DISCUSSION AND FINDINGS

5.1 DISCUSSIONS

Fuzzy logic is best for both linear and non-linear process to control, several research articles have reported on the implementation of fuzzy to handle complicated process in simple way. Initially to implement this controller require us to understand the whole process and how can we convert human experience into form understandable by computers. Second is Empirical modeling which is a good guide to start with the project. It is easy to implement and easy to evaluate if the model data is collected from the plant. After getting the process reaction curve from the plant operation it is easy to approximate the parameters such as process gain (K_p), Dead time (θ) and time constant (τ). The transfer function is formed from the differential equations and the transfer function is easy approach for input-output models rather than solving differential equations. The parameter values are substituted in the transfer function to get the process response.

Ziegler Nicolas technique will also be used inorder to get accurate tuning constant to make our process more efficient but here our aim was to design Fuzzy controller which is absolutely different from PID controller but this can be designed by translating working of PID controller into human imitative form, this technique is newly introduced to make controller robust, fast and efficient in handling process variables.

5.2 Next Project Recommendations

This project was sequel of previous projects, most of student tried hard to optimize PID control on UTP gas plant but in this project I tried to simulate same data but using Fuzzy logic which is totally new approach towards controllers at plants, so far I was able to translate PI controller into fuzzy controller on Matlab/simulink, although results are not really very impressive but it is a first step towards developing new kind of controller. I have done simulation on Fuzzy controller and it's totally ready but problem is, it is not working on real plant due to some reason which I didn't able to figure them out.

Now, in this regard, I would like to recommend that if any FYP Student is working on this project, he must try to implement it on real time plant, he can use my work as basis and then it will easy for him to save time on process reaction curve or getting transfer function and then building computer simulation, he can directly work on real plant and try to implement and I believe FYP time would be sufficient for him to build this controller on UTP Plant.

CHAPTER 6

CONCLUSION

Fuzzy Logic provides a different way to approach a control or classification problem. This method focuses on what the system should do rather than trying to model how it works. One can concentrate on solving the problem rather than trying to model the system mathematically, if that is even possible. On the other hand the fuzzy approach requires a sufficient expert knowledge for the formulation of the rule base, the combination of the sets and the defuzzification. In General, the employment of fuzzy logic might be helpful, for very complex processes, when there is no simple mathematical model (e.g. Inversion problems), for highly nonlinear processes or if the processing of (linguistically formulated) expert knowledge is to be performed To design a fuzzy controller, one should be knowledgeable about the process because it is a fully knowledge-based and to translate that knowledge into fuzzy logic controller modes, membership function and rules cannot be implemented directly, one has to monitor whole process very carefully and then to draw its behavior then a translation of this process functioning into fuzzy that includes the fuzzification reasoning, inference and defuzzification.

REFERENCES

[1]. Norman S. Nise, *Control Systems Engineering*, 4th Edition, John Wiley & Sons Inc, 2004.

[2]. David Macdonald, *Practical Process Control for Engineers and Technicians*, 1st Edition, Elsevier, 2005.

[3]. Thomas E. Marlin, *Process Control*, 2nd Edition, McGraw Hill, 2000.

[4]. Douglas M. Considine, *Process/Industrial Instruments & Controls Handbook*, 4th Edition, McGraw Hill International Editions, 1993.

[5]. Franklyn W. Kirk and Nicholas R. Rimboi, *Instrumentation*, Third Edition, American Technical Publishers.INC

[6]. J. Schwarzenbach, *Essentials of Control,* First Edition, Longman, 1996.

[7]. http://en.wikipedia.org/wiki/Controller_(control_theory)

MATLAB PROGRAM OF FUZZY

```
[System]
Name='pif1'
Type='mamdani'
Version=2.0
NumInputs=2
NumOutputs=1
NumRules=4
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='input1'
Range=[-1 1]
NumMFs=2
MF1='-':'trimf',[-1.8 -1 1]
MF2='+':'trimf',[-1 1 1.8]

[Input2]
Name='input2'
Range=[-1 1]
NumMFs=2
MF1='-':'trimf',[-1.79470899470899 -0.994708994708994 1.00529100529101]
MF2='+':'trimf',[-1 1 1.8]

[Output1]
Name='output1'
Range=[-1 1]
NumMFs=3
MF1='-':'trimf',[-1.8 -1 0]
MF2='z':'trimf',[-1 0 1]
MF3='+':'trimf',[0 1 1.8]

[Rules]
2 2, 3 (1) : 1
2 1, 2 (1) : 1
1 2, 2 (1) : 1
1 1, 1 (1) : 1
```

APPENDIX A

ACRONYMS

Accuracy:	The difference between the reading of an instrument and the true value of what is being measured, expressed as percent of full instrument scale.
Automatic Control:	The process of using the differences between the actual value and desired value of a variable to take the corrective action without human intervention.
Bias:	It is the set value for control valve incases of failure of instrument air and is provided by the process operator for safety purpose.
Bumpless Transfer:	Change from manual to automatic control or vice versa, without change in control system.
Control Valve:	Control valves or proportional valves are power operated devices used to modify fluid flow or pressure rate in a process system.
Calibration:	The procedure for determining, correcting or checking the absolute values to do the measurement on a measuring instrument.
Desired Value:	The value of the controlled variable which is to be maintained.
Fluid:	Any gas or liquid
Flapper Nozzle:	A combination of elements in a pneumatic controller.
Instrumentation	Include pressure / temperature / flow / level switches, transmitter,

and Control System:	controller, regulator, recorder, indicator, gauges.
Measured Variable:	Analogous to controlled variable when it is used in connection with the control applications.
Measuring Element:	The primary device for measuring the process variable.
Offset:	A sustained deviation of the controlled variable from the set point.
Open Loop:	A system in which no comparison is made between actual the actual value and desired value of a process variable.
Orifice Plate:	A thin, circular metal plate with an opening in it. It is used for measuring flow rate.
Overshoot:	The amount by which a changing process variable exceeds the desired value as changes occur in a system.
Process Reaction Curve:	A record of the reaction of a control system to a step change.
Set Point:	The position at which the control point setting mechanism is set. This is the same as desired value of the controlled variable.

APPENDIX II

FIGURES

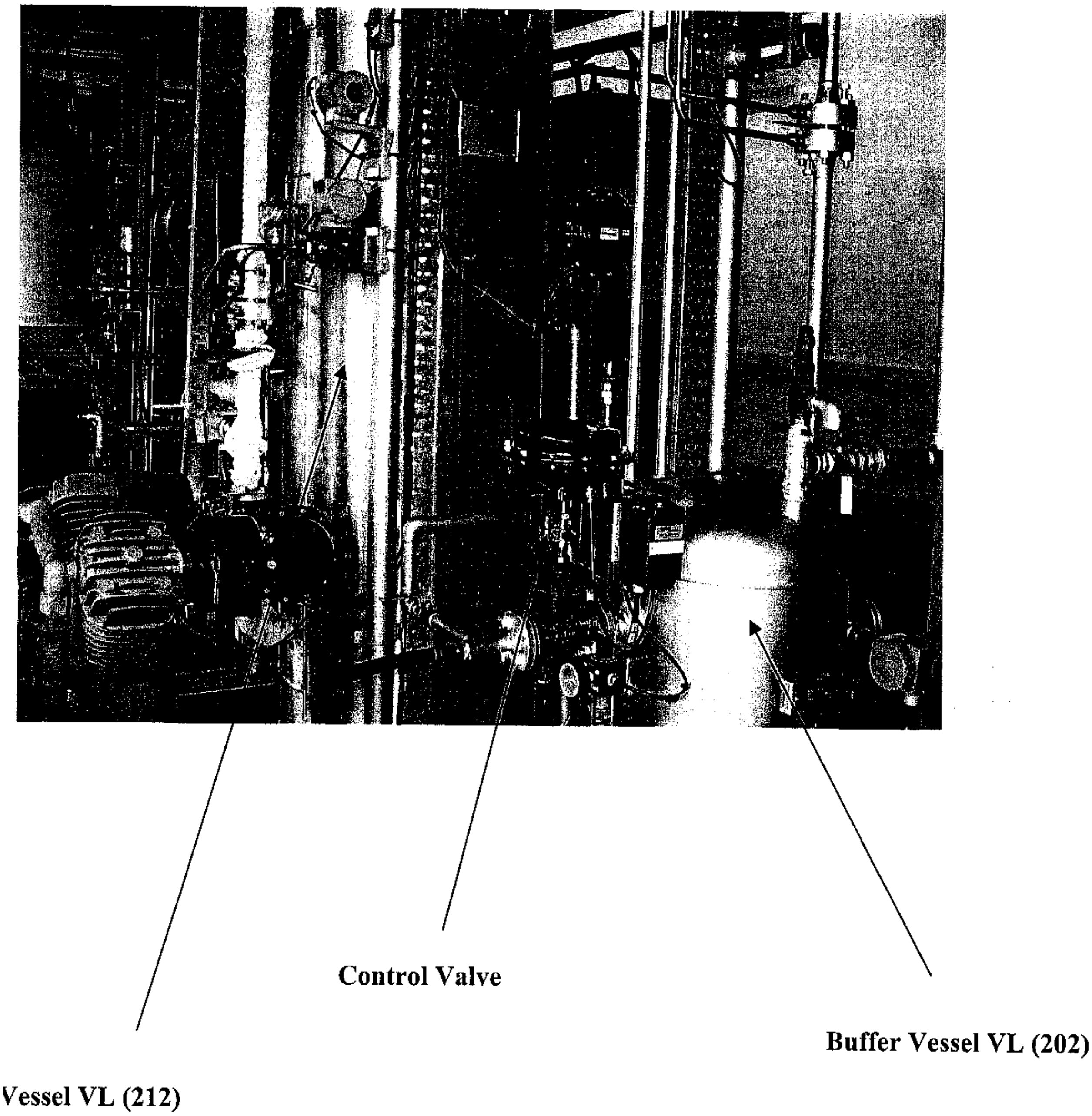


Figure 1: Gaseous Pilot Plant

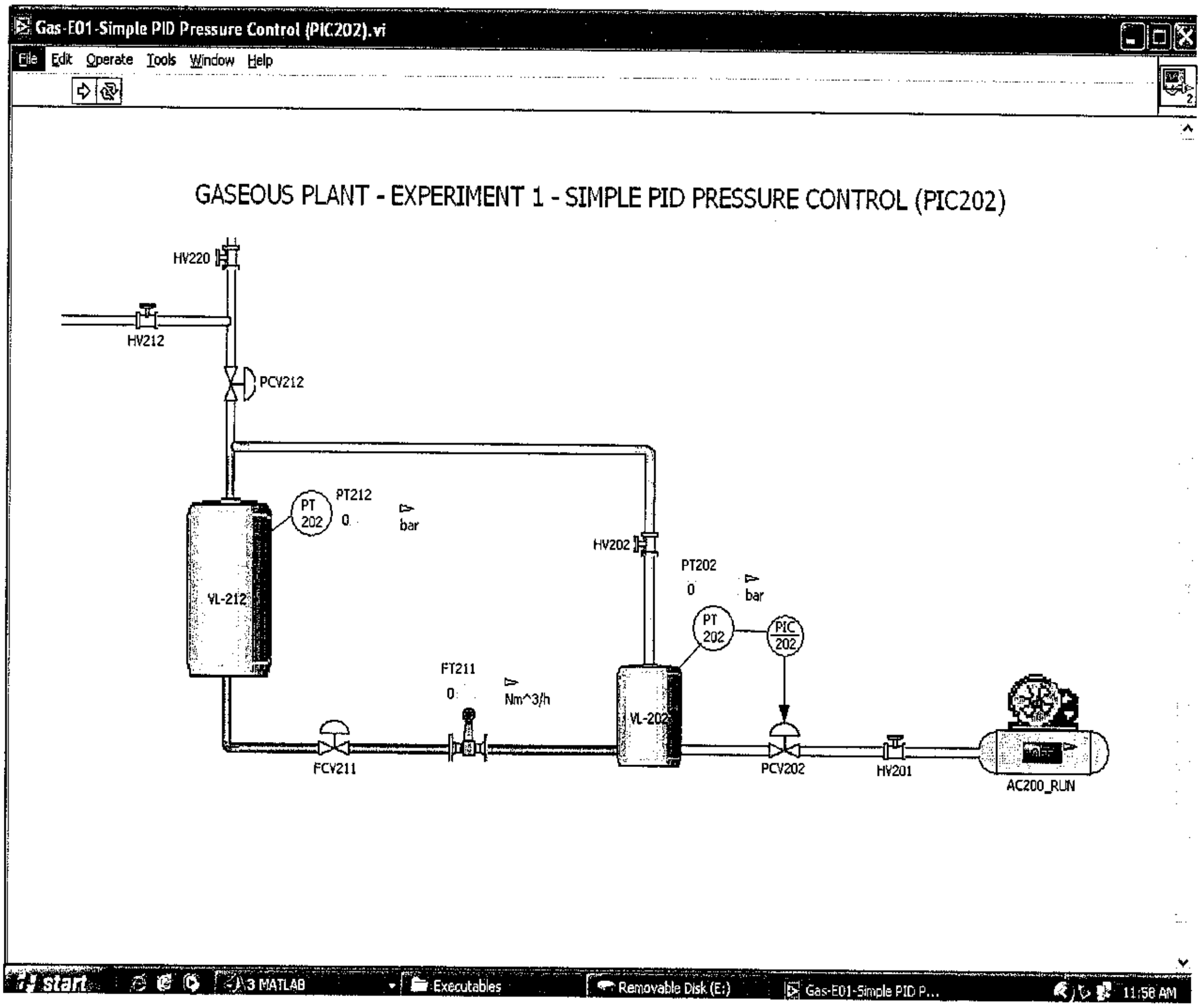
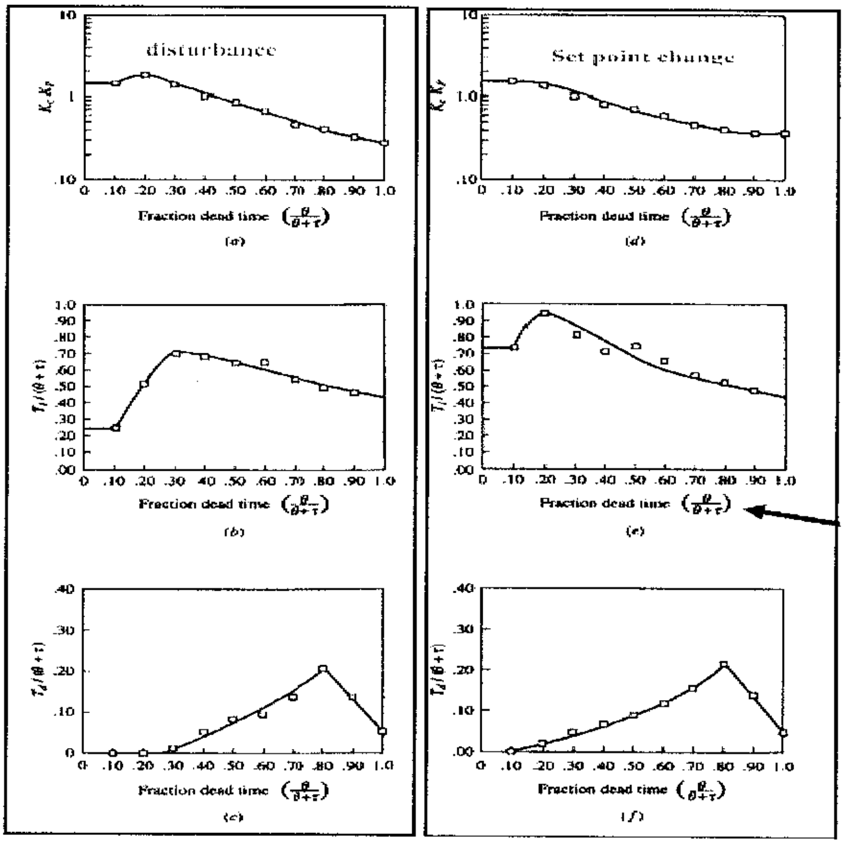


Figure 2: Simplified Pressure Diagram for Gaseous Pressure Plant



Fraction dead time

Figure 3: Ciancone Correlations for PID Controllers

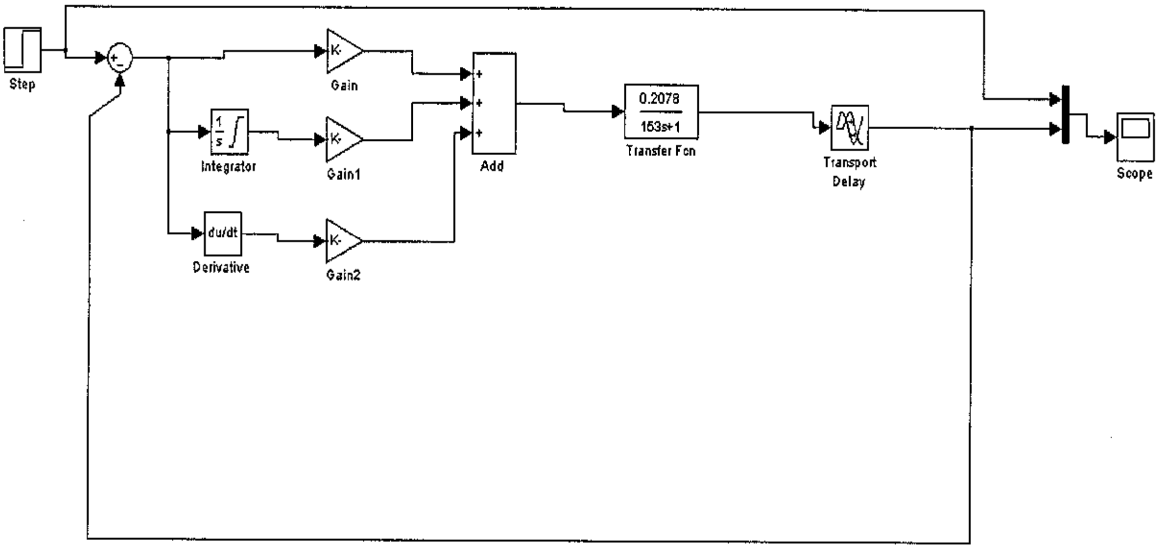


Figure 4: Closed Loop Block Diagram with PID Controller