

**A HYBRID INTELLIGENT CONTROL MODEL FOR
REGULATING pH IN INDUSTRIAL CHEMICAL
PROCESS**

BY

**NJOKU, DONATUS ONYEDIKACHI, M.Sc., PGDE, B.Sc.
20164999698**

**A THESIS SUBMITTED TO THE POSTGRADUATE
SCHOOL
FEDERAL UNIVERSITY OF TECHNOLOGY, OWERRI**

DECEMBER, 2022

**A HYBRID INTELLIGENT CONTROL MODEL FOR
REGULATING pH IN INDUSTRIAL CHEMICAL
PROCESS**

**BY
NJOKU, DONATUS ONYEDIKACHI, M.Sc., PGDE, B.Sc.
20164999698**

**A THESIS SUBMITTED TO THE POST GRADUATE
SCHOOL**

FEDERAL UNIVERSITY OF TECHNOLOGY, OWERRI

**IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE AWARD OF THE
DEGREE DOCTOR OF PHILOSOPHY (Ph.D) IN
COMPUTER SCIENCE**

DECEMBER, 2022

DEDICATION

This work is dedicated to Almighty God, my lovely wife Mrs Janefrances Ebere Njoku and my two daughters Felicity Chukwunonso, Perpetual and Chukwunoyerem and Chinemerem Eleena Njoku. This work is also dedicated to the sponsors of this Ph.D programme Jim Ovia-Nigeria Computer Society (NCS) and National Information Technology Development Agency (NITDA)

ACKNOWLEDGEMENTS

I sincerely express my profound gratitude to my Supervisors, Dr (Mrs). E.C. Nwokorie, Dr. Mrs J.N. Odii and Engr. Dr. O. C. Nwokonkwo for the wonderful criticisms, corrections and contributions made towards the success of this thesis.

My sincere thanks and appreciations to my Dean and Associate Dean School of Information and Communication Technology (SICT) Prof. (Mrs) U.F. Eze and Dr. H.C. Iwu for the dynamic and proactive manners in piloting our School to a greater height I also wish to thank my former Dean Prof (Mrs) G.A. Chukwudebe. My special thanks to my Head of Department Dr. Mrs J.N. Odii thanks so much for the speedily moves in taking Computer Science to the next level. I thanks you Prof. E.N. Erumaka and also all my wonderful and dynamic lecturers who have contributed immensely during this programme, My special thanks to Prof. E.O. Nwachukwu, Dr. C. N. Njoku, Prof. P.O. Asagba, Dr. O.F. Uzoh, Prof S.C. Inyama, Dr. S.A. Okolie, Dr. I.I. Ayogu, Dr. Ayiam Chukwuma, Dr. Mrs U.C. Onyemuche, Dr. Mrs C.G. Onukwughu. I will not forget to appreciate the department postgraduate coordinator Dr. S. A. Okolie and other academic staff Mrs J.C. Odirichukwu, Dr. Mrs L.C. Okpalla, Mr. S.O. Diala, Dr. Mrs T. Onwuama, Mrs F.O. Nwokoma, and non academic staff in the department. I thank my internal reviewer Dr. A.M. John-Otumu, also HOD of information Technology Dr. A. Otuonye as well as well as Dr. C.O. Ikerionwu.

I am also grateful to my dear wife Mrs Janefrances Njoku for all the inconveniences, to dearly beloved parents Mr. & Mrs J.J. Njoku and to all my sisters; Asagwara Susan, Esther, Chidimma, Nkechi, Ifunanya and to my special friend and brother Engr. Eze Paulinus I say to you all thank you.

I will not forget to thank the Dean of Postgraduate School, Prof. C.C. Eze and my able schedule officer Barr.(Mrs) B.U. Njemanze and entire PGS staff and my fellow postgraduate students I really appreciate all efforts towards the realization of this Ph.D programme. God bless you richly.

TABLE OF CONTENTS

Title	ii
Certification	iii
Dedication	iv
Acknowledgements	v
Table of Contents	vi
List of Tables	xi
List of Figures/Charts	xii
List of Abbreviation	xiii
Abstracts	xiv
CHAPTER ONE: INTRODUCTION	
1.1 Background Information	1
1.2 Problem Statement	5
1.3 Objectives of the Study	7
1.4 Justification of the Study	7
1.5 Scope of the Study	9
CHAPTER TWO: LITERATURE REVIEW	
2.1 Conceptual Framework	10
2.1.1 AI Applied as Estimators in Chemical Process Systems	10
2.1.2 Continuous Stirred Tank Reactor in Chemical Plant	12
2.1.2.1 Chemical Reactors	12
2.1.2.2 Continuous Reactor	14
2.1.3 Process Modeling	15

2.1.4	Challenging Issues in the Design of the Hybrid Systems	18
2.1.5	PID/PD Control	19
2,1.5.1	Proportional Control Mode	21
2.1.5.2	Integral Control Mode	22
2.1.5.3	Derivative Control Mode	22
2.1.6	Fuzzy Logic Controller	23
2.1.6.1	Nonlinear PD Control	26
2.1.6.2	Artificial Neural Networks	27
2.1.7	Learning Control	28
2.1.7.1	Switching Control	29
2.1.7.2	Fault Tolerant Control Methods	31
2.2	Theoretical Framework	32
2.2.1	Particle Swarm Optimization (PSO) Algorithm	34
2.2.2	Architecture of Intelligent Machine Systems	37
2.2.2.1	Hybrid Intelligent Machine System	39
2.2.2.2	Intelligent Supervision Based on Multi-Agent Systems	39
2.2.2.3	Supervisory Control System	41
2.3	Empirical Framework	43
2.3.1	pH System Architecture	52
2.3.1.1	Continuous Flow: pH Adjustment Systems	53
2.3.1.2	Batch Flow: pH Adjustment System	54
2.3.1.3	pH Neutralization	57
2.3.1.4	Fuzzy Inference System	59

2.4	Related Work	60
2.5	Summary of Related Work	81
2.6	Object Oriented Programming	88
2.7	Research Gap	89

CHAPTER THREE: METHODOLOGY

3.1.	Methodology	90
3.1.1.	Object Oriented Methodology	90
3.2	Analysis	93
3.2.1	Analysis of the Existing Study	93
3.2.2	Mathematical Model for the CSTR for pH Neutralization Process	93
3.2.3	State Space Equation	96
3.2.4	Study of the Existing System	98
3.3	Architecture of Existing Study	98
3.3.1	Challenges of Existing Study	99
3.4	Proposed System Architecture	99
3.4.1	Proportional Integral and Derivative (PID) Algorithm	100
3.4.2	Analysis of the Proposed System	101
3.5	Fuzzy Logic Control Design	105
3.5.1	Fuzzy Logic Intelligent Framework	107
3.5.2	Fuzzy Inference System	111
3.5.2.1	Mamdani Fuzzy Inference System	111
3.5.2.2	Sugeno Fuzzy Inference System	112
3.5.3	Flow Diagram of Fuzzy Logic Design	113
3.5.4	Advantages of Fuzzy Logic Control Algorithm	114

3.5.5	Simulink Fuzzy Logic Tool	115
3.6	Design of Fuzzy-PID Control Algorithm	116
3.6.1	Integration Fuzzy Controller in pH Neutralization	117
3.7	Detailed Architecture of the Proposed System	118
3.7.1	Benefits of the Proposed System	119
3.7.2	Design of the Proposed Fuzzy-PID Compensator	119
3.7.3	Algorithm of Proposed System	124
3.8	Unified Modeling Language (UML)	129
3.8.1	Flowchart of the Proposed System	132
3.9	Integrated Development Environment Design Tool	134
3.9.1	Netbeans Integrated Development Environment (IDE)	133
3.9.2	pH Data for Water treatment	135

CHAPTER FOUR: RESULTS AND DISCUSSION

4.1	Implementation	136
4.2	Programming Language Used	136
4.2.1	System Requirements	136
4.2.1.1	Hardware specification	136
4.2.1.2	Software Specification	136
4.3	Results	137
4.3.1	Unit Step Response of Uncompensated Process	137
4.3.2	Unit Step Response of PID Compensated pH Process	138
4.3.3	Unit Step Response of Fuzzy-PID Compensated pH Process	132
4.4	Performance of Analysis in Terms Unit Step Input	139

4.4.1	Performance Indices	141
4.5	Performance Responses for pH Mediums using PID and FPID Control Model	142
4.6	Validation of Fuzzy-PID Unit Step Response for varying pH value	145
4.7	Discussion of Results	148
4.8	Interface of Application for pH Neutralization	149

CHAPTER FIVE: CONCLUSION AND RECOMMENDATIONS

5.1	Conclusion	152
5.2	Recommendations	153
5.3	Contribution to Knowledge	154

REFERENCES	155
-------------------	-----

APPENDIX A: SOURCE CODES	154
---------------------------------	-----

APPENDIX B: SAMPLE OUTPUTS	203
-----------------------------------	-----

LIST OF TABLES

Table		Page
2.1	Comparisons of several hybrids System Structure (Jarinah et al (2015))	60
2.2	Summary of Related Work	82
3.1	Summary of the Elements of PID Controller	101
3.2	Fuzzy Rules	121
4.1	Step response Performance Parameters.	140
4.2	Remark for each of the Simulation result.	141
4.3	Error Performance Responses for pH value-5	141
4.4	Error Performance Responses for pH value-7	142
4.5	Error Performance Responses for pH value-9	143
4.6	Step responses performance parameters at varying pH values.	146

LIST OF FIGURES

Figure	Page
2.1 Block Diagram of fuzzy –PID Controller (Darius and Sivagamasundan, 2016).	14
2.2 CSTR with Cooling Jacket. Prabhu et.al (2014).	15
2.3 Closed Loop Control with PID Controller (Smuts, 2002)	19
2.4 Fuzzy Logic Controller (FLC) Architecture (Malhotra and Sodhi, 2011).	23
2.5 Fuzzy PID Control Structure (Zhang and Wang, 2013).	25
2.6 Closed- loop control System with Single Neuron (Chopra et al., 2014)	26
2.7 A Switching Control System and the Interaction. Antsaklis (1998).	29
2.8 Block diagram for PID Intelligent controller tuning using GA (Eberhart, 1995).	33
2.9 Iterative Particle Dimension for PSO (Kennedy, 1995)	37
2.10 Architecture of Intelligent Machine systems (Puren, 2005).	38
2.11 Generic Scheme of an Agent (Jennings, et al, 1999)	42
2.12 Block diagram representation of decentralized control scheme for the ideal CSTR (Darius and Sivagamasundan, 2016)	47
2.13 Block diagram representation of decoupled control scheme for the ideal CSTR (Darius and Sivagamasundan, 2016)	48
2.14 Modelling of CSTR plant with one MPC Controller (Debnath and Tripsthi 2016)	51
2.15 Continuous pH System Architecture	54
2.16 Batch Flow: pH Adjustment System	56

2.17	pH Scale Measurement	57
2.18	Schematic of a pH Neutralization Process (Kambale et. al, 2015)	59
2.19	Cascade control scheme (Dostàl et al 2015)	65
2.20	CSTR with Adaptive Control (Singla, 2010)	68
2.21	Control and Optimization Structure (Soukkou et al 2008)	70
2.22	Tuning of PID-GA Controller (Yadav and Trivedi 2014)	71
2.23	Tuning of PID-PSO Controller (Yadav and Trivedi 2014)	71
2.24	TITO Self-tuning Control Scheme configuration of the Proposed Control Design Method (Chalupa, 2009; Ang et al, 2018)	76
2.25	Proposed CSTH TITO Control Scheme Configuration (Ang et al, 2018)	77
3.1	Object Oriented Methodology Framework	93
3.2	pH Process Model	95
3.3	Existing System Architecture (Ram et al, 2016)	99
3.4	Proposed System Architecture	100
3.5	Block Diagram of PID	102
3.6	Matlab PID GUI Tuner	103
3.7	Simulink Model of pH Process with PID Compensator	105
3.8	Elements of a Fuzzy Logic Control Algorithm	107
3.9	Flow Diagram of Fuzzy Logic Control Algorithm	113
3.10	Fuzzy Logic Block Simulink	115
3.11	The Basic Tools in the FIS	116
3.12	Fuzzy logic compensated pH process	117
3.13	Detailed Architecture of the Proposed System	118
3.14	CSTR Fuzzy Inference System (FIS)	121
3.15	Error Input	121

3.17	Change in Error	122
3.18	Fuzzy Output	122
3.19	Simulink model of the proposed pH process with Fuzzy-PID	124
3.20	Fuzzy inference System Surface view	127
3.21	Flowchart of the proposed PID Controller	125
3.22	Matlab IDE Window	126
3.23	MATLAB/Simulink GUI environment for simulation	127
3.24	Netbeans IDE	128
3.25	Proposed System Flowchart	129
3.26	pH Data for Water treatment	135
4.1	Unit step Response of uncompensated pH process	137
4.2	Step response of PID compensated pH process	138
4.3	Step response of Fuzzy-PID compensated pH process.	139
4.4	Analyzed Response of System State.	140
4.5	Error Performance Index PID, and FPID, pH-5	142
4.6	Error Performance Index PID, and FPID, pH-7	145
4.7	Error Performance Index PID, and FPID, pH-9	144
4.8	Step response at pH setpoint 5 (Acid medium)	145
4.9	Step response at pH setpoint 7 (Neutral medium)	145
4.10	Step response at pH setpoint 9 (Base medium)	146
4.11	pH responses of Acid Base and Neutral Point	147

4.12	Login Interface Window for pH Neutralization Process	149
4.13	Stages of Neutralization Process of Water Treatment Plant	149
4.14	Neutralization Process of Water Treatment Plant	149

LIST ABBREVIATIONS

AI- Artificial Intelligence

ADRC- Active Disturbance Rejection Control (ADRC)

AWPSO- Adaptive Weighted Particle Swarm Optimization

ARO-Asexual Reproduction Optimization

BIBO-Bounded-Input Bounded- Bounded output (BIBO)

ANN- Artificial Neural Network

ANFIS- Adaptive Neurofuzzy Inference Systems

CSTR-Continuous Stirred Tank Reactor

CSTH-Continuous Stirred Tank Heater

COG-Centre of Gravity

CV- Constant Velocity

ERN- Externally Recurrent Net

ES-NN-Expert System Neural Network

pH- Potential Hydrogen

ES-Expert Systems

EKE- Extended Kalman Filter

FLC Fuzzy Logic-base Control

FL- Fuzzy Logic

FTLR NN-Focused Time Lagged Recurrent Neural network

FNN- Fuzzy Neural Network

FPI- Fuzzy proportional integral

FIS-Fuzzy Inference System

FuREAP-Fuzzy-Rough Set

FTC-Fault-Tolerant Control

FOPTD-First Order Plus Time Delay

FOPID- Fractional Order Proportional Integer Derivative

FFN-Forward Neural Networks

HNN-Hybrid Neural Network

HCl- Hydrochloric Acid

GCE = Gain of Change in Error

GE = Gain of Error

GU = Gain of control input

GIE = Gain of Integral Error

ΔE - Change in Error

NFPI-Nonlinear Fuzzy Proportional Integer

SAHNN-Structure Approaching Hybrid Neural Network

NL-Negative Large

NM-Negative Medium

NS-Negative Small

HCS -Hybrid Control System

MIMO-Input Multi-Output

MR- Magnetor Heological

MRAC- Model Reference Adaptive Control

MIT-Massachusetts Institute of Technology

ISE- Integral Square Error

IAE-Integral of Absolute Error

ITAE- Integral of Time Absolute Error

NPD- Nonlinear Proportional Derivative

SISO- Single-input single-output

SA-Stability Analysis

MPC-Model Predictive Controller

MMPC- Multiple Model Predictive Control

IMC- Internal Model Control

LSSVM-Least Square Vector Machine

SIR- Susceptible Infection- Recovered

SOPTDZ-Second Order plus Time Delay System with a Zero

P-Proportional

PSO Particle Swarm Optimization

I-Integral

KQML -Knowledge Query Manipulation Language

PI- Proportional and Integral

QbD- Quality by Design

IRN- Internally Recurrent Net

PID- Proportional Integral and Derivative

RBFN-Radial Basis Function Networks

MFNN -Multilayer Fuzzy Neural Network

MIMO-Multi-Input Multi-Output

GUI-Graphical User Interface

PLC-Programmable Logic Controller

ICA-Imperialist Competitive Algorithm

IO-Input-Output

IT-Information Technology

IDE- Integrated Development Environment

OOM-Object Oriented Methodology
OOP- Object Oriented Programming
UML-Unified Modelling Language
SISO-Single-Input Single-Output
SSADM-Structured System Analysis and Design Methodology
NMSE-Normalized Mean Square Error
NaOH- Sodium Hydroxide
OMG- Object Management Group
ODE-Ordinary Differential Equation
MSE- Mean of Square Error (MSE)
TITO-Two-Input Two-Output
SSE-Sum of the Square Error
PS-Positive Small
PM-Positive Medium
PM-Positive Large
ROFLC-Robust and Optimal Fuzzy Logic Controller
MATLAB-Mathematical Laboratory
MF-Membership Function
GA- Genetic Algorithm
NNMPC- Neural Network Model Predictive Controller
ZN-Zeigler Nichols
ZE-Zero Error

ABSTRACT

This work presented a hybrid intelligent control model for pH regulation in industrial chemical process. The problems of most chemical process are due to the complexities in manipulating non-linear and uncertainty variables. Again, the conventional models used to control processes are prompt to disturbance. The dynamic model approach adopted the first order plus delay time (FOPDT) process transfer function to analyzed the system. The research adopted proportional integral and derivative (PID) and Fuzzy logic system the hybrid compensator model for the developed system. PID model was designed to be tuned in MATLAB window to obtain the variable parameter the Fuzzy Logic system were further integrated with PID to control perturbation and uncertainty among the parameters. The fuzzy logic was designed using Fuzzy Inference System (FIS) in the MATLAB software tool. The gains of the PID algorithm were combined with the Fuzzy Logic system to produce the proposed hybrid intelligent Model called Fuzzy-PID. The model was analyzed in MATLAB/Simulink environment to examine the effectiveness of the hybrid system. Simulation was initially conducted considering loop response of pH system in terms of step input without the addition of the proposed intelligent hybrid model, and the step response shows that the system was not able to reach desired pH set value. A PID controller was simulated in a closed control loop of the pH neutralization process and the step response indicated that the desired pH value was achieved. But the PID control system showed some degree of instability uncertainties. The Fuzzy-PID was implemented as part of the component of the closed loop pH control system and the step response obtained revealed that an improved performance was obtained and the system attained the desired pH value with no instability. The process was modeled in MATLAB/Simulink environment to examine the effectiveness of the hybrid system. The application was developed using JAVA, and Netbean. The work adopted Object Oriented Methodology in the development of the application. Simulation was initially conducted considering loop response of pH system in terms of step input without the addition of the proposed intelligent model. In order to validate the effectiveness of the proposed system, simulations were conducted in acid, neutral and base media with pH setpoint target of 5, 7 and 9 respectively on settling time, rising and overshoot as well the error performance analysis with Integral Absolute Error (IAE), Integral Square Error (ISE) and Integral Time Absolute Error (ITAE). It was observed that less error performance indices were offered by the hybrid model (FPID) control pH neutralization process compared to PID controller in various media –acid, neutral and base. However, the results obtained showed that the proposed hybrid system was able to maintain the setpoint pH values in the various media than individual model,.

Key words Algorithms; Intelligent; Neutralization; Transfer Function; Setpoint,; pH; PID; Fuzzy; Simulink; Compensator; Regulating

CHAPTER ONE

INTRODUCTION

1.6 Background Information

Modern chemical processing plants comprise of process units that are configured in a network. The various units of an industrial Chemical Process are interconnected with one another, usually through quite complicated network structures. These process units are characterized by control loops which interact to obtain an optimized product quality (Tippett and Bao, 2015).

According to Kishore, Rosdiazli, Mohd, Tranduc and Sabo (2018) stated that the fundamental model of pH neutralization process plant was developed by taking into consideration the overall plant dynamics. Tippett and Bao (2015) control chemical processes are characterized by their scale, strong interactions and different dynamics of each process unit. In practice, the application of control system design for industrial chemical processes are characterized by nonlinearity, uncertainty and larger number of environment factors which changes continuously.

Moreover, in the work done by Park, Let, Quynh, Long and Hon, (2021) started that in enhancing the performance of a model base intelligent hybrid system with suitable gains with conventional Control models will provide an improved learning ability and flexibility of the system for better operative performance.

Today, effort is being made to ensure that the product through continuous processing techniques is of high quality. Ramasubramaian and Thirumarimuru (2022) developed a hybrid controller for temperature control, based products through a continuous processing technique used in industry, thus makes it possible for the online/inline combination of measurement tools in the form of sensors and the introduction of efficient feedback control system (Maitraye,

Ravendra, and Robit, 2014). In the work by (Saravanan and Kumar, 2019) Controllers act as the backbone of process control and automation-based industries. Controllers are often used because of their robustness and easy accessibility. Though process industry varies from one area to another based on the activities of production involved, it is expected to run at optimal performance. However, an industrial process may run at maximum throughput, this important point is worth considered in process industry in plant optimization by process control. This control should actually be flexible, that is, be adaptable to the condition or situation.

In many chemical process industries like Refining industries, Waste Water Plant, Digestion Process, Biochemical Processes and Mechanical Engine's Coolant system, pH value has great importance. pH measurement and controlling is extensively used for; Quality control of the product and its uniformity; Corrosion inhibition where pH monitoring is used in conjunction with oxygen control to minimize corrosion in high pressure boilers and other water streams; Effluent pH controlled by neutralization of liquid discharge in to public streams (Ghatule, 2017).

Chemical processing networks are usually designed with steady state efficiencies and capital costs in mind (Liu *et al.*, 2010; Tippett and Bao, 2015). From their statements it entails that it is difficult to design a plant wide- wide control systems for such processes. The challenges are caused by (Tippett and Bao, 2017); high nonlinearity in the process systems, which greatly increases the analysis and associated control design complexity; there exist large scale systems which have strong interaction effects as a result of complex network arrangement; significant time-scale is allowed to separate between and within, unit processes as a result of different physical and chemical process. For instance, mass transportation and chemical reaction processes could take place at different time and also the issue of plant model misrepresentation (or approximation) as

often time complex physical chemical process model obtained from first principles are usually simplified for control design purposes.

In Control model design, model error due to approximation of complex processes for the purpose of design has been highlighted as part of the challenges to industrial processes, also different control techniques have been successfully implemented to effectively produce desired results. However, the conventional control techniques are increasingly being replaced by more sophisticated control strategies. When the process can be reasonable described or represented accurately in advance, the application conventional control like the proportional integral and derivative (PID) controller become suitable. Nevertheless in a situation that the process dynamics are difficult to model precisely probably due to environmental uncertainties, it may be difficult to use the conventional controller design techniques (Kumar, Rana, and Sinha., 2011), for this reason, the control of systems with complexities, uncertainties and nonlinearities has become an issue of considerable importance to researchers and different advanced techniques have been proposed in literature (Kumar *et al.*, 2011).

A big drive has been seen in the academic community to design new control systems, either by traditional or contemporary methods. Introducing an ingenious control system can be the key factor in improving performance as well as for dealing better with the challenging features of nonlinear and complex processes. Generally, when implementing conventional control systems, a reasonable performance is attained over a narrow operating range. However, the range of process tasks is a prerequisite, nonlinearity become more evident and hence, the control performance may be sacrificed. In line with control performance, (Mujahed, 2022) constructed and evaluated a control mechanism for Fuzzy fractional Order PID(FOPID) to compare the intelligence among control schemes. The result of FOPID produces better performance. Application of intelligent techniques in process control systems with reasonable level of

autonomy should enhance performance even in the presence of significant uncertainties and nonlinearities in the systems and environment. This should be able to compensate for certain failures without human intervention. These systems have evolved one time from conventional control systems combining intelligent methods to more advance techniques involving hybridized intelligent systems. There are excitement in the field of intelligent process control which has attracted more attention with appreciable progress in the area of Fuzzy control, Neural network, Genetics algorithms and Expert systems etc (Kumar *et al.*, 2011). Also in the research carry out by Snehal, Kambale, Sebastian and Zope (2015) on controllers used in pH regulating process, the work centred on different control strategies using PID, fuzzy based model, neural network based model for pH neutralization process.

According to (Aitbayevich *et al.*, 2015), it is possible to exploit the advantages of traditional (or conventional) techniques and algorithms with analytical tool of artificial intelligence (AI) theory to generate a new algorithm of control known as hybrid control system (HCS). This offers a new dimension to process control performance in industry. In practice, realistic events or activities that requires the tasks of management and control, optimization and modelling would be well handle if intelligent techniques for monitoring and control have given practical way for various applications in science and engineering with external intervention.

As a result of the increasing scale and complexity of modern chemical processes, the number of output or controlled variables is significantly large (Ming and Zhao, 2017). Also according to (Thulasi *et al.*, 2017), many of these processes in chemical industries are of complex dynamic and highly non linear because of the many processes and manipulated variables. Such systems with more than one

input (or manipulated) variable and more than one output (or measured) variable is referred to as multi-input and multi out system.

Most of the conventional controllers are used for the systems with linear time invariant (LTI) applications (Prabhu and Bhaskram, 2014). In practice, the physical properties of the process such as wear and tear are the cause of changes in functional parameters and nonlinear characteristics which are not negligible. However the aim will be to deal with system and their uncertainties in real applications (Soheilrad et al., 2012; Prabhu and Bhaskaram, 2014). For this reason, the application of intelligent and adaptive control algorithms (or controllers) with working parameters for nonlinear system in real environment like in chemical industrial process is of a great importance (Rahmat *et al.*, 2011; Prabhu and Bhaskaram, 2014).

Due to the global ambition for the more reliable attainment of high product quality, more efficient use of energy, tighter safety as well as environmental regulations, industrial processes have evolved over recent years into very complex, highly nonlinear and integrated systems. Rigorous demands like these naturally lead to more difficult and challenging control problems for today's industrial control engineers; problems requiring more efficient solutions than can be achieved only by conventional techniques. It also requires inter- and cross-disciplinary research and development, in addition to collaboration in both industry and academia. Cooperation between control and other disciplines has been consistently fruitful (Karl *et al.*, 2014) in this work, a hybridized intelligent system for industrial chemical process intended to be developed.

1.7 Problem Statement

The work done by Kishore *et al.*, (2018) explained that as industrial chemical process and their control systems become more complex in its activities,

achieving higher efficiency and improving the control performance becomes more challenging. Also the control of systems with difference dynamics that have complexities in design, uncertainties and nonlinearities has become an issue of considerable challenges was also pointed out by Kumar *et al.*, (2011). In the work done by the (Park *et al.*, 2021) also started the problem encountered with use of PID controller in modelling PID-Multilayer Fuzzy neural Network design for quadcopter was that PID was ineffective in the control model and again, the conventional PID was relatively complicated and time consuming because of its level of dependents on external disturbances and dynamic model of the plants.

Also, the process become more complex to control and maintain at steady state. For this reason, the need to develop a system which will assist and enhance the general production process in industries becomes necessary. Again, most Chemical reactors are indispensable and have influential factors in their process activities which may be in the form of continuous or batch processing which formed a strong determinant on the input and output process.

However, in Snehal *et al.*, (2015) stated that the challenging problem of pH control in the chemical process industry is difficult because of their inherent nonlinearity, high sensitivity at neutralization point and varying gains when uncertainties inflows and concentrations of neutralization agents are present. Moreover, typical industrial process systems are prone to certain challenges of presence of multi- variables, which interacts between manipulated, controlled variables, disturbances, process delays and other constraints on the input and output variables. Moreover, many different control algorithms and techniques have been developed and applied to chemical systems, these conventional control techniques used in most chemical plants design such as the Proportional Integral Derivative (PID), though it does not provide or guarantee an optimal performance when applied to process systems having kinetic reaction and thermodynamic interactions, adequate system performance are not provided and as such fails to compensate for the entire operating factors. In order to exploit the benefits

offered by hybrid intelligent control model that will provide efficient and optimal control performance for an industrial chemical process, this research work proposed a hybrid control model called Fuzzy-PID for regulating pH neutralization process in water treatment.

1.8 Objectives of the Study

The primary objective of this work is to develop a hybrid intelligent Control Model for regulating pH process in water treatment plant in chemical industry.

The specific objectives are to:

1. Develop a hybridized model for an industrial chemical process to handle uncertainties parameters for the control system.
2. Perform simulations on the developed hybrid system Model using MATLAB/Simulink to ascertain their effectiveness in terms of response to a setpoint signals; in rising time, settling time and overshoot.
3. Compare the behaviours of uncompensated system with the compensated proposed system performance based on rise time, settling time and overshoot parameters.
4. Develop software application for regulating pH in industrial water treatment plant, for Federal University of Technology, Owerri.

1.9 Justification of the Study

The use of conventional PID controllers in the control loop for industrial processes has been enormous because of its simple structure and acceptable performance this was proposed in by (Ianni and Rossi, 2021). The work also justified the claimed that PID controller's mechanisms which are used in various applications of complex systems can be modelled even in social restriction on the outbreak of COVID-19 epidemic by government. Certain assumption was set as to obtain feedback in understanding the behaviour of the infected pollution during

the intermittent lockdown policy in order to establish an optimal control. In Upadhye, Sebastina and Mobeen (2015) clearly state the need of pH regulation process is being affected by highly nonlinear and varying process, hence cannot be controlled efficiently using conventional controllers. Hence, the need to apply fuzzy logic controller is capable of providing optimized solution for such process. Fuzzy logic controller can be used with the conventional controller to obtain better performance for pH neutralization and regulation process.

When properly tuned, PID controllers give improved time response and frequency response. However, with increase in demand for efficient and more effective control of industrial processes for systems with uncertain dynamics, time delays and nonlinearity, the performance of traditional controllers tend to degrade this was proposed according to (Park *et al.*, 2021) their work further added that the application of Multilayer Fuzzy Neural Network(MFNN)in the designing of attitude of quadcopter using the PID-MFNN model eliminated the need for mathematical system model and complicated calculation and therefore improving the learning and flexibility on the use of Fuzzy Neural Network..

The use of intelligent control is based on the fact that human intelligence is included into the architecture of the controller (control algorithm) to enable the emulation of human behaviour in the control decision. The human expert knowledge is necessary based on the heuristic information acquired in relation to the system process. The Fuzzy logic base control (FLC) is an intelligent based control that offers a powerful tool for controlling complex system. It provides a control algorithm that converts expert knowledge into an automatic control technique. Fuzzy logic approximate information and is suitable for nonlinear systems control and useful for modelling system of complex dynamics where an inaccurate model exists or system where ambiguity or vagueness are common.

Although the applications of PID or FLC have been somewhat successful, however there are challenges of not achieving optimal performance with the

industrial processes getting complicated. The need to have a combination of more than one control strategy becomes necessary. Integration of two or more than controllers or algorithms to control a process gives a hybrid controller. This work Oumar, Atif , and Khan (2013) supported that pH neutralization and regulation process mainly consist of pH measurement of an acid base chemical reaction where hydrogen ions and hydroxide ions are neutralized with each other to form water, while the other ions involved remain unchanged. (Oumar *et. al.*, 2013) the hybrid control method widely used pH control feed forward in compensating any disturbance before it affect process control consist of The use of intelligent method based on fuzzy logic (FL) Artificial Neural Network (ANN), Adaptive Neuron Fuzzy Inference System (ANFIS) or Genetic Algorithm (GA) combining any PID controller or both gives a hybrid intelligent syste

1.5 Scope of the Study

This work focuses on developing a hybrid model for industrial chemical process for regulating pH neutralization process. The model considered a Continuous Stirred Tank Rector (CSTR) that supports the continuous process system. The model was represented in terms of transfer function for mathematical model anaysis. The main focus is to design a system that will regulate and improve the system performance using a hybrid control model thjat is PID, FLC and Fuzzy-PID controller that was proposed. The developed model was done using MATLAB/SIMULINK simulation environment and the obtained results analyzed. A standalone application system to regulate and determine the pH values for water treatment plant. However, this work did not consider the effect of temperature, pressure of the chemical plant.

CHAPTER TWO

LITERATURE REVIEW

2.1 Conceptual Framework

This section discussed the concepts of hybrid intelligent systems and its models. Currently, intelligent and smart system are promising research field for modern computational intelligence concerned with the development of the next generation of intelligent automated systems especially in the applications of industrial process activities. A fundamental stimulus to the investigations of hybrid intelligent systems is the awareness in the academic communities that combined approaches might be necessary if the remaining tough problems in artificial intelligence are to solved existing control strategies for chemical reactors are presented with focus on Continuous Stirred Tank Reactor (CSTR). The usefulness of Artificial Intelligence (AI) in process systems is not constrained to modeling and control only, but also as estimators to estimate the unmeasured parameters as an alternative to the conventional observers and hardware sensors. These estimators, also known as software sensors have been successfully applied in many chemical process systems such as reactors, distillation columns, and heat exchanger due to their robustness, simple formulation, adaptation capabilities and minimum modeling requirements for the design. However, the various types of AI methods available make it difficult to decide on the most suitable algorithm to be applied for any particular system.

2.1.1 Artificial Intelligence Applied as Estimators in Chemical Process Systems

The usage of Artificial Intelligence algorithms as estimators in chemical process systems is becoming popular due to their robustness, simple formulation, and easy-to-design and flexible adaptation capabilities.

They also do not require full knowledge of process kinetics, which are at times difficult to be obtained from the highly nonlinear behaviour of chemical processes themselves. The AI algorithms that have been applied include the artificial neural network (ANN), fuzzy logic, genetic algorithm (GA), expert systems (ES) and hybrid systems. However, the variety and unique features of these algorithms make it difficult to decide on the suitable method to be used in any particular system. Therefore a survey of each algorithm applied as estimators in chemical process systems are important to highlight the significance of each algorithm and guide researchers towards the design of the AI-based estimators for any particular chemical process system.

For estimating parameters in chemical process systems, several types of ANN can be used such as the feed forward neural networks (FFN), internally recurrent net (IRN), externally recurrent net (ERN), radial basis function networks (RBFN), and the shape-tunable neural network (MNN). Apart from ANN and fuzzy logic, expert systems (ES) and genetic algorithms (GA) have also been applied to build estimators in several chemical process systems. ES is an intelligent system that can provide information and expert advice and is also known as Knowledge-Based-Expert-System as it depends on large amount of knowledge to solve problems. There are basically five types of Expert System namely Rule-Based ES, Frame-Based ES, Case- Based ES, Fuzzy ES and Neuro-Fuzzy ES. Rule-Based ES is an expression of knowledge in the form of rules to solve problems.

It is divided into two parts which are IF: Antecedent (premise/condition) and THEN: Consequent (conclusion/action). If there are many antecedents, the operator AND as well as OR is used to combined them all. These rules are the most popular type of knowledge representation. ES has been applied as estimators due to its ability in expressing relations, recommendations, directives, strategies and heuristics. There are five basic components of the Expert system

namely; knowledge base, database, inference engine, explanation facilities and user interface.

2.1.2 Continuous Stirred Tank Reactor in Chemical Plant

There are many qualities of literatures that have been presented by researchers in the area of modelling, controlling and parameter estimation of chemical reactors. Academic and industrial focus has been placed upon intelligent systems applications, in the area of chemical reactors as a results of the challenges or difficulties in operating such processes, for example, the presence of high nonlinearity of systems.

There are different types of modeling and control techniques for chemical reactors. Nevertheless, CSTR is specially reviewed in this chapter.

Development and improvement of the control techniques for the CSTR has been a concern for research, the objective is aim at developing hybrid intelligent system for improving the performance of a chemical process.

2.1.2.1 Chemical Reactors

Reactors are the dispensable and leading influential factor for any industry in the field of chemical engineering (Prahhu and BhasKaram, 2014). There are different types of reactors used in chemical industries such as the Continuous Stirred Tank Reactors, batch reactors. These reactors form the heart of the any manufacturing process plant, where a chemical change is taking. In batch reactors, reactants are charged at the start of the reaction and products are then removed from the reactions at the end. CSTR, reactants are continuously charged and product are continuous by removed (Danis and Sivagamasundan, 2016).

The design of different control techniques to control different control parameters of the reactors have been performed due to the widespread application of chemical reactors. In the research done by (Vishnoi *et al.*, 2012) stated that

dynamic systems of chemical reaction processes are usually nonlinear and as such require to be guided by a mechanism that will ensure that the desired product quality is attained with cost effectiveness in mind.

Industrial process plants integrate process control strategy as part of the production loop. The mechanism of control employed in chemical processes can be in form of a traditional control, an intelligent control, or a combination of both (hybrid-control).

As a device in which chemical reaction takes place, chemical reactor represent the most essential unit of a chemical process system used for unit operation (Vishnoi *et al.*, 2012). According to Vishnoi *et al.* (2012), chemical reactors can be classified according to different properties given by: Reaction Phase; Operating Modes or according to the reaction phase as:

Heterogeneous reactors are based on the operating mode such as: Continuous Stirred Tank Reactor (CSTR); batch Stirred tank reactor; semi batch; tubular reactor. In order to design a chemical reactor, there are factors to be considered such as; reactor overall size, products emerging from reactor, reactor temperature, reactor pressure, rate of chemical reaction, activity and mode of catalyst, and stability and controllability of reaction (Vishnoi *et al.*,2014).

According to Nabi (2015), chemical reactors can classified into three types with respect to three ideal models as continuously stirred tank reactor, tubular reactor and batch and semi-batch reactor. The most frequently used types of reactors in chemical industry are the continuous and semi batch reactor (Bequett, 2003; Nabi, 2015), with the continuous reactors being preferred mainly in the chemical industry because of their potential for good quality control and large scale production capacity (Artin and Remoundaki, 2005; Nabi, 2015).

This work does not aim to investigate new facts about chemical reactions or processes of their physical properties, but to design a system that leads to an improved process for producing quality products in continuous reactor.

2.1.2.2 Continuous Rector

Many chemical plants use continuous stirred tank reactor (CSTR) in the form a single tank or more tanks in series Damish, Almesfer and Rashid, (2015). The computational efficiency of process system has been increased by the study of dynamic characteristics of CSTR. The CSTR is essentially an indispensable unit in many chemical plants, where it frequently serves multi-purpose production objective form and special chemicals (Pablo and Ricardo, 2009). The CSTRs are commonly employed in the production of industrial chemicals, when therefore, a chemical is high and very consistent (Alejandro *et al.*, 2010).

According to Zhao *et al.* (2015), the operation of CSTRs are performed around a certain stable point which is linked to the optimal output or optimal production process to achieve high conversion rate and maximize economic gains. In chemical process control point of view, the dynamics of CSTRs are regarded as highly nonlinear. Some of the notable dynamic characteristics of CSTRs are relatively degree, unmeasured states zero dynamics. The dynamics characteristics make the design of the process algorithm for effective CSTRs very challenging, especially in the presence of the external perturbation and/or system uncertainty (Seki and Nka, 2008; Zhao *et al.*, 2015). Also, another challenging from the economic point of view that is facing process engineers will be the variable cost reduction while keeping product quality (Pablo and Ricardo, 2009).

In order to enhance the operation of the CSTR process system, control algorithm has been made an integral part. An automatic controller must be able to enhance the process operation within a given range of operating condition (Danius and Sivagamasundan, 2016). Many control strategies have been used to facilitate the

performance of the CSTR process in industry. Such controllers are Proportional Integral (PI) or Proportional Integral Derivative (PID) Controllers, Model Predictive Control (MPC) and Fuzzy logic control. Figure 2.1 shows a closed loop block diagram of a CSTR using Fuzzy-PID Controller.

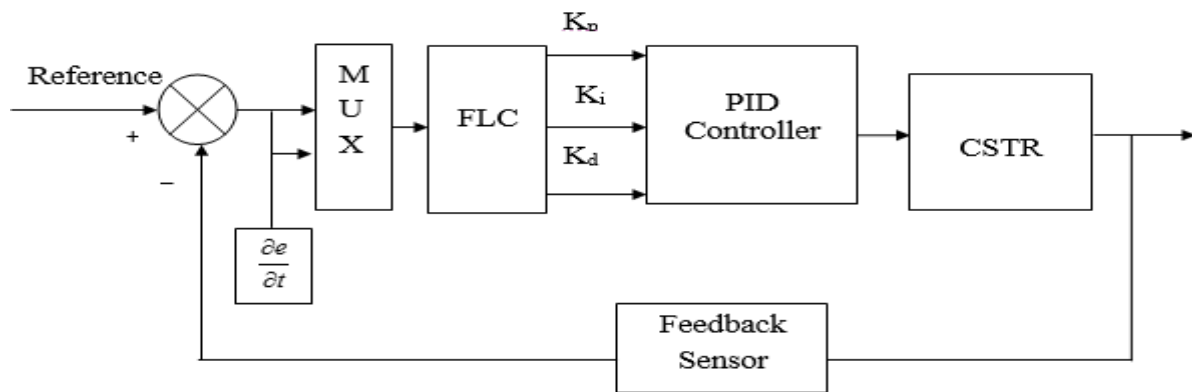


Figure 2.1: Block Diagram of fuzzy –PID Controller (Danius and Sivagamasundan, 2016).

2.1.3 Process Modelling

According to Nabi (2015), modelling, optimization and advanced control of chemical processes have more uses as a result of their abilities to analyzed the characteristics of real processes, impose some operating conditions which would not have been possible to perform on site as a result of safety of financial implications, investigate exiting processes elaborately and faster, compare different alternatives without altering the real process and make decision based on obtained results.

In order to design more efficient Continuous Stirred Tank Reactor (CSTR) process and to create improve products, proper understanding of chemical reactions in CSTR and behaviour of process is needed. This can be obtained from the basic chemistry and physics of chemical processes in the CSTR that can be

employed to calculate reaction rates and reactants parameters architecture. This is referred to as modelling from first principle.

Chemical reactions are either exothermic or endothermic processes that require the input or output of energy to keep the temperature of system constraint. Figure 2.2 shows a CSTR process flow schematics of operation. The process as presented in Prabhu and Bhaskaran (2014), assumes an irreversible exothermic reaction node as the working surroundings. The reactor's heat is separated by coolant medium that envelop the reactor in form of jackets. The fluid stream of reactant A is fed to reactor in the presence of catalyst arranged at the core of reactor (Prabhu and Bhaskaram, 2014). The overall process modelling is achieved using material or energy balance principle in reactor.

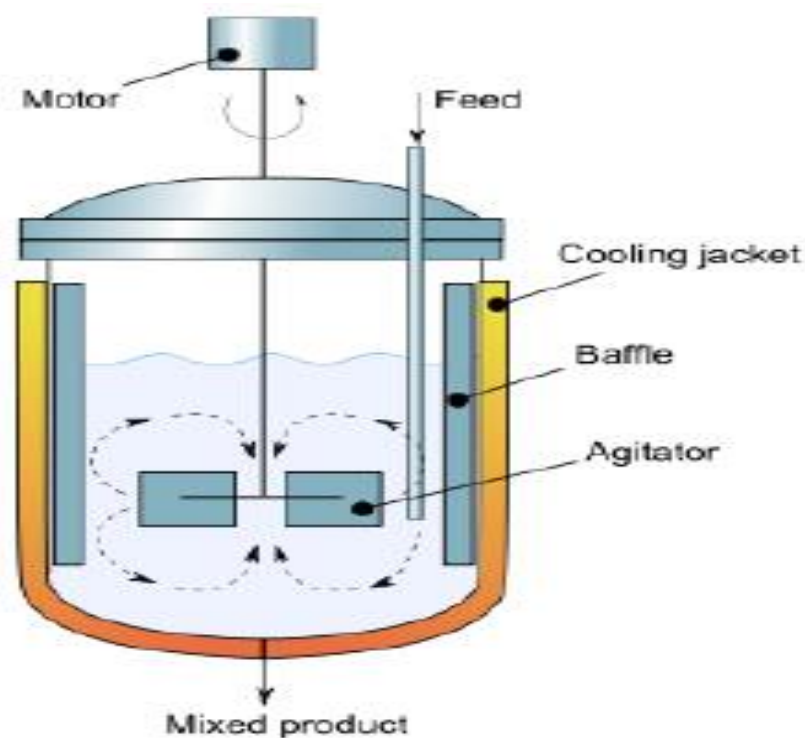


Figure 2.2: CSTR with Cooling Jacket. Prabhu and Bhaskaram (2014).

A reactor operates at a constant temperature, then that is called as the isothermal reactor. If any exothermic or endothermic reactions are involved in the reactor,

the temperature of the reactions mixture varies with time and we need to develop the energy balance equation for this non-isothermal reactor.

CSTR is one of the most important unit operations in chemical industries and it exhibits highly nonlinear behavior and usually has wide operating ranges. Chemical reactions in a reactor are either exothermic or endothermic and therefore require that energy either to be removed or added to the reactor to maintain constant temperature. Most of the reactors dealt in the literature for control purposes have been modeled as ideal CSTRs.

In an ideal CSTR, it is assumed that the reactor is well-mixed, meaning that the concentration at different positions of the CSTR is identical throughout the reactor. On the contrary, the mixing in the non-ideal CSTR is not uniform, resulting in bypass and stagnant regions. Due to bypass, the flow passing through the reactor will be less than the total volumetric flow rate and as a result, there will be slower decay of the transients in the concentration response in the reactor than in the case of perfect operation. In the dead space where the fluid does not enter, there results a lesser reactor volume than in the case of perfect operation. Hence, the fluid will pass through the reactor more quickly and as a result the transients in the concentration in the reactor will decay more rapidly than in the case for the perfect operation.

On the other hand, majority of chemical reactors are naturally nonlinear and Multi-Input Multi-Output (MIMO) systems. Compared with single-input single-output (SISO) counterparts, MIMO systems are more difficult to control because of the existence of interactions among input and output variables. Although considerable efforts have been dedicated to this problem and many design techniques have been proposed over the decades, control system design and implementation for MIMO processes are still a big challenge for practical control engineers.

2.1.4 Challenging Issues in the Design of the Hybrid Systems

As there are two or more entities which come from different domains of ontologies yet potentially fulfill a similar function in a hybrid system, these entities and their interactions can certainly play important roles in the performance of the whole hybrid system. The general goal of designing a hybrid system should be to make these different entities behave complementary to each other; as such an “emergent” behavior which can certainly not be achieved by each of them alone can be achieved.

The way to achieve this general goal can be:

- i. To understand two domains of ontologies involved in the interaction
- ii. To identify the strengths and weaknesses of each entity which may further change with respect to time, and
- iii. To assign function responsibilities for each of them so that they can behave complementary.

Despite its generality with the general procedure, specific design challenges with respect to three types of hybrid systems need to be identified. The challenging issue for the hybrid actuation system is the interaction between the Constant Velocity (CV) motor and the servomotor. For the hybrid motion system the challenging issue is how to minimize the interaction between the macro motion and the micro motion. The main challenging issue in the hybrid control system is how to reduce or eliminate the transient problem caused by the interaction of two or more different control laws.

These issues about the interaction of the hybrid systems are not well studied in the literature, or ad-hoc developments and methodologies are dispersed in the current literature without a coherent framework. In a specific domain of the hybrid actuation system, the use of a servomotor with a constant velocity profile

for the substitution of a constant velocity motor is still very popular in the literature. This is an erroneous practice of mechatronics design because the constant velocity (CV) motor may not perform “constant” velocity (there is usually a velocity fluctuation in the CV motor due to the variation in the operating condition and / or the variation in the workload on the system, or the disturbances) yet the servomotor with the constant velocity profile acts as a (perfect) constant velocity motion generator.

Process Control performs an important function in the safe manufacture of quality products at market demand, while protecting the environment. In order to meet the objective of production at market demand, physical parameters like flows rate, pressures and temperature within pipes and vessels, inventories of liquids and solids, and product quality must be controlled. The physical parameters that must be controlled are known as measured or controlled variables. There are various means available in literature and in practice for controlling these variables apply some control strategies.

As the control system is the core part of the mechanisms where the desired trajectory needed to achieve. There are many types of control algorithms being used in the research areas and industrial applications. In this section, some common control algorithms are reviewed as a fundamental of the hybrid control systems discussed in this thesis.

2.1.5 PID/PD Control

The Acronym PID is named after three control modes- Proportional, Integral and Derivative control algorithms/actions. The family of PID Controllers comprises: P-Proportional, I-Integral, PI- Proportional and Integral, PD- Proportional and Derivative, and PID controllers. The application of the D-mode of the PID Controller as a single unit is rare because of its sensitivity to noise and its difficulty in tuning; through it can make a control loop respond faster with less

overshoot (Smuts, 2002). Figure 2.3 shown a closed loop block diagram of a typical process control unit with a PID controller, and it will be used to explain the control techniques of the PID.

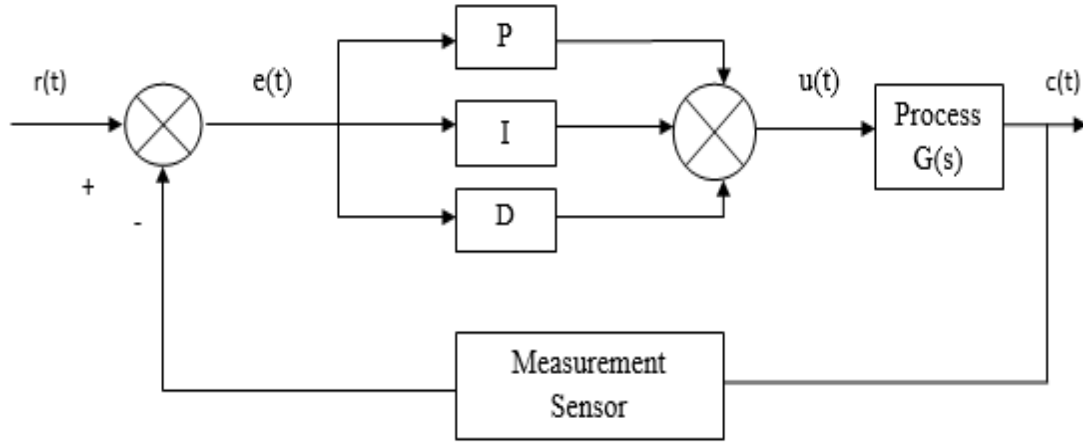


Figure 2.3: Closed Loop Control with PID Controller (Smuts, 2002).

The popular widely used feedback controller is the Proportional-Integral-Derivative (PID) controller that has the following form:

$$T(t) = K_P e(t) + K_I \int_0^1 e(\tau) d\tau + K_D e(t) \quad (2.1)$$

where $P K$, $I K$, and $D K$ are constant proportional, integral, and derivative gains, respectively, and $T(t)$ is the torque. Such a simple and effective controller has been extensively studied for different types of systems and widely used in many industrial processes, especially in robot control. If the control gain $I K$ is set to zero, the PID control turns to be the PD control.

It is well known that when position and velocity measurements are available, a PD control is the simplest control law and can be implemented using commercial control modules. In Rakesh, Borase, Maghade and Pawar (2020), their work provides a classical approaches using PID tuning and its applications in various areas. The techniques reviews the modern optimization rules used for the PID

tuning. The PD control with desired gravity compensation was shown to guarantee global and asymptotic stability for a point-set tracking problem.

To improve the trajectory tracking performance of robot manipulators, significant effort has been made to seek advanced control strategies. Achievements were made in developing adaptive control and robust control approaches that ensure the globally asymptotical convergence of the tracking errors. However, since these approaches are all based on centralized control structures that require rather tedious computation, their practical applications are not very promising. In fact, in spite of the theoretical results obtained, the majority of the industrial robots are still controlled by the decentralized (independent joint) PD law in favor of its simple computation and low-cost set-up. This is because the PD controllers, despite their simple structures, assure acceptable performances for a wide range of industrial plants. Furthermore, their usages are well known among industrial operators. Hence, the PD controllers provide a cost effective mean in industrial environments.

2.1.5.1 Proportional Control Mode

This control mode changes the output of the controller (Manipulated Variable, $u(t)$) in proportion to the error $e(t)$ in this case, the manipulated variable adjusts or changes through a proportional gain, K_p .

$$\text{Hence, } u(t) = K_p e(t) \quad (2.2)$$

2.1.5.2 Integral Control Mode

The integral control mode acts on the manipulated variable at a rate related to the deviation of the controlled variable $c(t)$, from its referenced or setpoint (or desired) value, $r(t)$. It produces a long term corrective change in manipulated variable, and thereby driving the error or offset to zero. The Integral control mode relates the changes in manipulated variable through an integral gain, K_i .

$$\text{Hence, } u(t) = K_i \int e(t) dt \quad (2.3)$$

2.1.5.3 Derivative Control Mode

This control mode acts on the manipulated variable in response to the rate of change of the controlled variable. The derivative variable relates the changes in manipulated through a derivative gain, K_d .

$$\text{Hence, } u(t) = K_d \frac{de(t)}{dt} \quad (2.4)$$

The error should be noted as $e(t) = r(t) - c(t)$

PI- Control Mode

This control action is achieved by combining the control modes of P. and I. This combination brings about stability with elimination of offset.

$$u(t) = K_p e(t) + K_i \int e(t) dt \quad (2.5)$$

PD- Control Mode

This control action adds the derivative control mode to a proportional controller thereby adding response for the rate of change of the controlled variable.

$$u(t) = K_p e(t) + K_d \frac{de(t)}{dt} \quad (2.6)$$

There are basically two types of PID controllers commonly used, which differ in the way the different modes are combined. These are series type and parallel type. In the series form combination of the modes, the actions of the modes are multiplied rather than added-up.

In this case, the various elements of the PID controller interact and as such is called interacting PID controller. The parallel form PID controller combines the

functions of the various modes by addition rather than multiplication. In this case the resulting PID control action is none interacting and thereby producing a mathematically PID expression.

$$u(t) = k_p e(t) + k_i \int e(t) dt + k_d \frac{de(t)}{dt} \quad (2.7)$$

In the above according to (Ahmmed, 2011) the PID Controller algorithm involves the three separate constant parameters which is the P, I, and D that combined, the proportional mode determines the reaction to the current error. The integral mode determines the reaction based on the sum of recent errors, and the derivatives mode determines the reaction based on the rate at which error has been changing.

2.1.6 Fuzzy Logic Controller

Conceptually Fuzzy Controllers are very simple. This control provides a formal methodology for representing, manipulating and implementing experience based knowledge gain by human about how to control a system.. The Fuzzy Logic Controller (FLC) uses human knowledge and expertise to compensate for the uncertainties in the process control (Kaur and Kaur, 2012). Fuzzy logic controller consists of an input stage, a process stage, and an output stage (Malhotra and Sodhi, 2011). Figure 2.4 shows a block diagram representation of FLC. It consists of four main parts:

- a. Fuzzification; this parts simply modifies and converts input into suitable linguistic values so that they can be compared to the rules in the knowledge base.
- b. Rule based or fuzzy knowledge base; this part holds knowledge in the form of set rules of how best to control system.

- c. Fuzzy reasoning or Inference mechanism; this evaluates which control rules are relevant at a given time and then decides what input to the plant should be.
- d. Defuzzification; this convert the conclusion arrived at by the fuzzy reasoning or inference mechanism into crisp ones.

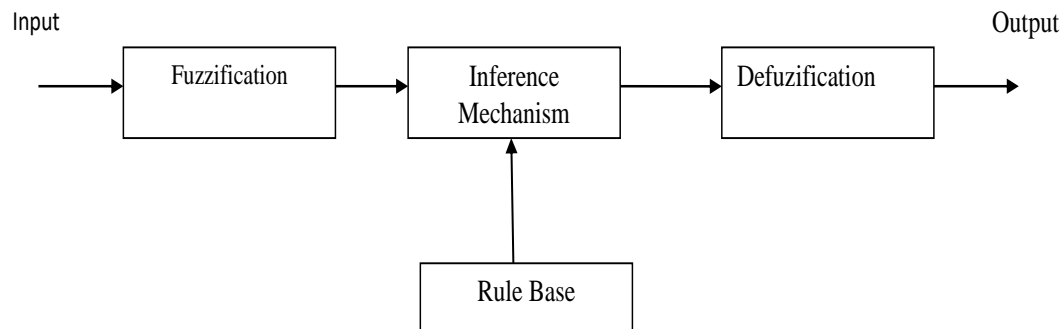


Figure 2.4: Fuzzy Logic Controller (FLC) Architecture (Malhotra and Sodhi, 2011).

The input stage maps sensor or other inputs, such as switches thumbwheels etc (Malhotra and Sodhi, 2011), to the right membership function (MF) and truth values. The processing stage maps each appropriate rule and provides a result for each and then combines the results of the rules. In the output stage, final conversion takes place such that the combined result of the processing stage is converted back into a specific control or manipulated value (Zhand and Wang, 2013).

There are different shapes of membership functions amongst which, the triangular membership shape is most common others are trapezoidal and bell curves which also used as shapes of membership functions. The membership shape is less important compare to the number of curves and their placement.

In the processing stage, a collection of logic rules in the form of IF-THEN statements are used. The IF Statement is called the “antecedent “and the THEN Statement is called the “Consequent “A typical FLC system have dozens of rules. Example a rule for a thermostat will read: IF (temperature is “Cold”) THEN (heater is “low”).

The conventional PID controllers has been well developed and are widely implemented for industrial automation and process control today because of its simplicity of operation, ease of design and high robustness (Chen et al, 2009). The performance of PID is greatly distorted and the efficiency is reduced due to non linearity in the process, plant (Malhotra and Sodhi, 2011). In the case of non linear systems, higher order and time-delayed linear systems and particularly complex and vague that have no precise dynamic models, fuzzy control, which does not dependent on dynamic models is a promising control method in current application.

The Fuzzy PID controllers are the extensions of linear structure of PID Controller (Malhotra and Sodhi, 2011). The structure of Fuzzy-PID controller is shown in figure 2.5 in which the parallel combination of PID controller and Fuzzy controller is exploited. The difference between the reference or setpoint or input) value $r(t)$ and the controlled variable (output value), $c(t)$ is the error $e(t)$, and both the value of $e(t)$ and its derivative $\dot{e}(t)$ applied to the fuzzy controller. The gain parameter K_p , K_i and K_d are calculated according to offline rules in fuzzy controller. The gain parameter of the PID can be continuously updated according to the error $e(t)$ and its derivative $\dot{e}(t)$

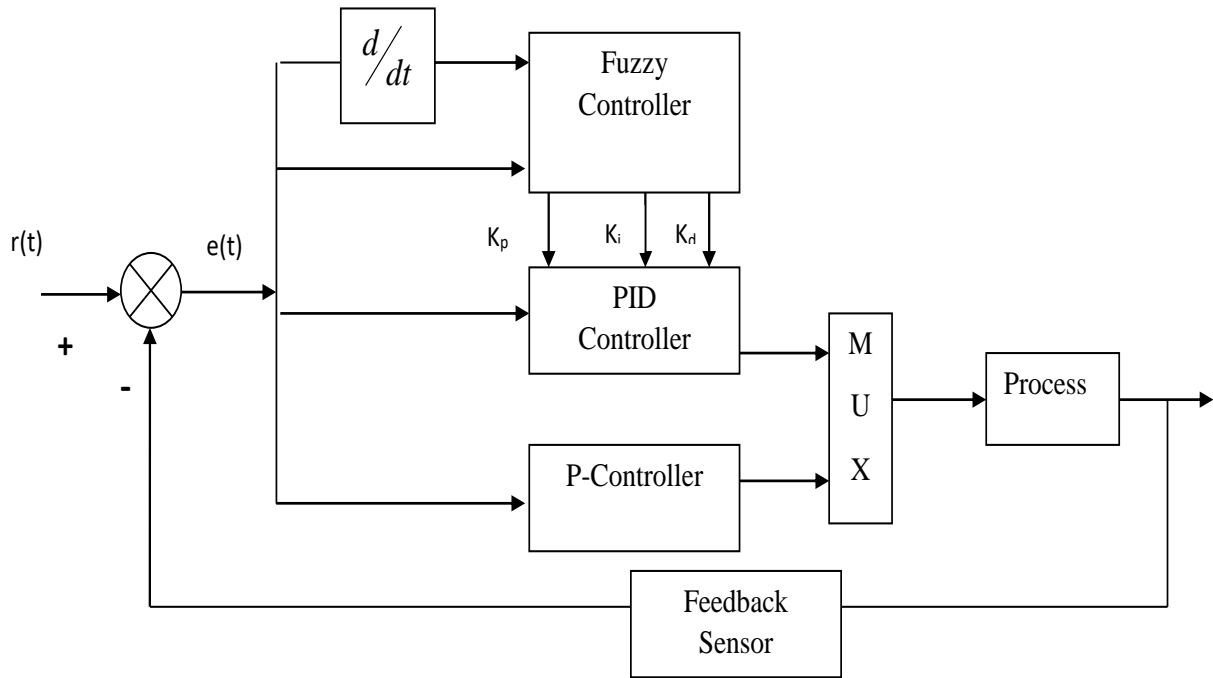


Figure 2.5: Fuzzy-PID Control Structure (Zhang and Wang, 2013).

2.1.6.1 Nonlinear PD Control

Trajectory tracking with PD/PID control may not lead to a high-accuracy tracking performance, especially at high speeds, because PD control is only a linear control. One of the refinements for a PD controller is the Nonlinear PD (NPD) controller.

Generally speaking, the NPD control is a control where the control gains are functions of the tracking errors. Designing a nonlinear PID controller is done by extending the linearization technique in which the three gains of the controller are functions of the error state. Seraji (1998) proposed a NPD control for the force control tasks, and also pointed out that the method is also applicable to position control also the work introduced a new class of simple NPD controllers and provided a formal treatment of their stability analysis based on the Popov criterion. It should be noted that many previous studies on NPD control were applied to the point-set control of linear systems. There are only few papers describing the NPD control for trajectory tracking of nonlinear systems such as

parallel manipulators. In this thesis, the NPD control will be extended to form a NPD learning control for the repetitive tasks of the mechanisms.

2.1.6.2 Artificial Neural Networks

Artificial Neural Network (ANN) is a computational Model which consists of artificial neurons operating as a unit for processing information (Chopra *et al.*, 2014). Figure 2.6 shows a closed loop control system with single neuron ANN structure (Liu *et al.*, 2010).

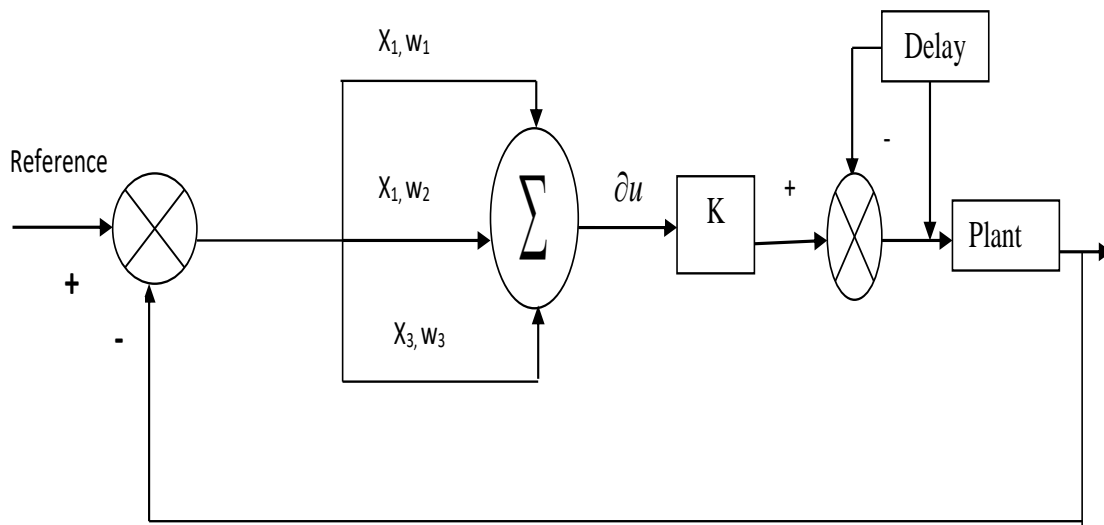


Figure 2.6: Closed- loop control System with Single Neuron (Chopra *et al.*, 2014).

In the closed- loop control system the neuron weight w_1 , w_2 , and w_3 will act as the gains of the proportional integral and derivative (PID) controller. With the aid of some intelligent algorithms the weights of the neural network are modified to attain the setpoint target (or desired object). The proportional error $x_1(k)$ integral error $x_2(k)$ and the derivative error $x_3(k)$ are given by (Chopra *et al.*, 2014)

$$x_1(k) = e(k) - e(k-1) \quad (2.8)$$

$$x_2(k) = e(k) \quad (2.9)$$

$$x_3(k) = e(k) - 2e(k-1) \quad (2.10)$$

These error signals are multiplied with their corresponding weights and act as the input to the single neuron. The output of the neuron is given by (Liu *et al.*, 2010, Chopra *et al.*, 2014)

$$u(k) = u(k-1) + K \sum_{i=1}^3 w_i(k) x_i(k) \quad (2.11)$$

2.1.7 Learning Control

Controllers are considered as a device which can sense information from linear or nonlinear system to improve the systems performance (D'Azzo, 2003). The aim of designing a control systems are stability, good disturbance rejection, and small tracking error (Farzin *et al.*, 2012) Several industrial systems are controlled by linear methodologies (e.g., Proportional-Derivative (PD) controller, Proportional- Integral (PI) controller or Proportional- Integral-Derivative (PID) controller), but when system works with various payloads and have uncertainty in dynamic models this technique has limitations. From the control point of view, uncertainty is divided into two main groups: uncertainty in unstructured inputs (e.g., noise, disturbance) and uncertainty in structure dynamics.

A learning control is such a control that has the ability to improve its performance in the future, based on experimental information it has gained in the past, through interactions with the plant and its environment. Learning control has the autonomous capability, has memory, and it is dynamic since it may vary over time. The learning control operates in the context of an objective function and it receives performance feedback that characterizes the appropriateness of its current behavior in that context. The advantages of learning control include:

1. It is not necessary to have a parametric model of the system;
2. The unstructured uncertainty can be compensated for to the extent that it is repeatable from one trial to the next. The difference between the adaptive control and the learning control. They pointed out that a control system that treats every

distinct operating situation as a novel one is limited to adaptive operation, whereas a system that correlates past experiences with past situations, and that can recall and exploit those past experience, is capable of learning. The purpose of using adaptive and learning control is to simplify the implementation process, to improve the system's reliability and thus achieve a good performance of the system.

2.1.7.1 Switching Control

In the control area, the switching control systems, also called as hybrid control systems in the control research area, mean the complex control systems which have discrete event dynamics as well as continuous time dynamics. The switching in system and control according to Liberzon (2003), this broad class of systems includes continuous systems with phased operation (bouncing ball, walking robots, biological cell growth and division), continuous systems controlled by discrete logic (aircraft autopilot modes, thermostats, chemical plants with valves and pumps automobile automatic transmissions), and coordinating processes (air and ground transportation systems, swarms of air vehicles). Figure 2.7, one can see that the switching control systems are characterized by the interaction of the continuous parts, governed by differential or difference equations, and the discrete parts, described by finite state machines, if-then-else rules, propositional and temporal logic. Such systems switch between many operating modes where each mode is governed by its own characteristic dynamical laws. Mode transitions are triggered by variables crossing specific thresholds (state events), by the elapse of certain time periods (time events), or by external inputs (input events). Typical switching control systems are embedded systems, constituted by dynamical components governed by logical/discrete decision components. Application areas of switching control systems include automotive,

manufacturing, communication networks, aerospace, robotics, traffic control, and chemical processes

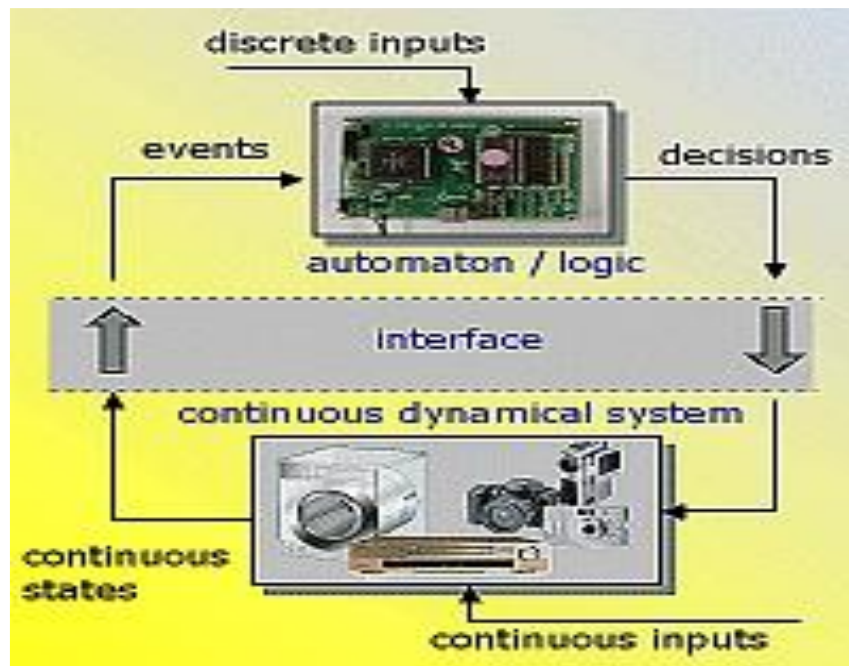


Figure 2.7: A switching control system and the interaction Liberzon (2003).

Excellent reviews of switching control systems were given by (Antsaklis *et al.* (2013) developed a framework for the modeling of certain classes of mixed continuous/discrete dynamic systems. Liberzon (2003) proposed a general framework that systematizes the notion of a switching control system, combining differential equations and automata, governed by a hybrid controller that issues continuous-variable commands and makes logic decisions.

Switching control systems produce switches between a finite, or at least countable, number of fixed controllers until stability is detected. Such controllers are very attractive from a theoretical viewpoint, in providing stabilization and asymptotic performance for a broad class of plants. However, such controllers are also known to have very poor transient properties, due to the long time required to search for a stabilizing feedback.

2.1.7.2 Fault Tolerant Control Methods

Fault tolerance may be called upon to improve system reliability, maintainability and survivability. The requirements for fault--tolerance are different in these three cases. Reliability deals with the ability to complete a task satisfactorily and with a period of time over which that ability is retained. A control system that allows normal completion of tasks after component fault improves reliability. Maintainability concerns the need for repair and the ease with which repairs can be made, with no premium placed on performance. Fault—tolerance could increase time between maintenance actions and allow the use of simpler repair procedures, Survivability relates to the likelihood of conducting an operation safely (without danger to the human operators of the controlled system), whether or not the task is completed. Degraded performance following a fault is permitted as long as the system is brought to an acceptable state.”

Fault-tolerant control (FTC) is not just a safety-critical issue; it is of interest to process plant operators who are keen to increase productivity. The reliability of a system component (including the controller) affects the plant operation. Fault-tolerance in control requires effort at every stage and in all aspects of system design.

FTC is classified as Active FTC and Passive FTC. In active fault-tolerance, a new control system is redesigned using desirable properties of performance and robustness that were important in the original system, but with the reduced capability of the impaired system in mind. In order to achieve feedback control reconfiguration/restructuring of feedback control, an active fault-tolerant system requires either a priori knowledge of expected fault types or a mechanism for detecting and isolating unanticipated faults. In the latter case, decisions concerning the location and nature of faults are then used to reschedule the controller function. Active approaches are divided into two main types of methods: projection-based methods and on-line automatic controller redesign

methods. The latter involves the calculation of new controller parameters in response to control impairment.

2.2 Theoretical Framework

The hybrid control model of Genetic Algorithm and PID was researched by (Eberhart, 1995) the work stated that a Genetic Algorithm (GA) is a stochastic optimization method based on the biological principles of Darwinian evolution GA has both global search and local search abilities. Genetic Algorithms: GA is a derivative free random optimization technique based on the ideas of natural selection and evolutionary processes. The fundamental components of GA are encoding, reproduction, crossover and mutation. GA encodes a number into a binary string called chromosomes. Depending upon the value of evaluation function also called the fitness function the parents are selected from a group of binary strings to perform the operations of crossover and mutation. Figure 2.8 shows the block diagram for tuning of PID parameters using GA.

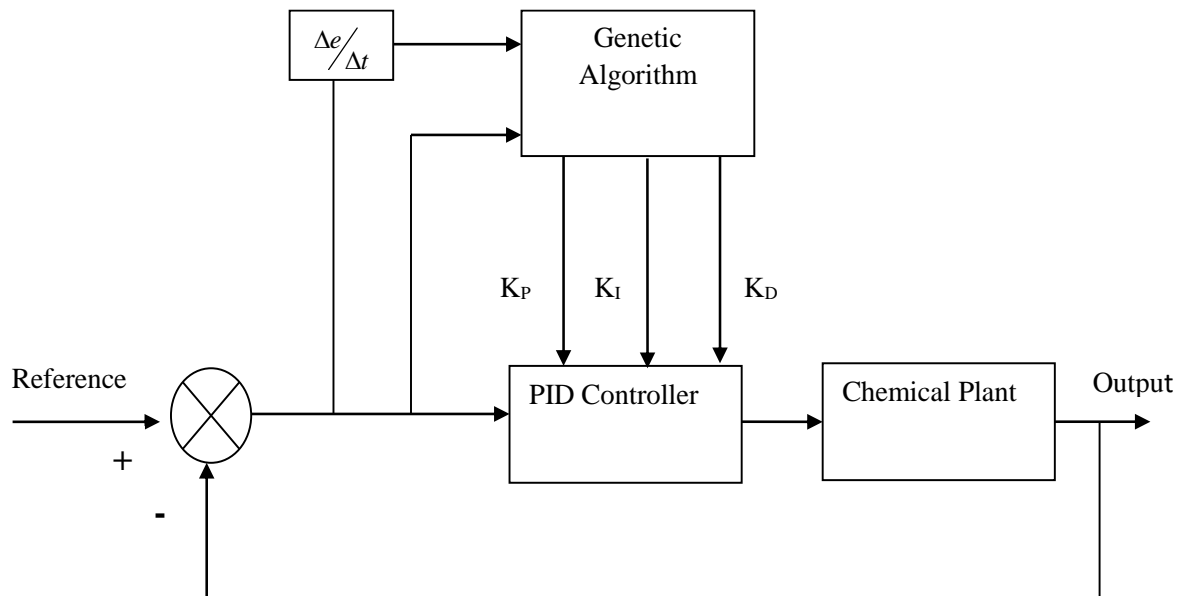


Figure 2.8: Block diagram for PID Intelligent controller tuning using GA (Keneddy and Eberhart, 1995).

Auto-tuning of the PID controller has been done using GA by minimizing the integral of square error (ISE). The ISE criterion is defined as.

$ISE = \int (r(t) - y(t))^2 dt$. The various steps in finding the parameters of a PID controller are:

- i. Step 1: Define the Plant transfer function.
- ii. Step 2: Initialize and calculate ISE. d i P K K K & ,
- iii. Step 3: Obtain pbest and gbest values.
- iv. Step 4: Calculate new population using mutation.
- v. Step 5: Obtain pbest1 and gbest1.
- vi. Step 6: Compare pbest and pbest1.
- vii. Step 7: Compare gbest and gbest1.
- viii. Step 8: Obtain the new values of K_p , K_i , & K_d and find out the step response for the closed loop system.

Gases are developed as a framework for a global search of the design space. It offers an alternative approach both for identification and control of nonlinear processes in process engineering.

Component needed for implementation for GA:

- i. Representation
- ii. Initialization
- iii. Evolution

Genetic Operators:

- ix. Selection
- x. Crossover
- xi. Mutation

2.2.1 Particle Swarm Optimization (PSO) Algorithm

Particle Swarm Optimization is a population based stochastic optimization technique first introduced by Kennedys and Ebert they were inspired by social

behaviour of bird flocking or fish schooling. According to Xin-She (2021) uses both the current global best and the individual best at iteration. One of the reasons of using the individual best is probably to increase the diversity in the quality solution. However Xin-She (2021) simulated using some randomness, there no compelling reason for using the individual best unless the optimization problem of interest is highly nonlinear and multimodal. In figure 2.9 shows the explanation of the iterative dimension of particle Swarm Optimization It is also based on swarm intelligence. The PSO has no systematically calculation method and it has no definite mathematic foundation. At present, the method can only be used successfully in the aspect of Evolutionary neural network, and its other applications are still being explored. PSO is widely used in engineering applications due to its high computational efficiency, easy implementation and stable convergence and there are few parameters to adjust and has been successfully applied in many areas such as function optimization, fuzzy gain scheduling, PID Auto tuning and fractional order PID controller design (Zhang and Wang, 2013).

Particle Swarm Optimization” became very popular due his continue optimization process allowing variations to multi targets and more. Consisting in the constant search of best solution, the method moves the particles (in this case represented as a (x,y) position) with a certain velocity calculated in every iteration. Each particle’s movement has the influence of his own the best known position and also the best known position in the space-search. The final result expected is that the particle swarm converge to the best solution. It’s important to mention that PSO doesn’t use Gradient Descent, so it can be used to non linear problems once it doesn’t require that the problem have to be differentiable.

The particle Swarm Algorithm as show below operates as follow:

At first, in the 2 for loops, it initializes the particles’ positions with a random uniform distribution within a permissible range for all its dimensions (Some

problems require handling to several dimensions). After that, for each particle, it calculates its fitness value and compared with his own best position (The p_best value is the best position of that specific particle has ever been) and then it chooses the best position of all particles in g_best.

The pseudo algorithm of Particle Swarm Optimization

For each particle i

 FOR each dimension d

 Initialize position x_{id} randomly within permission range

 Initialize velocity v_{id} randomly within permissible range

 End FOR

END FOR

Iteration k=1

DO

 FOR each particle i

 Calculate fitness value

 IF the fitness value is better than p_best_{id} in history

 Set current fitness value as the p_best_{id}

 END IF

END FOR

Choose the particle having the best fitness value as the g_best_d

FOR each particle i

 FOR each dimension d

 Calculate velocity according to the equation

$$V_{id}(k+1) = w \cdot x_{id}(k) + c_1 \text{rand}_1(p_{id} - x_{id}) + c_2 \text{rand}_2(p_{gd} - x_{id})$$

 Update particle position according to the equation

$$x_{id}(k+1) = x_{id}(k) + v_{id}(k+1)$$

 END FOR

END FOR

K = k+1

WHILE maximum in iterations or minimum error criteria are not attained

.

Let's take a closer look to the equation that defines the velocity of the next iteration of a particle dimension:

- $V_i(k+1)$ is the next iteration velocity

- W is an inertial parameter. This parameter affects the movement propagation given by the last velocity value.
- C_1 and C_2 are acceleration coefficients. C_1 value gives the importance of personal best value and C_2 is the importance of social best value.
- P_i is the best individual position and P_g is the best position of all particles. In the equation, is measured the distance of each of these parameters to the particle's actual position.
- $rand_1$ and $rand_2$ are random numbers where $0 \leq rand \leq 1$ and they control the influence of each value: Social and individual as shown in figure 2.9

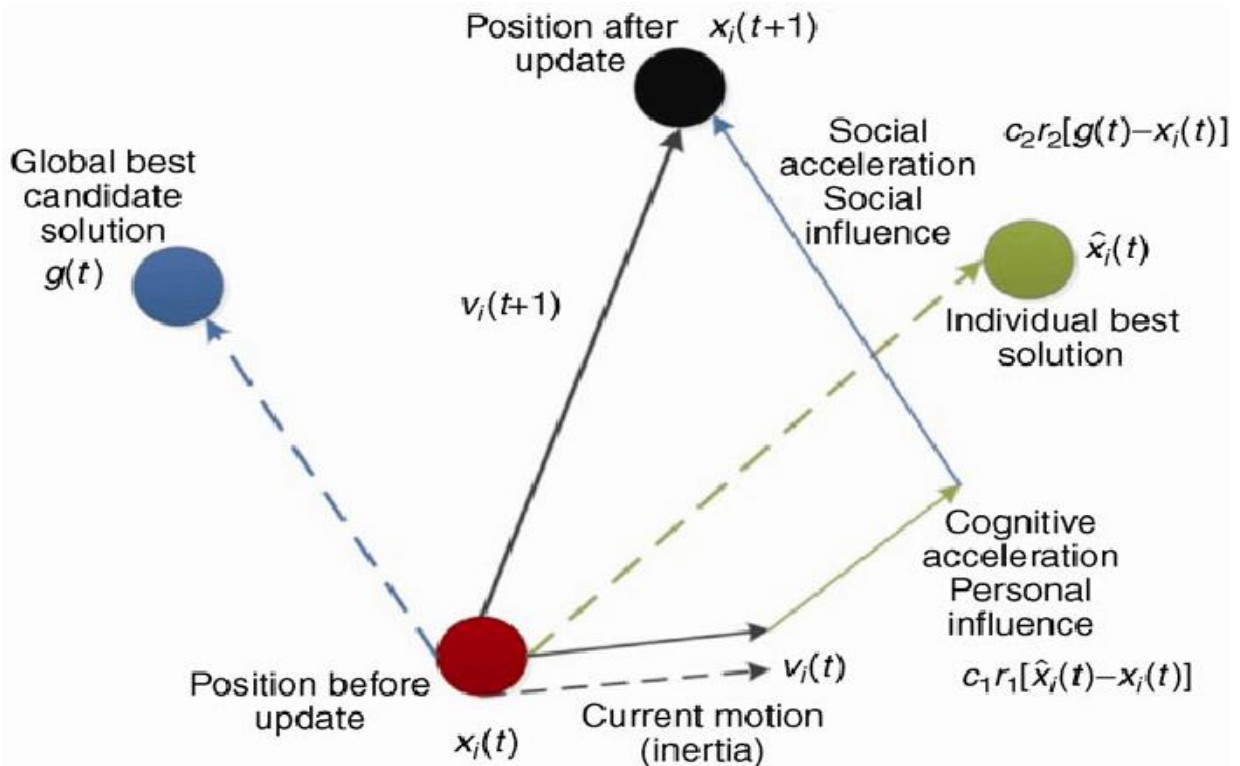


Figure 2.9: Accelerated Particle Dimension for PSO (Xin-She, 2021).

2.2.2 Architecture of Intelligent Machine Systems

An intelligent machine system can be represented by a general architecture as shown in Figure 2.10. In this architecture, the structure is defined as a set of entities connected in a meaningful way. Entities are represented by a set of

properties, and these properties are called states. The name of the state is called the state variable. The behavior is the causal relationship among a set of related state variables. The function is defined as the purpose in the mind of humans and can be realized by the system (structure) due to the provision of certain behaviors generated by the structure. The controlled behavior is defined as the behavior resulting from the adjustments on the structure and the state; the adjustments are needed for the reduction of any discrepancy between an actual function and a desired function. The principle governs the behavior in the sense that the causal relationship is derived from the principle. The principle can be viewed as the constraint that determines a particular behavior.

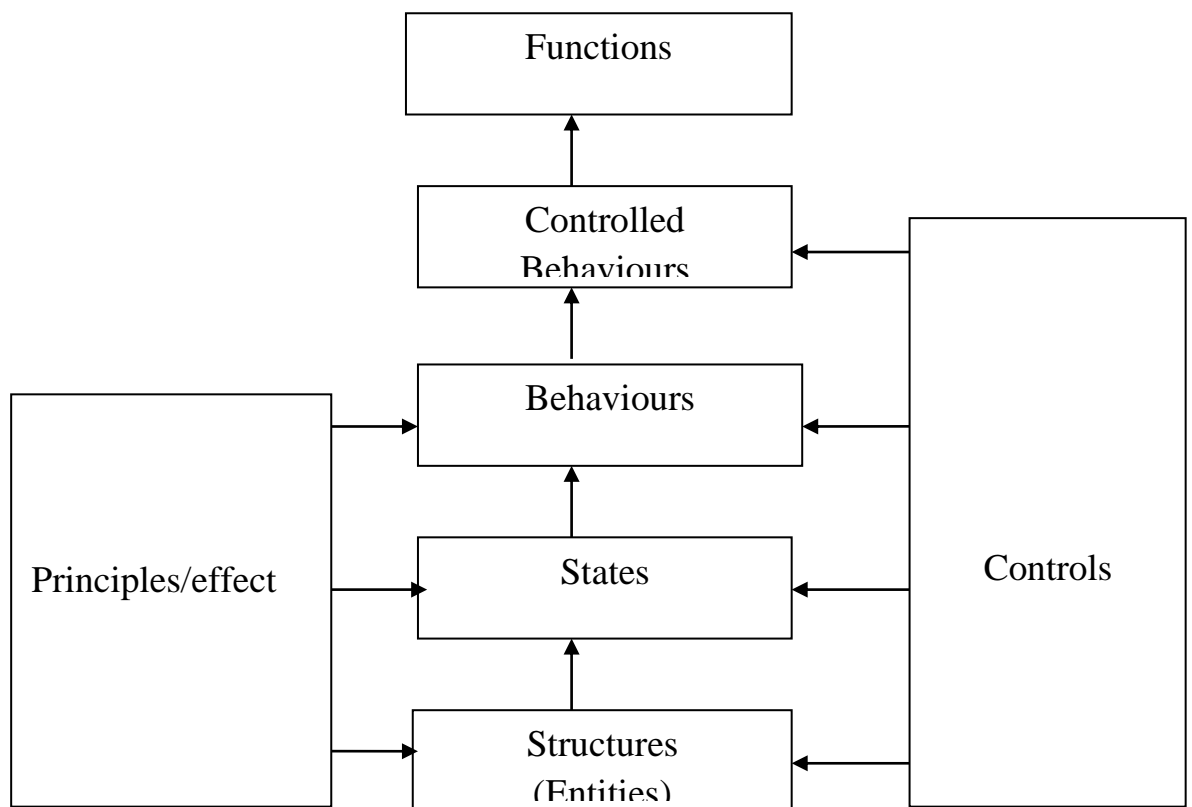


Figure 2.10: Architecture of intelligent machine systems (Puren, 2005).

In (Puren, 2005) architecture may be called the function-control-behavior-structure (FCBS) which is a specialized architecture of a more generalized architecture called the function-behavior-structure (FBS). The behavior concept

in the FBS architecture is further specialized into the sub-behavior, the controlled behavior in this case, in the FCBS architecture.

An intelligent machine system can also be viewed as consisting of (i) sensors, (ii) end effectors, (iii) mechanisms, (iv) actuators, and (v) controllers function view.

Actuators may be classified based on how they work or based on principles (or effects) which govern their behaviors as showed in figure 2.10 for a general relationship among the structure, state, behavior, and principle). Mechanisms can also be classified in the same way, so are the controllers.

2.2.2.1 Hybrid Intelligent Machine System

The word "hybrid" is often used as a synonym of the word “heterogeneous” in the nature or synthetic realms. In biology, the improvement of capabilities of a plant through some hybridization has been known for years. In material sciences, the notion of composite materials is another example of improvement of material properties. Still another example is the hybrid control system for controlling cell cycles (Antsaklis and Koutsoutous, 2003).

The common feature of these hybrid systems is the presence of two entities with each having different characteristics. The combining of these two entities can improve the behavior of a system with their presence. Further, the improvement of the behavior can make sense in enhancing or extending or both. In the case of the extending, a unique advantage of the hybrid system is that two contradicting behaviors of a system may exist harmonically; for example the ductility and the strength in material engineering.

A system is called a *hybrid intelligent machine system* if the system is constructed in such a way that within its scope (components, subsystems) two or more elements are combined to contribute to the system behavior by playing at least one of the roles: sensor, end-effector, mechanism, actuator, or controller.

2.2.2.2 Intelligent Supervision Based on Multi-Agent Systems

Intelligent supervisory system that combines the principles of fuzzy logic, the Internal Model Control (IMC) architecture and the paradigm of Multi-Agent Systems (MAS) was proposed (Jorge *et al.*, 2014). The methodology has been conceived to be applied in an intelligent supervisory system, specifically for two kinds of complex industrial processes: the gas-oil separation process and the oil heating process. According to (Jorge *et al.*, 2014) supervision of industrial processes is an area of intense study and many approaches have been used based on artificial intelligence techniques; where most of the resultant methods are based on hybrid systems. Supervisory control is typically based on a global assessment of the current operation point (Lunze, 2000), that leads the parameters of a system to be characterized with the intention of providing diagnosis and to anticipate possible changes that affect its current state of operation through the adjustment of scalar parameters of the controller or changes in the setpoint.

Among the broad variety of intelligent supervisory system schemes proposed in many publications, (Ortega *et al.*, 2007), there exist three basic elements: the event detector, the model of the process and the decision system. The event detector may be implemented in many different ways; in particular, some common approaches are based on fuzzy logic and neural network; which are used to detect possible operation states in a particular process. According to (Diarmuid and Ramirez, 2016), there are many alternatives for the determination of the model of the process such as qualitative models, broadly recommended in supervisory scheme and non linear models based on local models such as the Takagi-Sugeno fuzzy model.

Finally, the decision system guided by a set of objectives and priorities of the process makes some decisions that result in changes of the reference point of the process or changes in the scalar parameters of the controller. It is important to

highlight that in a general perspective, the supervision process implies the following connotations:

- i. Detection of the plant's parameters, deviations and instability
- ii. Guidance of controller tuning, adaptation and synthesis
- iii. identification and diagnosis of process faults

2.2.2.3 Supervisory Control System

In order to design an intelligent supervisory control system as proposed by (Jorge *et al*, 2013) under the conception of multi-agent system, it is necessary to define the following aspects:

- i. The definition of a distributed platform, normally determined by the system to be supervised.
- ii. The type of intelligence and knowledge presentation strategy that will be used.
- iii. The development of a communication, coordination and cooperation protocol between the different intelligent modules.

In Figure 2.11 shows the generic scheme of an agent, also, multi-agents systems technology addresses some of the before mentioned difficulties. In particular, (Liberzon, 2003) stated the process control area is a natural one for the application of intelligent agents because controllers may be regarded by themselves as autonomous reactive systems.

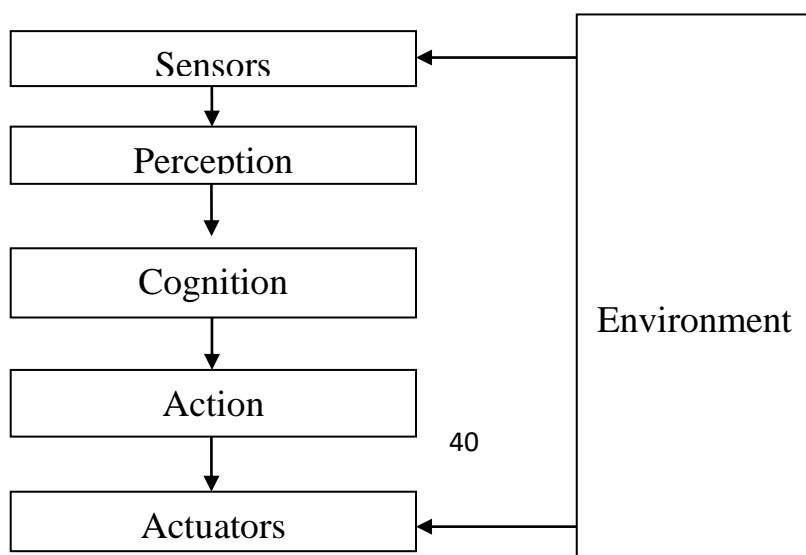


Figure 2.11: Generic scheme for Actuator and Sensor (Liberzon, 2003)

A significant number of definitions of agents can be found in the literature and one of the simplest and most accepted is the one presented by (Liberzon, 2003) that considers an agent as an entity that acts and perceives in a particular environment. Nevertheless, a formal definition of agent was established in where an agent is considered as a computational system situated in an environment and is capable of autonomous and flexible actions in that environment, in order to meet its design objectives.

Communication, cooperation and coordination are three key issues in multi-agent systems. Coordination is the process by which an agent reasons about its local actions and the anticipated actions of others to try to ensure the community acts in a coherent manner. To coordinate their goals and tasks, agents have to explicitly take dependencies among their activities into consideration. Two basic, contrasting patterns of coordination are cooperation and competition. In the case of cooperation, several agents work together and draw on the broad collection of their knowledge and capabilities to achieve a common goal. Against that, in the case of competition, several agents work against each other because their goals are conflicting.” We add that agents also perform a third type of coordinating behavior which is neither cooperation, nor competition. It is collaboration: agents assist others even though they do not have common goals. This is possible by means of negotiation. Coordination by means of cooperation or negotiation seems to require some form of communication. In agent based systems, communication can occur at different levels of sophistication ranging from primitive to high-level.

Software agents can be developed on different platforms and using different programming languages. They usually interact by exchanging complex symbolic information and possibly have to conform to complex interaction protocols. Several proposals have been designed to develop agent communication languages such as the KQML (Knowledge Query Manipulation Language).

On the other hand, it is imperative, that agents use a common vocabulary to communicate where the symbols used in messages have the same, predefined meaning for all of the agents. This amount to concede the importance that agents share ontology. A computational ontology is an informational model that describes concepts and relations in some specific domain.

The capacity of communication between agents can be interpreted as an additional element of perception (receiving messages), of cognition (interpretation and actions to be executed) and actions (sending of messages).

This characteristic allows agents to conform societies where global objectives may exist apart from the individual objectives of each agent. In this context, cooperation becomes relevant for those societies. A cooperative situation is validated if either adding a new agent could result in an increase in the performance level of a determined group or agent actions serve to avoid actual conflicts.

2.3 Empirical Literature

In this section the review of empirical studies of related works are presented based on the control technique or method used to achieved desired performance result. Conventional and particle Swam optimization based PID Control technique. In work carried out by Thulasidharan *et al.* (2017) on tuning of PID Control using optimization techniques for a multi-input-multi-output (MIMO)

process. It considered two processes which consist of Quadruple Tank process and CSTR process. Dynamic model of the two processes was performed by linearization of the system due to MIMO process. In order to tune the controller parameters, two optimization techniques consisting of PSO, and GA were use. Performance comparison was for the two different optimization techniques used for tuning of PID controller gain parameters for the two process considered. It stated that the simulation results showed that PSO based turned provided better response than that of GA whereas, for the Quadruple Tank process, both optimization techniques provided almost the same response with slight difference in their peak overshoot values.

Also, (Iannic *et al.*, 2021) in their research proposed an susceptible Infection-Recovered and Proportional Integral Derivative (SIR-PID) model the behaviour of the COVID-19 epidemic in the presence of social restrictions with aim of changing the basic reproduction numbers. The work also apply a simple time dependent modification of the SIR Modeling of the COVID-19 outbreak revolution to built a list possibility of using a PID controller to achieve the desired containment threshold smoothly with the aim of avoiding damages in terms of economic crisis and human cost.

According to Karishna *et al.* (2012) presented a turning of proportional integral and derivative (PID) controllers for unstable continuous stirred tank reactors (CSTR). It proposed two PIDs controller design techniques for unstable Second Order plus Time Delay System with a Zero (SOPTDZ) based on internal model control (IMC) and Stability Analysis (SA) principle. The controllers were used to control unstable CSTRs whose reaction of order irreversible reaction performance comparison was carried out with the proposed method and the synthesis method. The simulation, results obtained showed that the controller designed by proposed showed more robust performance than the one designed using synthesis method on nonlinear unstable CSTRs.

Ojiabo and Igbokwe (2015) presented Proportional Integral Derivative (PID) of a continuous stirred tank reactor (CSTR). It used non isothermal CSTR and different control modes. Simulation parameters were chosen at equilibrium state and dynamic point. It maintained that simulation results obtained indicated that stability of the non isothermal CSTR at different turning point and disturbances. Artur (2016) presented modelling and control of CSTR with PID Controller. A model of dynamics control for CSTR in methanol synthesis in a three-phase system was developed. Simulation of the reactor was performed for steady and transient states. Efficiency ratio for achieving maximum performance of the output for a unit reactor volume was calculated. Simulation in closed loop was conducted which allow the control process to receive data for optimum production capacity, with the elimination of local hot spot or temperature runaway.

Maheshwan *et al.* (2016) studied design and analysis of PID controller for CSTR process. The objective of the study was to control temperature and load disturbance rejection of CSTR. Simulation result showed that the PID controller provided a less percent overshoot with efficient load disturbance rejection with a minimum setting time. In the research of Poovarassan *et al.*, (2014) studied design of fractional order PID controller for a CSTR process. It proposed the application of fractional order PID (FOPID) controller in CSTR process controls. It employed soft computing techniques which comprises genetic algorithm (GA) and particle swarm optimization (PSO) to model the CSTR and for obtaining model parameters. It maintained that the developed model was able to compensate for the nonlinearity present in the CSTR. Simulation results were presented in terms PID and FOPID. The performance was analyzed with respect to Integral Square Error (ISE). It was observed that FOPID provided better performance than PID.

Aboelela., Hennas, and Dorrah, (2015) presented design and implementation of a PID controller for a CSTR system using particle swarm algorithms. It applied proportional integral (PI) and PID controller tuned with PSO, Adaptive Weighted PSO (AWPSO) algorithms to CSTR process to take care of the temperature and concentration control. Three error criteria were used to achieve the optimization process. These include integral of square error (ISE), the Integral of Absolute Error (IAE) and integral of Time Absolute Error (ITAE). In order to test the robustness of temperature and concentration performance of the process, in (Aboelela *et al.*, 2015) some of the parameters of CSTR were altered. It maintained that better performance algorithm was observed.

Khanduja (2015) presented CSTR control by using model reference adaptive control and PSO. A comparative analysis of CSTR control based on model reference adaptive control (MRAC) and optimal turning of PID control based on PSO. The Massachusetts Institute of Technology (MIT) rule and Lyapunov's rule were used to develop the adaptation law for the MRAC techniques. The PID Controller parameters were turned based on PSO to obtain optimized operating point for minimum ISE condition. The simulation results show that the PID controller tuning based on PSO provided better control response of the MRAC.

Pathak, Markana and Parekh (2010) studied optimal control of CSTR. It developed a second order dynamic model for CSTR. A nonlinear Model Predictive Control (MPC) of CSTR was performed. An optimal state estimation was carried out using Extended Kalman Filter (EKF). It stated that with the addition of MPC strategy to CSTR control loop, an optimal temperature control inside the reactor was achieved compared to conventional control methods. It then concluded that the MPC of CSTR indicated satisfactory setpoint tracking performance.

In the work carryout by Bayer *et al.* (2011) studied the problem of multiple hard output constraints imposed on a continuous stirred tank reactor (CSTR) subject to external disturbances. Constraints on the concentration and on the temperature are considered. The authors showed, analytically and with simulations, that there were critical combinations of constraints, where robust constraint satisfaction cannot be guaranteed. As a consequence violation of at least one constraint has to be allowed.

Darius and Sivagamasundan (2016) presented design and implementation of controllers' algorithm for CSTR process the controller provided the most efficient one for CSTR process temperature control was chosen. Various process variables like temperature, concentration were controlled in this Process using different control techniques. The structures of the prosed scheme for decentralized and decoupled control are shown in Figures 2.12 and 2.13. It chose an MPC because of its efficient algorithm for multivariable control. It carried out quantitative and quantitative comparison of the closed loop response stimulations for the MPC control techniques base on IAE and ISE criteria. The results obtained showed that the MPC control techniques provided better setpoint tracking and load distance rejection than other control techniques.

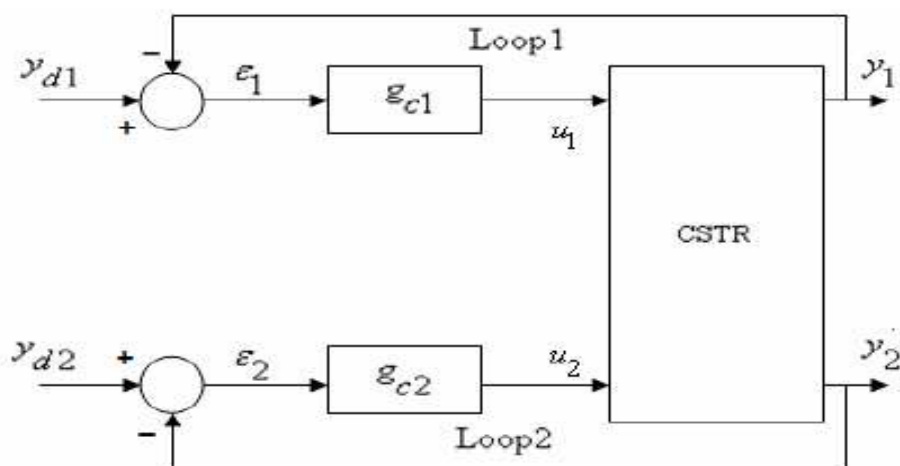


Figure 2.12: Block diagram representation of decentralized control scheme for the ideal CSTR (Darius and Sivagamasundan, 2016)

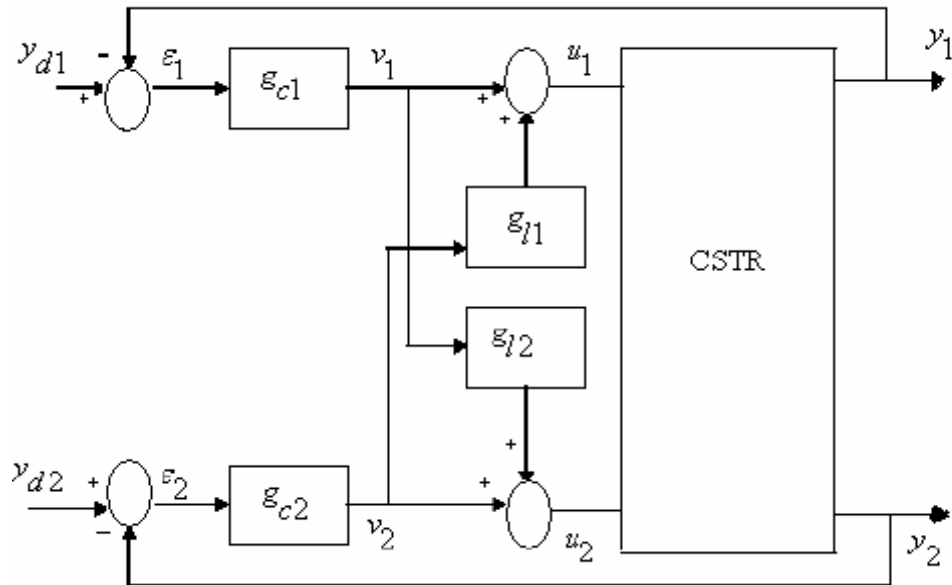


Figure 2.13: Block diagram representation of decoupled control scheme for the ideal CSTR (Darius and Sivagamasundan, 2016)

Gao, O'Dwyer, and Coyle (2003) studied model predictive control of CSTR based on local model networks. It presented nonlinear predictive controller effectively integrated predictive controller with a local model network which uses a neural network like gating system rather than using the quadratic optimization calculation which it stated was time consuming. Simulation was performed on a CSTR and the result obtained revealed satisfactory in terms of setpoint tracking and regulation performance over the entire operating range.

Vijayakumar and Manigaudan (2015) presented design of soft computing based nonlinear model predictive controller for the CSTR process. It proposed a soft computing least square vector machine (LSSVM) based nonlinear Model Predictive Controller (MPC) for CSTR process. It stated that the use of linear MPC was not sufficient to describe performance nonlinear dynamic systems. Order to adequately describe the performance of nonlinear system such as CSTR, process; nonlinear MPC which is characterized by non linear models was necessary. The proposed nonlinear MPC based on LSSVM was process.

The performance of the system was measured through Integral of Square Error (ISE) and Integral Absolute Error (IAE). The effectiveness of the proposed technique was proven through the closed loop performance and predictive performance simulations of the CSTR process. The results obtained were compared with those of neural MPC (its Counter-part). The comparison reveals that the proposed scheme provided better prediction with high accuracy and improved stability.

Seki *et al.*, (2004) studied nonlinear Model Predictive Control using successive linearization with application to chemical reactors. It developed a nonlinear process model along its trajectory and an infinite-time horizon linear optimal regulator with integrator. This was implemented in a receding horizon pattern. In order to demonstrate the effectiveness of the proposed controller simulations were conducted for highly non linear chemical reactors which showed static input or multiplicities of output. It maintained that the proposed algorithm has been successfully to a grade transition operation in an industrial reactor.

Mansouri and Rezaie (2012) studied nonlinear model predictive control based on neural network model for controlling CSTR system. It proposed a neural network based model predictive control technique for nonlinear CSTR system control. The proposed technique is such that a neural network model of the system is used to predict future characteristics of the process. It used nonlinear optimization tool to calculate the control signal. It maintained that the calculation of the control signal was based on the neural network model of the system so as to optimize the performance of the system over a given time horizon. The efficiency evaluation of the proposed techniques was performed by comparing the proposed technique with a nonlinear fuzzy controller and also with PID control. Simulations were performed for the CSTR system to study the performance of the controllers in normal conditions and in the presence of disturbance. It stated that the simulation

results obtained from MATLAB/SIMULINK software revealed the superiority of the proposed technique over other techniques both cases.

Poongodi and Sudhanan (2015) studied simulation of temperature control methodologies for chemical reactor. It compared the performance of PID controller, fuzzy-PID controller, MPC controller and MPC based PID controller. A state model of CSTR process was derived from the dynamic equation. Simulations were done considering the various controllers in the closed loop of CSTR process. Performance comparison reveal that the MPC based PID controllers provided better response performance with no overshoot, least rise time, delay time and setting time compared to other three techniques.

Debnath and Tripsthi (2016) carried out a case study to optimize design in linear CSTR using multiple model predictive control approach. It developed a control strategy for a nonlinear CSTR process, its multiple model predictive control (MMPC) technique. The MMPC was implemented using MATLAB software as shown in Figure 2.14. Performance comparison was out for the CSTR process with single MPC controller and multiple MPC controllers. The developed scheme provided better disturbance rejection than single MPC technique.

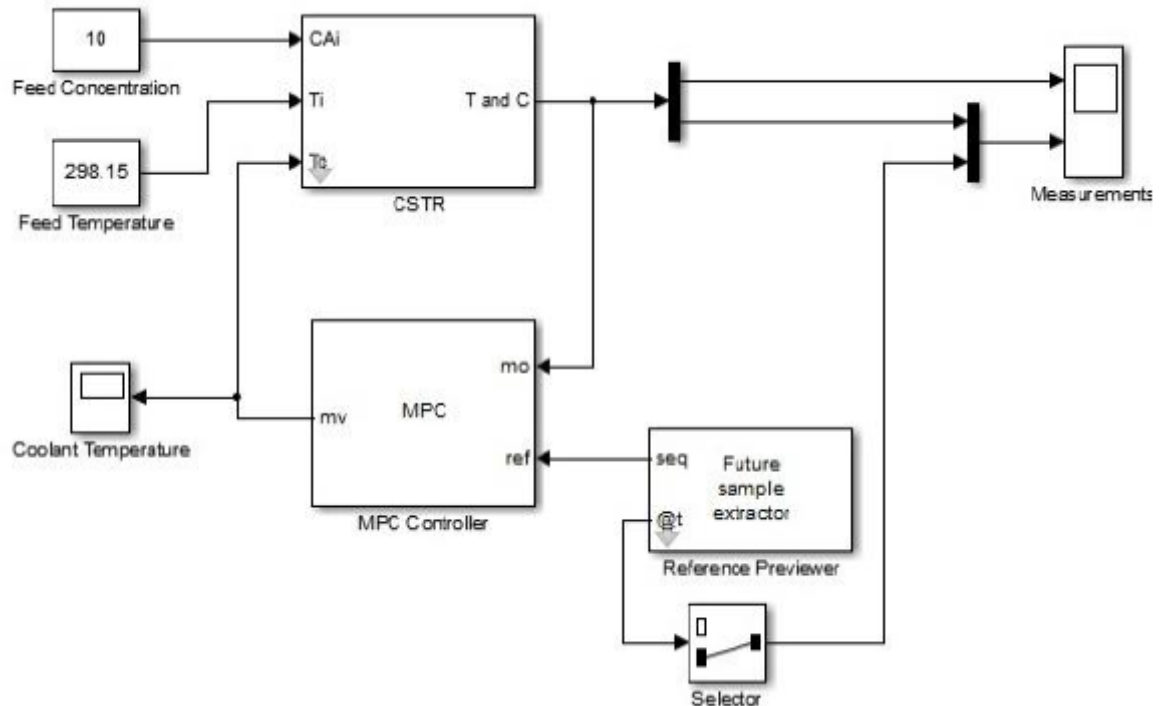


Figure 2.14: Modelling of CSTR plant with one MPC Controller (Debnath and Tripathi 2016)

Allwin *et al.* (2014) carried out comparison of conventional controller with model predictive controller for CSTR process. It compared the performance of PID controller with MPC controller in the closed loop temperature control of a CSTR process. It stated that the result obtained from the simulation conducted showed that the MPC controller provided better temperature control with zero percentage overshoot and minimum setting time compared to conventional PID controller.

In the work done by Deepa *et al.* (2015) presented design of controllers for CSTR. It designed different controllers for temperatures control in CSTR systems. A Proportional Integral (PI) Controller algorithm was initially designed by using Zeigler-Nichols modified Zeigler-Nichols Tyrenus –luyben, Shen-Yu, internal Model control (IMC) based tuning method. A Fuzzy Logic Controller (FLC) was then designed. Simulations were performed and responses were obtained for the designed controllers. Performance comparison was carried out

for the controllers in terms of maximum peak overshoot; setting time, rise time, Integral Square of Error (ISE) and Integral Absolute Error (IAE) as performance index it maintained that the FLC was observed to be a more promising controller than the conventional PI controller.

Jhon, Victor and Jorge (2018), presented a tuning proposal for direct fuzzy PID controllers oriented to industrial Continuous processes. It presented a simple but effective systematic method for many direct fuzzy PID tuning controllers based on static gain calculation of linear sub- models and controller scaling factors. The effective of the proposed method was demonstrated on two specific case of processes considered, which included CSTR process. It maintained that the proposed approach provided a considerable reduction in the tuning time and improving the performance of processes whose operating point's changes.

In the work by Kumar *et al.*, (2011) studied design, performance and stability analysis of a formula-based fuzzy PI controller. It proposed a formula-based fuzzy proportional integral (FPI) controller. The proposed controller algorithm was however, discrete-time fuzzy version of the conventional PI controller. The closed loop control stability analysis were performed using small gain theorem by allowing sufficient condition for the bounded-input bounded- bounded output (BIBO). The FPI controller performance was evaluated for setpoint tracking, disturbance rejection, and noise suppression. Computer simulations were performed in LabVIEW considering a nonlinear non-linear catalytic CSTR simulation results obtained from the proposed FPI were compared with that of nonlinear FPI (NFPI) controller. It maintained that the proposed system provided better result than NLFPI.

Applied ANN, Fuzzy logic, Expert Systems and Genetic Algorithms to build estimators in several chemical process system, research carried out by (Jarinah *et al.*, 2015) also review and provide a comprehensive list of AI based hybrid

algorithms that have been applied as estimator in chemical process systems by highlighting in Table 2.1 the comparison of several hybrid system structure.

Recently, researchers have developed algorithms that combined two or more AI methods in what is called hybrid systems. These algorithms are applied as estimators to overcome the limitations of the single algorithm and to further increase the estimator's performances.

For example, ANN will only allow reasoning from input to outputs and this can be overcome by using the adaptive neurofuzzy inference systems (ANFIS). Normally, the combination utilized the advantages of each of the algorithms such as the hybrid neural network (HNN), ANFIS, fuzzy neural network (FNN) and expert system neural network (ES-NN) (Sivan *et al.*, 2007)

2.3.1 pH System Architecture

There are two primary system designs for pH adjustment systems: continuous and batch.

Continuous flow through systems are typically used when

- Influent Flow is relatively constant and not characterized by large surges in flow.
- Influent chemistry is relatively constant and not characterized by large swings in incoming pH or chemistry
- Effluent pH limit is not overly tight (e.g. range of < 2.0 pH units)
- As a single stage in a more complex application such as heavy metal reduction or fluoride removal

Example of wastewater effluent streams that are candidates for treatment with Continuous pH Adjustment:

- Wastes from Pharmaceutical Manufacturing
- Life Science Laboratories
- University and Research Laboratories
- Semi-conductor and Nano-Technology Acid Waste Neutralization
- Heavy Metal Reduction and Fluoride Removal applications

2.3.1.1 Continuous Flow: pH Adjustment Systems

The system depicted below in Figure 2.15 in the system there are treatment tanks, mixer, acid and caustic metering pumps and a pH probe with a controller. The influent flow enters the tank at the bottom and exits the tank, through a distant port at the top. The objective with port placement is to create the longest possible path between the inlet and discharge ports.

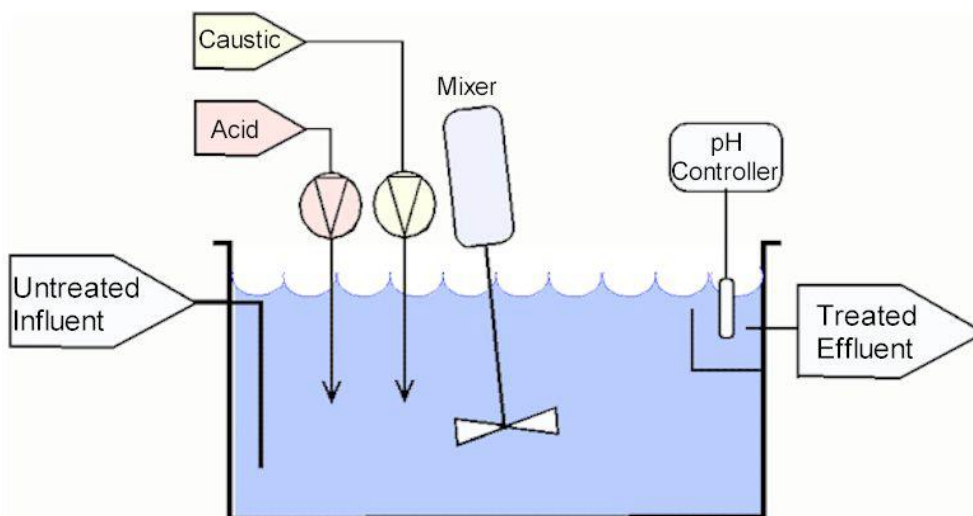


Figure 2.15: Continuous pH System Architecture

Therefore one gallon entering the influent port displaces one gallon through the discharge port. As the influent flow enters the treatment tank it is thoroughly mixed with the tank contents. If the pH of the influent varies from the tank contents (an obvious likelihood) then the influent flow will be pH adjusted through the resultant chemical reaction that occurs as the influent mixes with the

contents. There will obviously be an equal and opposite reaction within the tank contents. This opposite reaction is sensed by the pH probe which provides a continuous pH signal to the pH controller. The controller then paces the appropriate metering pump to bring the body of water within the tank back into range. If the influent flow was alkaline, for example, the result would be a steady rise in the pH of the tank contents as measured by the pH probe at the discharge port. The pH controller would then pace the acid metering pump at an appropriate rate to bring the pH back down into range.

One of the biggest advantages to a system of this basic design is that it is simple and capable of handling relatively high flows. However, since the tank is always full there is no guarantee, regardless of tank size or control system proficiency, that the effluent will always be in range. After all, the pH control uses a feedback loop, which by design does nothing until an error is sensed. If the influent flow and chemistry are high enough or strong enough then the effluent pH can easily drift out of the prescribed bounds. Therefore pH contraventions are a distinct possibility with this system since there is no mechanism to stop a discharge.

2.3.1.2 Batch Flow: pH Adjustment System

A batch system, however, treats a batch of a fixed volume and then discharges the batch only when it meets the discharge criteria in figure 2.16 shows the Batch pH Adjustment Systems.

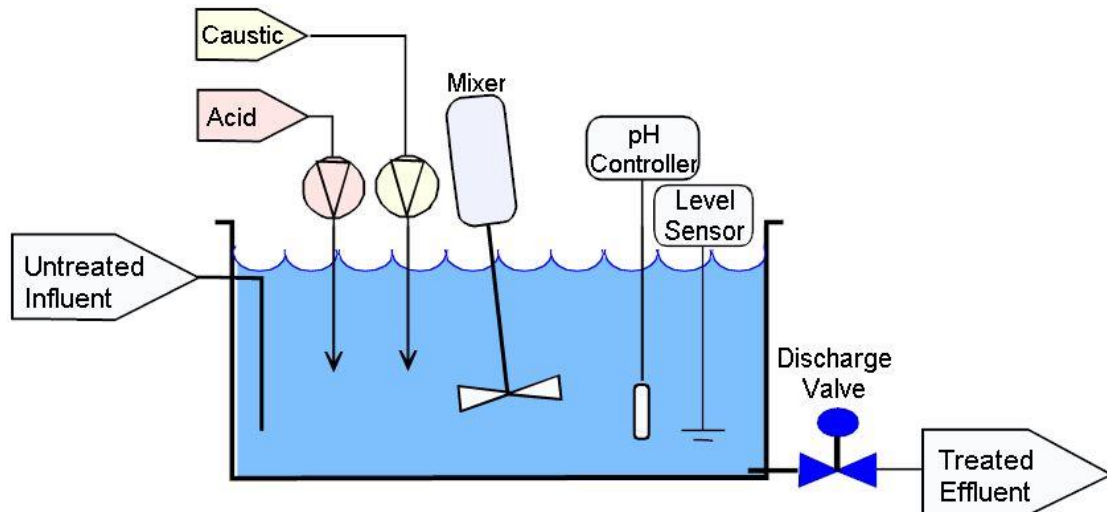


Figure 2.16: Batch pH System Architecture

Figure 2.16 depicts a treatment tank, mixer, acid and caustic metering pumps, a pH probe and controller, a level sensor, and a discharge valve. The influent flow enters the tank anywhere that is convenient and exits the tank via gravity near the bottom wherever a port can be conveniently located. Although the Flow Through system shown above appears to be almost identical to the batch system shown here they operate in very different modes.

In Batch system, the untreated influent fills the tank to the high level point as measured by the level sensor. Once the tank is full the pH adjustment process begins much in the same way that the Flow Through system performed. The difference, however, is that a large batch volume is treated in one cycle. Once the tank contents are within the acceptable discharge range, and have been for a minimum period of time the effluent Discharge Valve opens thereby draining the tank via gravity. Once the tank is drained the cycle repeats. The advantage of a batch system is that no effluent can flow to drain until the effluent meets the discharge criteria. No matter how large or well built a Flow Through system may be there is no way to ensure that this is the case. Batch systems are far more suitable for effluents that may be characterized by large swings in influent pH, concentrated discharges, or large swings in flow rate. The

majority of the systems that we install are batch systems since they are inherently safer and capable of handling streams of any composition.

2.3.1.3 pH Neutralization Process

pH is the measure of free acidity or free alkalinity of water as shown in Figure 2.17. Measured on a scale of 0 to 14, solutions with a pH of less than 7.0 are acids while solutions with a pH of greater than 7.0 are bases as showed in figure 2.17. In very simple terms bases are used to neutralize acids, while acids are used to neutralize alkalis (the term caustic, alkaline, alkalis, or base, although not truly synonymous, are often used interchangeably). The by-products are normally salts (which may or may not be soluble) and water.

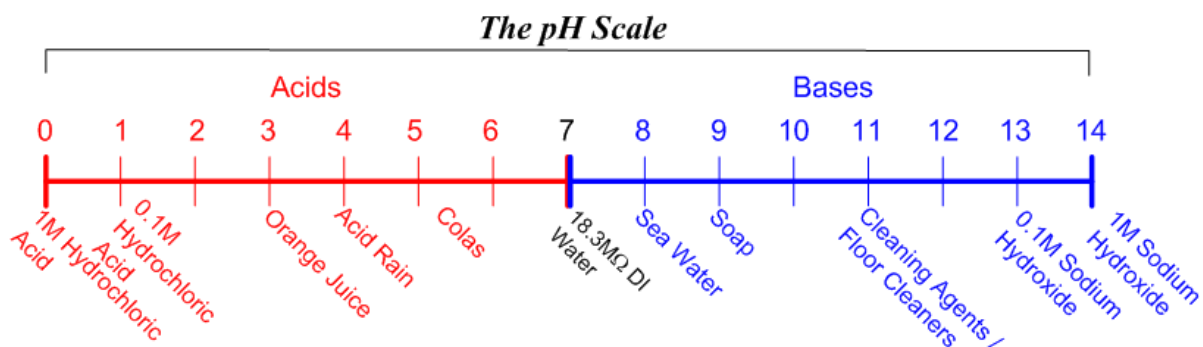


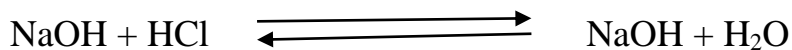
Figure 2.17: pH Scale Measurement

The pH scale measures how acidic or basic a substance is. The pH scale ranges from 0 to 14. A pH of 7 is neutral. A pH less than 7 is acidic. A pH greater than 7 is basic.

The pH scale is logarithmic and as a result, each whole pH value below 7 is ten times more acidic than the next higher value. For example, pH 4 is ten times more acidic than pH 5 and 100 times (10 times 10) more acidic than pH 6. The same holds true for pH values above 7, each of which is ten times more alkaline (another way to say basic) than the next lower whole value. For example, pH 10 is ten times more alkaline than pH 9 and 100 times (10 times 10) more alkaline

than pH 8. Pure water is neutral. But when chemicals are mixed with water, the mixture can become either acidic or basic. Examples of acidic substances are vinegar and lemon juice. Lye, milk of magnesia, and ammonia are examples of basic substances.

In a neutralization reaction between Sodium hydroxide (NaOH) is treated with hydrochloric acid (HCl), then salt (NaCl) and water H₂O are formed. The chemical equation for the reaction is as:



The process can be considered as a continuous stirred tank reactor (CSTR) to neutralize a strong acid with a strong base manipulated by a control valve. The process consists of an influent stream (HCl), reagent stream (NaOH) to regulate the pH of the effluent stream. The components of a general control system as well as the specific features and problems of pH-control are clarified, with an example consisting of a simple but excessively instrumented neutralization process (Figure 2.18), where strong base (control flow) is used for neutralising acidic liquid (inflow) in order to achieve neutral outflow. The inflow *FI*, its contents and the outflow *F3* change in an unpredictable manner and these variations are the main disturbances in the neutralization process. As a result of the changing flows the reactor volume *V* also changes (it can be assumed that there are safeguards so that the vessel never overflows or becomes empty). The only freely manipulated variable (control variable) is the valve opening

The control reagent (strong base) concentration can be assumed constant as well as the pressure inside the control flow pipe. The physical limits of the variables are never encountered (inflow is always acidic, there is always liquid in the vessel and it never overflows, etc.). The block diagram structure of the neutralization process can be built up from simple unit blocks. For instance, the inflow acidity has an obvious effect on the pH value inside the reactor. With crude

simplification, it can be said that the pH-value of the inflow (pH_1) affects the reactor pH-value (pH_R), which, in turn, has an effect on the outflow pH-value (pH_3)

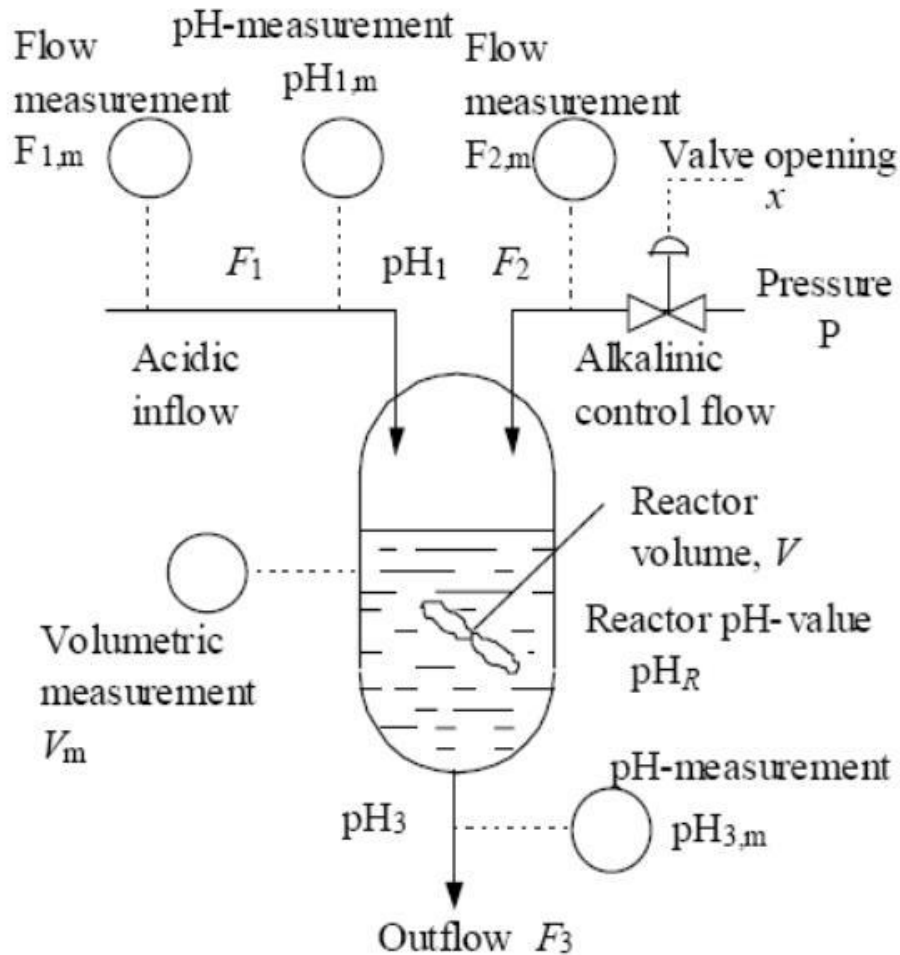


Figure 2.18: Schematic of a pH Neutralization Process (Kambale, *et al.*, 2015).

Acidic and basic are two extremes that describe chemical property chemicals. Mixing acids and bases can cancel out or neutralize their extreme effects. A substance that is neither acidic nor basic is neutral.

2.3.1.4 Fuzzy Inference System

Fuzzy logic Toolbox software support two types of Fuzzy Inference system which are Mamdani and Sugeno fuzzy Inference System.

a. Mamdani Fuzzy Inference System

Mamdani fuzzy Inference System was first introduced as a method to create a control system by synthesizing a set of linguistic control rule obtained from experienced human operators. These output fuzzy sets are combined into a single fuzzy set using the aggregation method of the Fuzzy Inference System (FIS). Then, to compute a final crisp output value, the combined out fuzzy set is defuzzified using one of the method. The reason while Mamdani Model in used in fuzzy logic system because each individual rule in a Mamdani System is intended to shape the final function somewhat locally, i.e. individual rule modify the input- output space without necessarily altering the relation on other region. Mamdani system can look particularly appealing because they are designed to incorporate expert knowledge in the form of IF-THEN rules expressed in natural language. While this is an attractive feature for modelling and simulation of social and complex system, the Mamdani system provides a function that complies with its generating set of IF- THEN rules, which is different from finding the relation of the rules. The inference process of a Mamdani System is described in the Fuzzy inference Process.

The advantage of Mamdani system is that it is intuitive, well- suited to human input; more interpretable rule base and have a widespread acceptance.

b. Sugeno Fuzzy Inference Systems

Sugeno fuzzy inference also referred to as Takagi-Sugeno-Kang fuzzy inference. It uses a singleton output membership functions that are either constant or a linear function of the value of the input values. The defuzzification process for a Sugeno System is more computationally efficient compare to that of Mamdani system, since it uses a weighted average or weighted sum of a few data points dimensional area. Some of the advantages of the Sugeno system is that it is computationally efficient; work well with linear techniques such as PID

Controller; it also work well with optimization and adaptive techniques; it guarantee output of surface continuity and it well- suited to mathematical analysis.

2.4 Related Work

In this section, the work examines the application of CSTR in industrial treatment of wastewater in terms of pH-regulation using different techniques, this is known as neutralization. Some research works have been presented in literature using some control approach to regulate the pH (acidity and alkalinity) of wastewater.

Ram, Kumar and Meenkshipriya (2016) designed a Proportional Integral and Derivative (PID) algorithm using different tuning approaches named Branical method, Smith method, and Marlin method for pH neutralization process. The performances of the various approaches were initially compared in terms of peak overshoot and settling time. It was found that Branical et al method outperformed the others. Also, a comparison performance test in terms of error indices such as Integral Absolute Error (IAE), Integral Square Error (ISE) and Integral of Time weighted Absolute Error (ITAE) of Branica et al. method was also comparatively less. This, the authors maintained that Branical. method was better than other two tuning approaches for pH Neutralization Process.

In Ram *et al.* (2016) designed and applied different control strategies to regulate pH neutralization process that was modelled based on the reaction between strong basic solution (NaOH) and strong acidic solution (HCl) in CSTR. The authors designed PID, Particle Swarm optimization (PSO) based Integral-Proportional and Derivative (I-PD), and Model Predictive Control (MPC) to solve problem of

nonlinear pH neutralization process. It was observed that the PSO based I-PD provided better response than the other two.

Jose, Anthony and Isaac (2013) developed pH control process so as to find its feasibility of applying advanced control strategy to it. The modeling of the pH process in continuously stirred tank reactor was presented. The objective of the pH control was to maintain the pH value of the process at desired value during transient operations by manipulating the alkaline flow rate. Nevertheless, it was found that control performance improvement or application of advanced control schemes required optimizing several process parameters. These parameters are the sampling time, tuning settings. Advanced control strategies such as model reference adaptive controller, and Model Reference Adaptive Control (MRAC) with FUZZY logic controller, MRAC with Neural network were developed. A comparative study of these control methods was carried out.

Ishaq, (2001) presented dynamic modelling and control of a semi-batch wastewater neutralization process. The wastewater neutralization process was modeled based on the reaction between a strong basic solution and a strong acidic solution in a semi-batch reactor. It was assumed that the wastewater was acidic in nature. The semi-batch system has two input streams and a mixing tank which contained an initial amount of acid solution. Simulation was carried out study the dynamics and control of the model formulated. The objective of the control was to regulate the base flow, while keeping the acid flow rate constant, until the pH in the mixing tank stabilized at pH 7. A digital PI control algorithm was used as the controller, and the control simulation was performed in Matlab's Simulink environment.

Alwan (2008) performed experimental investigations to study the pH-control problems of industrial electroplating wastewater treatment plants. It stated that conventional PID control system could handle most problems and disturbances during the normal operation of the water treatment. However, an innovative

technique for treatment process known as Aquasil technology was used by the author to carry out the experimental investigation. It was found that the Aquasil technology proved to more effective for a wide range of pH.

Manoj, Varukumar,, Manoranjith and Radihika, (2005) developed a fuzzy based control system for pH control. Since the pH control process is nonlinear in nature, maintaining the pH value in the process is difficult to achieve by the desired transient response. Hence, to overcome these problems a fuzzy based system was designed. It was found that the fuzzy system offered more efficient transient response than conventional controllers.

Ghatule (2017) fabricated a laboratory scale CSTR and then developed a PID algorithm for regulating the CSTR process in which acid-base reaction takes place. The PID system was used to regulate pH of solution which was a neutralization process. The PID was able to control and maintain the pH precisely.

Ahmed *et al.* (2016) designed a temperature control of the continuous stirred tank reactor (CSTR) using three configurations (feedback, feed forward and cascade). For each type of the control systems the controllers used was tuned and its optimum parameters were calculated. For the feedback control system in particular, the proportional integral derivative controller (PID) was selected other than the proportional (P) and the proportional integral (PI) controllers. Also for each configuration a stability analysis is made using (Bode, Nyquist and root-locus) criterions. The system stability was enhanced by the research done (Anna Vasickninovia and Alajos, 2022) by casscading the control structure in industrial application by applying type-1 and type-2 fuzzy fuzzy controller and PID for better performance indices and high energy saving measured by the coolant consumption during control of the tubular chemical reactor.

Table 2.1: Comparisons of several hybrids System Structure

S/N	Types of Hybrid System	Key features	Advantages
1	Adaptive Neuro-Fuzzy Inference System (ANFIS)	<ul style="list-style-type: none"> ▪ Construct input–output mapping based on human knowledge and simulated input–output data pairs ▪ Consist of two parts: antecedent and conclusion 	<ul style="list-style-type: none"> ▪ Minimize error by applying two learning algorithms (back propagation and hybrid) ▪ Able to handle complex processes ▪ Short learning time ▪ Fast in reaching optimum results
2	Hybrid Neural Network (HNN)	Hybrid combination of neural network and first principle model	<ul style="list-style-type: none"> ▪ Fewer nodes ▪ Less training time ▪ Fast convergence
3	Structure Approaching Hybrid Neural Network (SAHNN)	A hybrid type of neural network Use approximate mechanistic equation for characterising the unmeasured variables	Rapid convergence
4	Fuzzy Neural Network (FNN)	Contain theory of fuzzy logic and ANN	Implementation can be either from input to output or output to input
5	Fuzzy-Rough Set (FuREAP)	Based on rough set and fuzzy logic theories	Reduce measurement and data set for reducing cost
6	Fuzzy-Expert Systems (Fuzzy-ES)	Combination of IF-THEN rules with expert Systems	<ul style="list-style-type: none"> ▪ Able to solve problems involving variations in parameters ▪ Reduce quantity of rules

Hybrids System Structure (Jarinah *et al.*, (2015)

The research conducted by Manikandan, Geetha, Jubi, and Jerome (2013) studied fault tolerant fuzzy gain scheduling proportional Integral derivative controller for CSTR. It proposed an adaptive fuzzy PID (FPID) controller for fault-tolerant control of CSTR process in the presence of Sensor and actuator fault. Three cases

of faults were considered which are +25% additives faults in sensor faults in actuator, and fault in both sensor and actuator (double fault).

The performance of the proposed controller was tested experimentally by applying it in a CSTR process. It compared the performance of the proposed controller with that of conventional PID considering in the presence of the three faults scenarios. It then maintained that the proposed controller provided better performance than conventional PID.

In the research done by Vishnoi, Padhee, and Kaur, (2012) studied the controller performance evaluation for concentration control of isothermal continuous stirred tank reactor. A comparative analysis of conventional PID controller and fuzzy controller performance was studied based on time domain analysis. The performance evaluation was performed in terms of percentage overshoot, delay time, rise time, setting time, peak time, Integral square of error (ISE), Integral absolute error (IAE), Integral time absolute error (ITAE). The performance analysis revealed that the hybrid fuzzy provided a satisfactory control effect than conventional controller.

Yong *et al.* (2019) conducted a comparative analysis of tuning a PID controller using intelligent methods. It exploited the limitations of conventionally tuned PID controller and presented intelligent methods for tuning PID based on fuzzy logic, Artificial Neural Network (ANN), adaptive Neuro Fuzzy Inference System (ANFIS) and genetic algorithm (GA). A PID controller was developed using the various intelligent methods for tuning. The controller was used for controlling the concentration of a CSTR process. Simulations were performed considering the various intelligent fuzzy methods and conventional PID controller tuned using Zeigler Nichols (ZN) method in terms of rise time, setting time, overshoot, undershooting and steady state error. The simulation results revealed that the

ANFIS provided the best performance in comparison to other intelligent tuning techniques and conventional Zeigler Nichols method.

Dostàl and Vladimir (2015) studied simulation of cascade regulation of a CSTR. Simulations were performed for primary and secondary loops. The primary loop regulates the reactor concentration of reference reaction product. The secondary loop regulates the reactant temperature. The system architecture is shown is shown in Figure 2.19. The coolant flow rate serves as a common manipulated input. The primary regulation algorithm is a proportional controller whose gain is calculated employing simulated or measured steady state properties of the reactor. An adaptive control algorithm is used at the secondary loop.

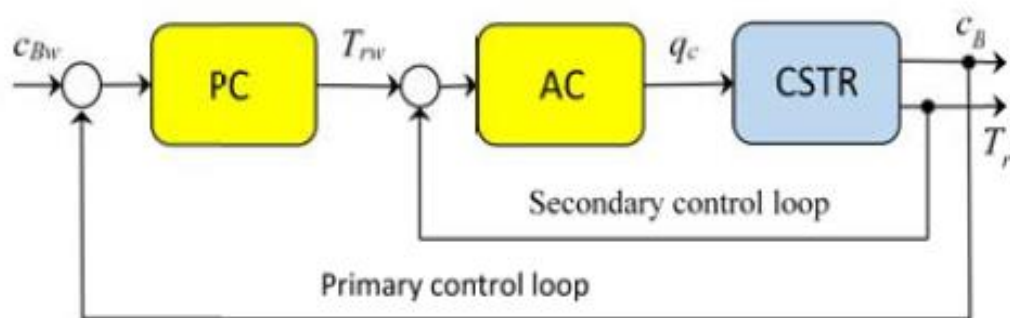


Figure 2.19 Cascade control scheme (Dostàl *et al.*, 2015).

Jorge, Romero- Bustamante Moguel-Castaneda Hector Pueblea Eliseo and Hernandez-Martinez (2017) proposed a robust model-based cascade algorithm for concentration regulation in chemical reactors. The algorithm is based on two loop regulation in a concentration and temperature scheme. State-observers are introduced to provide robustness to the controller via the estimation of lumped model uncertainties. The second loop is introduced with a favourable choice of the Arrhenius equation as a virtual control input. The control design is illustrated based on two cases studies, a tubular reactor exhibiting two temperature dynamics (quasi-linear and hot-spot) and a CSTR presenting multiple steady-states. Numerical results indicate that proposed controller is robust in the face of external disturbances, operation changes, and model uncertainties.

Pablo, Lopez Perez and Richardo, Aguilar-Lopez (2009) presented a methodology for the design of a class of integral high order sliding mode control algorithm applied to a class of continuous chemical reactor with complex behaviour for temperature tracking purposes. The proposed design is based on the differential geometry framework, where the named reaching trajectory contains a high order sliding mode term in order to diminish chattering. Considering that the proposed technique is model based, an observer based uncertainty estimator is coupled, which provides robustness against model uncertainties and noisy measurements. Numerical simulations are performed in order to show the capacities of the proposed controller, which is compared with other nonlinear methodologies.

In the work of Antonelli and Astolfi (2003), methodologies and tools from Lyapunov theory in the design of bounded control algorithms for the temperature stabilisation of a class of continuous stirred tank reactors with exothermic or endothermic reactions. Simple output feedback control algorithms are designed. It was shown that the developed system yielded global asymptotic stability and do not require precise knowledge of the system parameters. The capabilities of the proposed method to handle manipulated input constraints and model uncertainties are also discussed.

Alshammari, Muhammad and Mahyuddin (2018) surveyed the issues related to modelling problems control strategies of a Continuous Stirred Tank Reactor (CSTR), a highly nonlinear plant containing numbers of stable and unstable operating points is considered. The issues discussed are categorised into regulation, feedback linearization, flatness, observation and estimation as well as challenges related to equilibrium points concerning CSTR. The study discussed the limited capability of a conventional PID controller based on preliminary

description and a dynamic modelling of the nonlinear plant. Moreover, the limitations of the conventional PID are illustrated through a simulation using nonlinear model of CSTR carried out under input constraint and the presence of bounded disturbances. The result shows that a fixed PID will not guarantee consistent performance throughout operating set points (Alshammari *et al.*, 2018). The feedback linearization formalism is presented to prove that only regulation in the neighbourhood of operating point is possible. Non-minimum phase property exhibited by a CSTR is investigated as well. Flatness control is demonstrated as one of the possible linearization control technique achieving the objective of the trajectory tracking.

Upadhyay and Singla (2010) presented a nonlinear feedback controller is analysed for temperature control of continuous stirred tank reactors (CSTRs) which have strong nonlinearities. A regulator mechanism that will make the proper changes on the process to cancel the negative impact that such nonlinearities may have on the desired operation of chemical plant was introduced. A method for adaptive regulation of a continuous stirred tank reactor with output temperature constraint is developed. The structure of the proposed system is shown in Figure 2.20. The algorithm is robust to modelling errors and random disturbances occurring in the system. The controller design is analysed for this situation and two algorithms, Adaptive and PID were analysed so as to ascertain which of them provide most linear response. The conventional PID controllers have difficulty in dealing with problems that appear in complex nonlinear processes Simulation studies give satisfactory results.

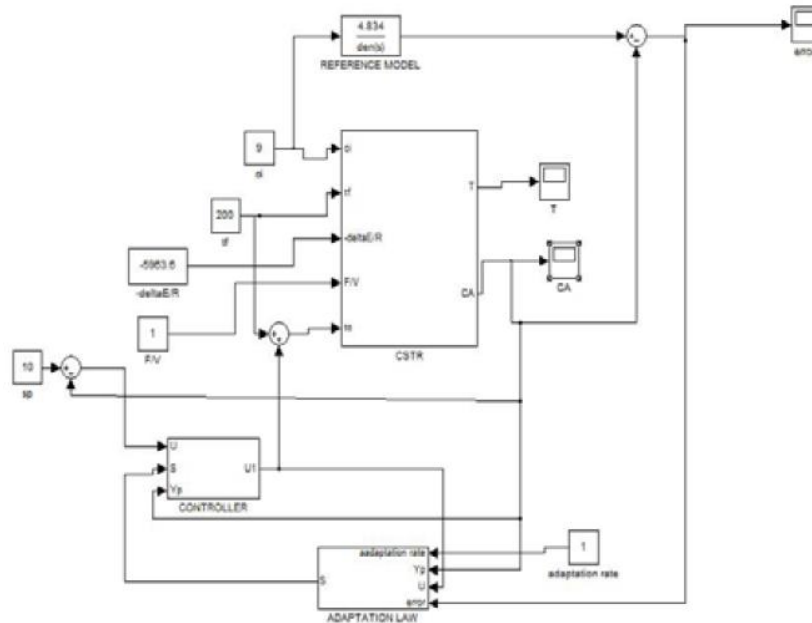


Figure 2.20: CSTR with Adaptive Control (Updadhay and Singla, 2010).

Shrivastava (2012) studied a predictive control strategy based on neural network model of the plant is applied to Continuous Stirred Tank Reactor (CSTR). This system is a highly nonlinear process; therefore, a nonlinear predictive method such as a neural network (NN) predictive control can be a better match to govern the system dynamics. The NN model and the way in which it can be used to predict the characteristics of the CSTR process over a certain prediction horizon are described, and some comments about the optimization procedure are made. Predictive control algorithm is applied to control the concentration in a continuous stirred tank reactor (CSTR), whose parameters are optimally determined by solving quadratic performance index using the optimization algorithm. An efficient control of the product concentration in CSTR can be achieved only through accurate model. Here an attempt is made to alleviate the modeling difficulties using Artificial Intelligent technique such as Neural Network. Simulation results demonstrate the feasibility and effectiveness of the NNMPC technique.

Soukkou *et al.* (2008) designed an optimal system for a CSTR process. A fuzzy optimal control algorithm methodology is applied to the design of the feedback loops of an Exothermic Continuous Stirred Tank Reactor system. The goal of the study is to design a process to obtain optimal structure/gains of the Robust and Optimal Takagi Sugeno Fuzzy Controller (ROFLC). It is expected that the manipulated signal obtained will minimize a performance index, which is a function of the tracking and regulating errors, the quantity of the energy of the control signal applied to the system, and the number of fuzzy rules. The genetic learning is proposed for constructing the ROFLC. The chromosome genes are arranged into two parts, the binary-coded part contains the control genes and the real-coded part contains the genes parameters representing the fuzzy knowledge base. The architecture of the developed system is shown in figure 2.21. The effectiveness of this chromosome formulation enables the fuzzy sets and rules to be optimally reduced. The performances of the ROFLC are compared to these found by the traditional PD controller with Genetic Optimization (PD_GO). Simulations results show that the proposed ROFLC and PD-GO the design specifications.

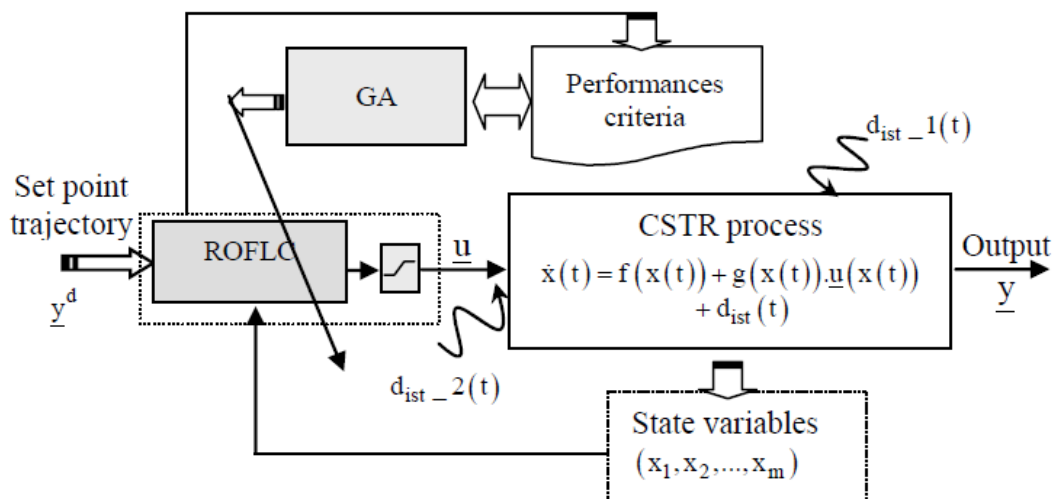


Figure 2.21: Control and Optimization Structure (Soukkou *et al.* 2008).

Raju *et al.* (2021) presented the design of a Dynamic Matrix Controller (DMC) analysed for concentration regulation of Continuous Stirred Tank Reactors (CSTRs) which have strong nonlinearities the two controllers DMC and PID and analysed. The control of the concentration of one chemical with the help of other has been analysed. Model design and simulation are done in MATLAB/SIMULINK, using programming. The concentration control is found better controlled with the addition of DMC to the algorithm of PID improved the system performance compare to the case of using only PID.

Putrus (2011) presented a dynamic characteristics and control of a jacketed continuous stirred tank reactor (CSTR) using different control strategies, conventional feedback control (PI and PID), and neural network (NARMA-L2, and NN Predictive) control. The dynamic model for CSTR process is described by a first order lag system with dead time. The optimum tuning of control parameters are found by two different methods; Frequency Analysis Curve method (Bode diagram) and Process Reaction Curve using the mean of Square Error (MSE) method. It is found that the Process Reaction Curve method is better than the Frequency Analysis Curve method and PID feedback controller is better than PI feedback controller. The results show that the artificial neural network is the best method to control the CSTR process and it is better than the conventional method because it has smaller value of mean square error (MSE). MATLAB program is used as a tool of solution for all cases used in the present work.

Lloyds, and Ahmed (2018) designed a cascade control strategy for regulating the temperature inside a jacketed exothermic continuous stirred tank reactor. The ultimate gains for the secondary and primary loops were determined using direct substitution, Routh-Hurwitz, Root-locus and Bode methods. To tune the controllers, the secondary controller was first tuned, and then the primary controller was tuned using Ziegler-Nichols technique.

Yadav and Trivedi (2014) designed a nonlinear feedback controller temperature control of continuous stirred tank reactors (CSTR) which have strong nonlinearities. A Particle Swarm Optimization (PSO) algorithm based PID controller tuning is used for the temperature control of CSTR, The performance is analysed for the controllers designed by using PID, PID-GA and PID-PSO methods, identifying a suitable controller and enhanced the system performance. The Integral Square Error (ISE) criterion is used to guide PSO algorithm to search the controller parameters like K_p , K_i , K_d . A comprehensive simulation is carried out with PID and GA controller Structures. The tuning structures of the PID-GA and PID-PSO control loops are shown in Figures 2.22 and 2.23. The simulation results show that the PSO based PID controller tuning approach provides better performance compared to other conventional PID tuning methods.

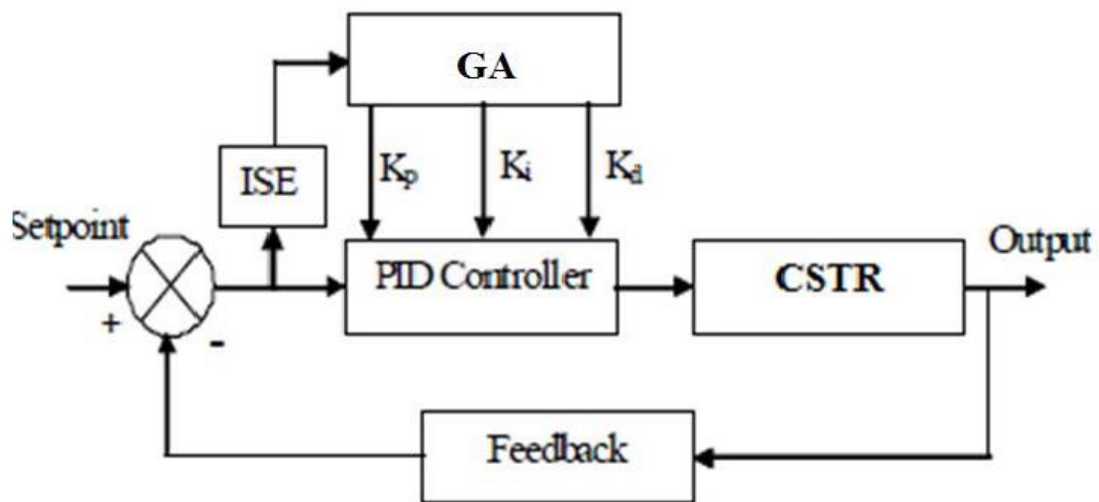


Figure 2.22: Tuning of PID-GA Controller (Yadav and Trivedi 2014).

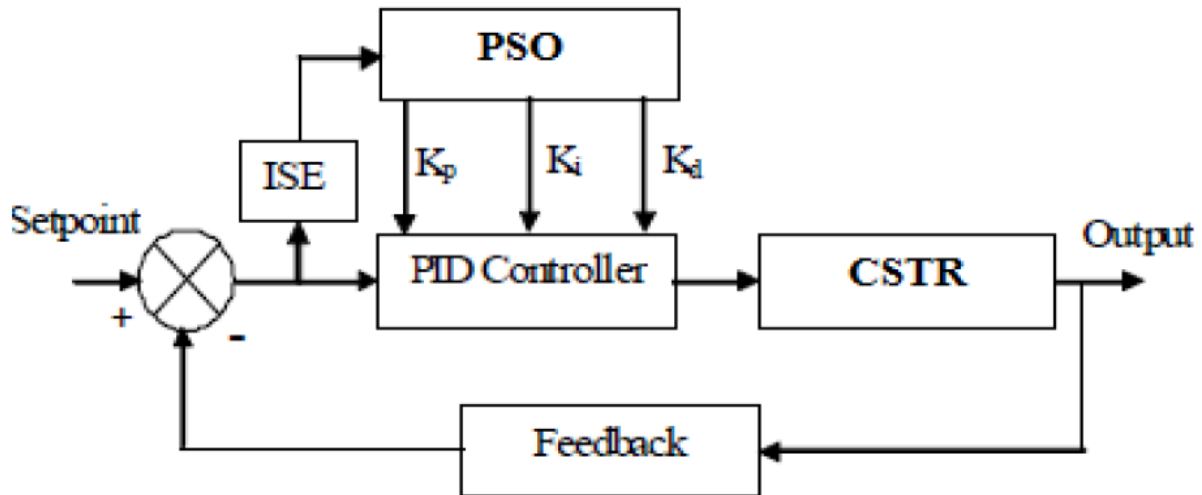


Figure 2.23: Tuning of PID-PSO Controller (Yadav and Trivedi 2014).

Zhao *et al.* (2015) proposed an output feedback terminal sliding mode control (TSMC) framework which is developed for CSTR, and can estimate the system states and stabilize the system output tracking error to zero in a finite time. The corresponding stability analysis is presented in terms of the Lyapunov method. Illustrative examples are demonstrated by using Matlab simulations to validate the effectiveness of the proposed approach.

Bharathi and Selvakumar (2013) developed a Programmable Logic Controller (PLC) for continuous stirred tank reactor based blender controlled application. A PLC is designed and applied to oil movement and storage unit that comprises of two blenders for Euro 2 and Euro 3 grades, inputs from six (6) storage tanks. The objective is to maintain maximum reliability.

Saad *et al.* (2013) proposed a model for a CSTR designed using a mathematical model to derive the system parameters. Then, the linear model was derived from the nonlinear model. A conventional PI controller and PID controller for continuous stirred tank reactor are proposed to control the concentration of the linear CSTR. The simulation study was done in MATLAB/SIMULINK. The best controller has been chosen by comparing the criteria of the response such as

settling time, rise time, percentage of overshoot and steady state error. From the simulation result the PID controller has a better performance than conventional PI controller.

Muqheet *et al.* (2015) proposed and develop an experimental set-up of PLC and SCADA based mixing and control of Continuous Stirred Tank Reactor (CSTR). The set-up automatically monitors different parameters with the help of SCADA HMI. The PLC is connected to the level sensors and solenoid valves are controlled based on the level in the tank reactor. This model also has the stirrer control unit which continuously stirs the reactor and mix the reactants. The PLC used is Allen Bradley Micrologix 1200 and the SCADA software is reviewed. Thus the level in the CSTR is constantly monitored and brought to a constant level as required for the operation. The automation is further enhanced by constant monitoring using SCADA screen which is connected to the PLC by means of RS232 communication cable.

Maayedukondalu (2015) studied steady state simulation of continuous stirred tank reactor system (CSTR) using aspen plus. The author investigated production of ethyl benzene using ethylene and benzene as the reactants. The effect of feed flow rate on steady characteristics of CSTR is studied and optimum feed flow rate is obtained in order to maximize the conversion and yield. The facts that the effect of reactor temperature was always going to be crucial in this exothermic reaction mechanism, optimum values were evaluated. The optimum reactor volume was also found out from the simulation results. The optimum temperature was predicted to be 430 K and the optimum reactor volume was found to be 100 m³.

Vojtěšek and Dostál (2015) presented a program in MATLAB's Graphical User Interface (GUI) which can be used for investigating of the static and dynamic behaviour and adaptive control of the nonlinear system represented by

Continuous Stirred Tank Reactor (CSTR). Program provides computation for various values of the input variables which made this program universal for the wide range of the users without theoretical knowledge about the modeling, simulation and control.

Rahmat *et al.* (2011) presented two different control strategies based on the combination of a novel socio-political optimization algorithm, called Imperialist Competitive Algorithm (ICA), and concept of the gain scheduling performed by means of the least square and fuzzy logic approaches. The objective is to control the temperature of the CSTR in presence of the set point changes. The works followed with designing those controllers and simulating in MATLAB software. The performances of the proposed controllers have been considered based on the Sum of the Square Error (SSE) and Integral Absolute Error (IAE) Criteria. The results clearly indicate that both suggested control strategies offer an acceptable performance with respect to the functional changes of the process. In other word, robustness of the proposed methods in dealing uncertainties throughout the tracking of the reference signal take the highlighted point into account. Furthermore, fuzzy based structure strategy gives the more flexibility and precise behaviour in control action in comparison to the least square based approach.

Ishwar and Rajenda (2016) examined different controllers such as PID, Cascade, IMC controller, Differential Evolution, and Fuzzy Logic Controllers have been evaluated. The objective is to control the temperature of CSTR in presence of the set point and analyzed which controller provides the most linear response. Model design and simulation are done in the MATLAB/SIMULINK software. The response of the CSTR, which is observed with applying step input and improved by designing conventional and intelligent controllers.

Naikwad and Dudul (2009) designed a focused time lagged recurrent neural network (FTLR NN) with gamma memory filter to learn the subtle complex dynamics of a typical CSTR process. Continuous stirred tank reactor exhibits

complex nonlinear operations where reaction is exothermic. The authors observed from literature review that process control of CSTR using neuro-fuzzy systems was attempted by many, but optimal neural network model for identification of CSTR process is not yet available. They stated that since CSTR process includes temporal relationship in the input-output mappings, time lagged recurrent neural network is particularly used for identification purpose. The standard back propagation algorithm with momentum term has been proposed in this model. The various parameters like number of processing elements, number of hidden layers, training and testing percentage, learning rule and transfer function in hidden and output layer are investigated on the basis of performance measures like Mean Square Error (MSE), Normalized Mean Square Error (NMSE), and correlation coefficient on testing data set. Finally effects of different norms are tested along with variation in gamma memory filter. It is demonstrated that dynamic NN model has a remarkable system identification capability for the problems considered in this paper. Authors maintained that a major contribution of the study is to demonstrate that FTLR NN with gamma memory filter can be used to learn underlying highly nonlinear dynamics of the system.

Ang *et al.* (2018) presented effective and computationally less intensive method for the stabilization of a Two-Input Two-Output (TITO) system. A discrete-time model with its parameters estimated once every sample time is used to describe each of the Input-Output (IO) models of the TITO. The discrete-time models are used in conjunction with two proposed performance indices to yield control signals to improve the overall system performance. The ability of the proposed self-tuning control scheme to reject small and large disturbances, and to overcome strong couplings in Multi-Input Multi-Output (MIMO) systems were investigated and presented. Simulation tests showed that the proposed system enhanced self-tuning control scheme is suitable for Single-Input Single-Output (SISO) and MIMO systems. The proposed scheme is shown in Figure 2.24.

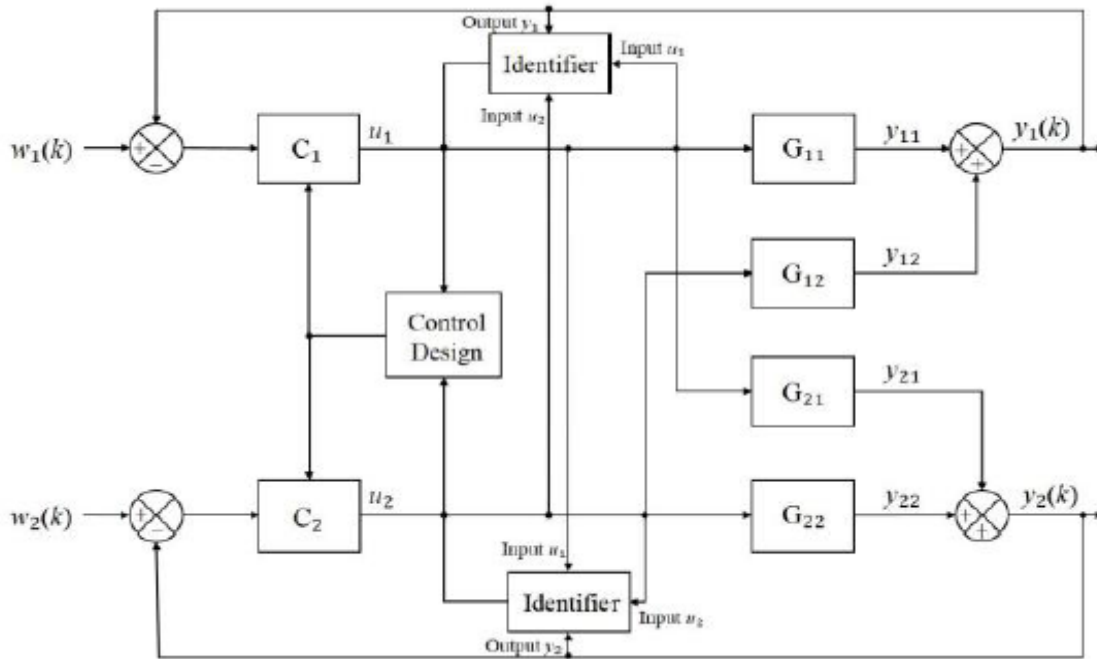


Figure 2.24: TITO Self-tuning Control Scheme configuration of the Proposed Control Design Method (Ang *et al.*, 2018).

Ang *et al.* (2018) proposed two linear Artificial Neural Network (ANN)-based controllers for the control of a nonlinear system. The optimization and pole placement control design strategies are used. For both strategies, linear ANNs are used to model a Continuous Stirred Tank Heater (CSTH) process. Third-order discrete-time models or Adaline ANNs are used whose parameters are updated at every sampling time to provide adaptability to the controllers. Simulation tests show that the linear ANN-based controllers can overcome strong nonlinear coupling effects, reject step disturbances, and provide adequate damping to the subsystem interactions. Both ANN-based control design strategies are shown to be interchangeable with each other for the control of the CSTH process. The proposed system architecture is shown in Figure 2.25.

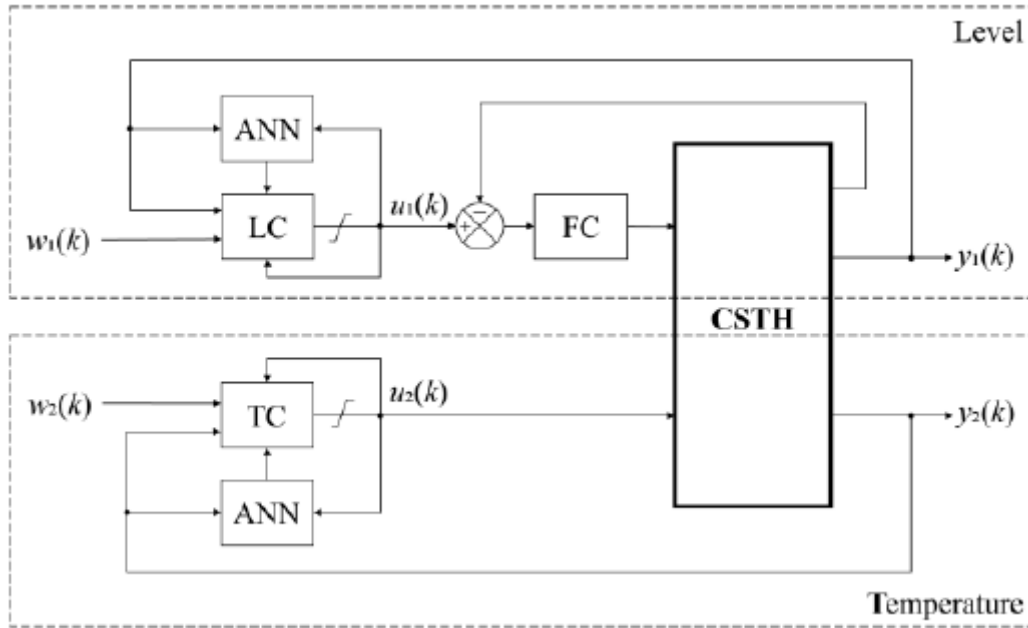


Figure 2.25: Proposed CSTH TITO Control Scheme Configuration (Ang *et al.*, 2018).

Maayedukondalu (2015) presented steady state simulation results of a continuous stirred tank reactor (CSTR) system using Aspen-plus simulation. The reaction, investigated work was production of ethyl benzene using ethylene and benzene as the reactants. The author has studied the effect of feed flow rate on steady state behaviour of CSTR and also obtained the optimum feed flow rate in order to maximize the conversion and yield with special consideration of consecutive reactions where one product was desired and another one was undesired. Since the effect of reactor temperature was always going to be crucial in the exothermic reaction mechanism, optimum values were evaluated. The optimum reactor volume was also found out from the simulation results. The optimum temperature was predicted to be 430 K and the optimum reactor volume was found to be 100 m³.

Navid, Mehdi, Masoud and Saeed (2015) proposed an optimal Active Disturbance Rejection Control (ADRC) based on using Asexual Reproduction Optimization (ARO) to control the temperature of a nonlinear CSTR. The

parameters of non-isothermal continuous stirred tank reactor (CSTR) are varying with time caused by fouling and the deactivation and regeneration of the catalyst. Furthermore, in the exothermal region, dynamic behaviour of this reactor is unstable. Therefore, designing an efficient controller in this complicated situation is difficult and challenging. ADRC is used as a robust method to control the temperature of CSTR in the situation that the CSTR parameters are varying with time. The parameters of ADRC are difficult to adjust and if these parameters tuned properly, it performs more efficiently in setpoint tracking and disturbance rejection. Controller design is represented as an optimization problem. The parameters of ADRC are tuned by ARO and then by Particle Swarm Optimization (PSO). The performance of ADRC tuned by ARO (ADRC-ARO) is compared with the performance of ADRC tuned by PSO (ADRC-PSO) and PID controller. The simulation results indicated that the proposed ADRC-ARO method is robustness and better performance in both setpoint tracking and disturbance rejection with faster response time and less settling time.

Ramli *et al.* (2020) introduced the concept of Fuzzy Logic approach towards the control system of non –isothermal continuous stirred tank reactor (CSTR). Simulation was carryout using the MATLAB SIMULINK, and comparison with PID controller in order to justify the effectiveness of the technology concept in the control is made using PSO and PID and PD.

Suvire *et al.* (2017) presented a novel controller for trajectory tracking in a CSTR. The system was evaluated and compared with others previously proposals in literature. The controller design was derived from a rather simple strategy based on a linear algebra approach. The main advantage of the proposed methodology is that the control action is easily obtained by solving a system of linear equations, and only the knowledge of the system model and the reference trajectory are necessary. The Monte Carlo Randomized Method is used for tuning the controller parameters and testing the system behaviour under modelling

errors. The controller performance is evaluated through several tests and compared with other controller reported in literature. In addition, proofs of convergence to zero of the tracking error are presented.

Hoang *et al.* (2013) used a passivity-based approach (PBA) and tools from Lyapunov theory to design a nonlinear controller for the asymptotic stabilization of a class of non-isothermal Continuous Stirred Tank Reactors (CSTR) around any desired stationary point. The convergence and stability proofs are derived in the port Hamiltonian framework. Asymptotic observers that do not require knowledge of reaction kinetics are also proposed for a system with incomplete state measurement. Numerical simulations are given to illustrate the application of the theoretical results to a CSTR with multiple steady states.

In the study by Ikutegebe *et al.* (2015), a Continuously Stirred Tank Reactor (CSTR) was modelled and the reactor performance analysed with a view to coming up with an essential parametric quantity useful for assessing control at the steady-state design stage. The steady-state design of CSTR and its implementation, using Matrix Laboratory (MATLAB) program for hypothetical reaction case and the commercial software Aspen Plus for a real chemical case was performed. The approach adopted considered two separate first-order irreversible exothermic reaction processes, implemented in MATLAB® and Aspen Plus software respectively and a target conversion and reactor stability ratio (RSR) determined. Results from the study revealed that at varied temperatures (320-360K) under steady-state design, a suitable target conversion of 95% and RSR values less than 0.50 present minimal control problems with irreversible exothermic reactions for both hypothetical and real chemical systems. Conversely, reactor design for low conversion and maximum temperatures can present control problems with exothermic reactions due to larger reactor size and low heat transfer area. The authors stated that the key to improved reactor

performance with minimal control problems was designing at a reasonable low RSR value and providing excess heat transfer area so that disturbances can be properly handled.

In the review of PID Controller with intelligent System research conducted by (Jasvir and Gursewak, 2017); their work brings an examination concerning the utilization of a PSO calculation to tune the PID controller for a semi-dynamic vehicle suspension framework joining magnetor rheological (MR) damper to enhance the ride solace and vehicle security. The proposed suspension framework comprises of a framework controller that decide the coveted damping power utilizing a PID controller tuned utilizing PSO, and a nonstop state damper controller that gauge the command voltage that is required to track the coveted damping power. The proposed PSO PID controlled suspension is contrasted with both the customary PID controller and the inactive suspension frameworks. Framework execution criteria are assessed in both time and recurrence areas, with a specific end goal to evaluate the achievement of the proposed suspension framework. The recreated comes about a mirror that the proposed PSO PID controller of the MR-damped vehicle suspension offers a huge change in ride solace and vehicle strength.

Nemani (2015) studied simulation of CSTR using orthogonal functions. The reactions in a CSTR are modelled by a set of differential equations which are discretised to a set of algebraic equations by orthogonal functions. The author stated that many numerical methods such as Runge-Kutta method, Euler method have successfully converted the set of differential equations into a set of algebraic equations in literature. But the orthogonality of the functions has never been used for discretization. The author then used orthogonal functions to simulate chemical reactors using the principle of orthogonality. Block-pulse functions have been used to obtain the dynamics of concentration and temperature of the continuous stirring tank reactors (CSTR's). In addition, a recurrence relationship developed using block-pulse functions and triangular functions were used in solving linear

and non-linear system of differential equations. The major importance of orthogonal functions lies in its application to optimal control of systems. A recursive algorithm developed using block pulse functions has been applied to a linear control problem to determine the states and optimality criterion.

2.5 Summary of Related Work

In table 2.2 shows the comprehensive summaries of the related work done in this area of research.

Table 2.2: Summary of Related Work

Author	Title of Work Done	Result Achieved	Limitation
Ram <i>et al.</i> (2016)	Designing a Proportional Integral and Derivative (PID) algorithm using different tuning approaches	The performances of the various approaches were initially compared in terms of peak overshoot and settling time. It was found that Branical <i>et al.</i> method outperformed the others, Smith <i>et al.</i> method, and Marlin method for pH neutralization process.	The work specially combines P, PI, PD Algorithms which may have similar characteristics when integrated, but failed to consider others hybridize control algorithm, to determine the behavior of the turning approaches.
Park <i>et al.</i> (2021)	Online Turning of PID Controller using a Multilayer Fuzzy Neural Network design for Quadcopter	The development of a method for Online Turing quadcopter PID parameters was enhanced with better performance achieved with MFNN	The work did not model the performance of MFNN, they uses the techniques MFNN to improve the PID shortfalls.
Alwan (2008)	Experimental investigations of pH-control problems in industrial electroplating wastewater treatment plants	It stated that conventional PID control system could handle most problems and disturbances during the normal operation of the water treatment.	The comparative studies carry out did not perform analysis using the some set parameters to evaluate

Manoj et al (2004)	Development of a fuzzy based control system for pH control	Since the pH control process is nonlinear in nature, maintaining the pH value in the process is difficult to achieve by the desired transient response	There are difficulties to desired transient response.
Jhon <i>et al.</i> (2018),	“A Tuning Proposed for direct Fuzzy PID Controllers Oriented to industries Continuous Process	The Work presented a simple but effective systematic method for many direct fuzzy PID tuning controllers based on static gain calculation of linear sub- models and controller scaling factors.	It maintained that the proposed approach provided a considerable reduction in the tuning time and neglect improving the performance of processes whose operating points changes
Jasvir and Gursewak, 2017	Review on PID Control Algorithms with Intelligent System	The Work Proposed PSO-PID Controlled Algorithm in CSTR PLANT to streamline the best Outcome of the Speed in the Chemical Plant	<ul style="list-style-type: none"> ▪ The work centre basically on Speed of the CSTR Plant ▪ The Algorithm contrasted with both the PID Controller and the inactive Suspension framework.
Jarinah <i>et al.</i> (2015)	Artificial Intelligence Techniques applied as estimator in Chemical Process Systems	Applied ANN, Fuzzy logic, Expert Systems and Genetic Algorithms to build estimators in several chemical process system	<ul style="list-style-type: none"> ▪ The hybrid Algorithms where only applied in design and development Process in the Chemical Plant. ▪ There was no parameter used to analysis the performance of the chemical plant
Pathak <i>et al.</i> , (2010)	Optimal control of CSTR	They developed a second order dynamic model for CSTR. A nonlinear Model	<ul style="list-style-type: none"> ▪ The addition of MPC strategy to CSTR control loop, an optimal

		Predictive Control (MPC) of CSTR was performed. An optimal state estimation was carried out using Extended Kalman Filter (EKF).	temperature was control inside the reactor was achieved. <ul style="list-style-type: none"> Comparison was not done by other conventional control methods.
Maheshwari <i>et al.</i> (2016)	Design and Analysis of PID controller for CSTR process.	The objective of the study was to control temperature and load disturbance rejection of CSTR.	<ul style="list-style-type: none"> Simulation result showed that the PID controller provided a less percent overshoot with efficient load disturbance rejection with a minimum setting time
Thulasidharan <i>et al.</i> , (2017)	Tuning of PID Control Algorithm using optimization techniques for a multi-input-multi-output (MIMO) process	The work considered two processes which consists of Quadruple Tank process and CSTR process. Dynamic model of the two processes was performed by linearization of the system due to MIMO process. In order to tune the controller parameters, two optimization techniques consisting of PSO, and GA were use	Both optimization techniques provided almost the same response with slight difference in their peak overshoot values.
Bustamante <i>et al.</i> (2017)	Proposed a Robust Model-Based cascade algorithm for concentration regulation in chemical reactors	<ul style="list-style-type: none"> The algorithm is based on two loop regulation in a concentration and temperature scheme. State-observers 	Numerical results indicate that proposed controller is robust in the face of external disturbances, operation changes, and model uncertainties.

		are introduced to provide robustness to the controller via the estimation of lumped model uncertainties.	
Darius and Sivagam asundan (2016)	Design and implementation of Controllers' algorithm for CSTR process	<ul style="list-style-type: none"> ▪ The controller provided the most efficient one for CSTR process temperature control was chosen. ▪ Various process variables like temperature, concentration were controlled in this Process using different control techniques. 	<ul style="list-style-type: none"> ▪ It carried out quantitative comparison of the closed loop response stimulations for the MPC control techniques base on IAE and ISE criteria
Poovaras san <i>et al.</i> (2014)	Design of fractional order PID controller for a CSTR process.	<ul style="list-style-type: none"> ▪ It proposed the application of fractional order PID (FOPID) controller in CSTR process controls. ▪ It employed soft computing techniques which comprises genetic algorithm (GA) and particle swarm optimization (PSO) to model 	<ul style="list-style-type: none"> ▪ The performance was analyzed with respect to Integral Square Error (ISE), ISE doesn't considered reference output, set points

		<p>the CSTR and for obtaining model parameters.</p> <ul style="list-style-type: none"> It maintained that the developed model was able to compensate for the nonlinearity present in the CSTR 	
Jimisha and Shinu (2016)	Analysis of Conventional Controllers for Pressure rated Modified CSTR System	<ul style="list-style-type: none"> The objective is to control the pressure of CSTR in presence of the set point and analyze the controller that provides the most linear response with PI & PID. Model design and simulation are done in the MATLAB/SIMULINK software. The response of the CSTR, which is observed with applying step input and improved by designing conventional and intelligent controllers 	<ul style="list-style-type: none"> The simulation result did not consider time delay in the output response The high level of uncertainty of input parameters based on the hybrid algorithms analysis.
Ang <i>et al.</i> ,	Proposed two linear Artificial Neural	Simulation tests show that the linear ANN-	<ul style="list-style-type: none"> The Simulation and studies did not

(2018)	Network (ANN)-based controllers for the control of a nonlinear system.	based controllers can overcome strong nonlinear coupling effects, reject step disturbances, and provide adequate damping to the subsystem interactions	checkmate other hybrid intelligent algorithms
Ikutegbe <i>et al.</i> (2015),	The steady-state design of CSTR and its implementation, using Matrix Laboratory (MATLAB) program	Continuously Stirred Tank Reactor (CSTR) was modeled and the reactor performance analysed with a view to coming up with an essential parametric quantity useful for assessing control at the steady-state design stage	reactor design for low conversion and maximum temperatures can present control problems with exothermic reactions
Vojtěšek and Dostál (2015)	Nonlinear Adaptive Control of Continuous Stirred-Tank Reactor	<ul style="list-style-type: none"> ▪ Program provides computation for various values of the input variables which made GUI universal for the wide range of the users. ▪ which can be used for investigating of the static and dynamic behaviour and adaptive control of the nonlinear system 	<ul style="list-style-type: none"> ▪ The work consider applications of MATLAB for control non linear variable. ▪ There was no theoretical knowledge on the modeling
Zhao <i>et al.</i> , (2015)	Terminal Sliding Mode Control for CSTR	<ul style="list-style-type: none"> ▪ CSTR operation are performed around certain Stable point to optimal output 	<ul style="list-style-type: none"> ▪ certain considerations where not made on inputs

		<ul style="list-style-type: none"> ▪ The work design process algorithm for system uncertain 	
Alshamari <i>et al.</i> , (2018)	Surveyed on issues related to modelling control strategies for CSTR	<ul style="list-style-type: none"> ▪ They examined a highly nonlinearity of plant containing numbers of stable and unstable operating point. ▪ The work is strongly concerned on linearization, observation and addressing the challenging of equilibrium points. 	<ul style="list-style-type: none"> ▪ In their results PID algorithms used did not guarantee consistent performance ▪ They feedback linearization was only to regulate operating point.
Strivastava, (2012)	Predictive Control Based strategy using NN and MPC a in CSTR	<ul style="list-style-type: none"> ▪ it was observed that NN predictive control algorithm produce better match for the dynamic system ▪ The simulation results obtain demonstrated more flexibility with use of NNMPC 	<ul style="list-style-type: none"> ▪ The parameters were optimally determined by quadratic performance index.
Yadav and Trivedi (2014)	Design of feedback Controller using PSO, PID and GA algorithms for Chemical Process	<ul style="list-style-type: none"> ▪ PSO-PID algorithm was used to control temperature ▪ The performance analysis was 	<ul style="list-style-type: none"> ▪ The work was possible based on the ISE to guide PSO in search of Controller parameter but was not with algorithms

		<p>done for controller design using PID, PID-GA and PID-PSO</p> <ul style="list-style-type: none"> ▪ The result obtained in their work shows that PSO-PID provide better performance 	to test for their effectiveness
Ahmed <i>et al.</i> (2016)	Designing a Temperature Control of the Continuous Stirred Tank reactor (CSTR) using three configurations	The three configurations (feedback, feed forward and cascade). For each type of the control systems the controllers used was tuned and its optimum parameters were calculated.	The PID was select as whole instead achieving the stability analysis of the Bode, Nyquist and root locus with P and PI

32.6 Object Oriented Programming

Object Oriented Methodology (OOM) is a system development approach encouraging and facilitating re-use of software components. With this methodology, a computer system can be developed on a component basis which enables the effective re-use of existing components and facilitates the sharing of its components by other systems.

Through the adoption of OOM, higher productivity, lower maintenance cost and better quality can be achieved. This methodology employs international standard of Unified Modelling Language (UML) from the Object Management Group (OMG). UML is a modelling standard for OO analysis and design which has been widely adopted in the Information Technology (IT) industry.

The considerations for adopting OOM in this work, given the characteristics of the methodology, OOM best suits for development of this work which exhibit any or all of the following characteristics:

- i. Projects of medium to large scale: The component-based development approach of OOM manages large-scaled applications by breaking down the complex solution into components. Furthermore, the modelling techniques of OOM are best used to model medium to large scaled applications, which usually involve complex business logic.
- ii. Departments/project teams with a planned series of developments within a similar project area.
- iii. One of the major benefits of adopting OOM is reuse. Application development is facilitated by the reuse of existing components which can greatly improve the productivity and facilitate rapid delivery.
- iv. Projects with development/implementation environment which provide adequate support for object technology the supporting features of the development/implementation environment, such as OO programming languages as discuss below in this work and appropriate development tools, are very important for realising the benefits of OOM.

2.6 Research Gap

Having studied various works done on industrial chemical process based on CSTR with the application of different control algorithms by previous authors, in this work, most related works have been reviewed and it is ideal to consider a process that was mostly and critically examined. It was observed that (Ram *et al.*, 2016) studied the same CSTR process for pH neutralization in wastewater treatment using different control algorithms. The author examined the response of the same CSTR process using conventional PID controller and obtained the

following better performance response for Branica Turning with 4.2 seconds rise time, 32.7 seconds settling time and overshoot of 32.21% . Also from the study of (Park *et al.*, 2021) bottleneck that support the existing studies on perturbation and discretized values that affects PID control performance in handling uncertainties parameters was as well identified.

In this work, studying the same CSTR process for pH- regulating Process control, a hybrid intelligent control model that combines Fuzzy Logic Control and PID which were created within the Fuzzy Inference System (FIS) frame works are proposed to improve response performance of the system in terms of rise time, settling time and percentage overshoot.

CHAPTER THREE

METHODOLOGY

3.1 Introduction

This section presents analysis of the algorithm used for most chemical Plants in the industries for pH neutralization it was observed that most Plants has process device known as Continuous Stirred Tank Reactor (CSTR) for most chemical systems. Some of the issues arising for this chemical process are due to low performance in system due to system respond time, setting up time, rise time and overshoots, these indices performances reflect in the output of the chemical process. This work is intended to develop hybridized system algorithms for industrial chemical for pH neutralization process which will address these challenges. In achieving this, there are different methodologies, which may include the Structured System Analysis and Design Methodology (SSADM), but this work adopted Object Oriented Methodology (OOM) and Simulation. The various simulations of the hybrid systems algorithms as applied in the pH neutralization plant was perform using Matlab Simulink R2014a and develop a software application that will provide interface using Netbeans Integrated Development Environment (IDE) version 8.0.1 and Operating System (64bits) for the implementation of the hybrid system algorithms using Object Oriented Methodology (OOM) approach. In Figure 3.1 the framework for the method to be used is described. First, the requirements for the chemical plant will be gathered including the control model requirement, then analysis on the existing model will be carried out. Details of the dynamic lodel will be reviewed, before carrying out simulation on the analysed models. The design of the system architecture will develop and the application will be programming using a higher level language program code.

Analysis Models

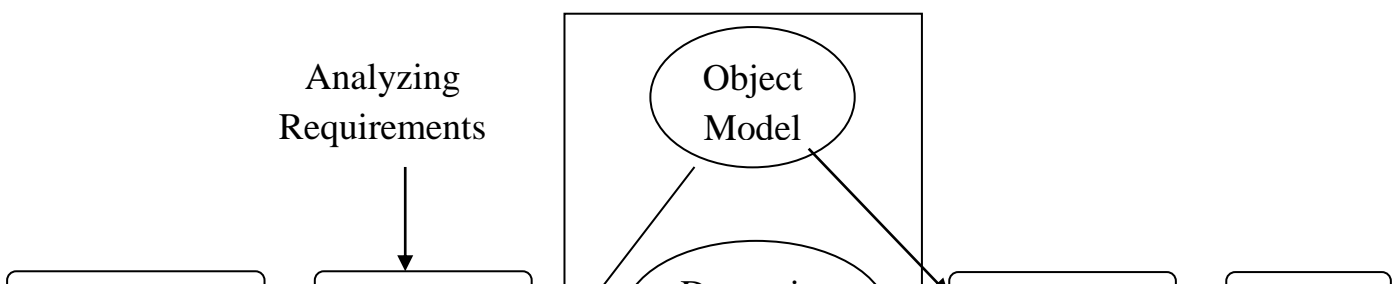


Figure 3.1: Object Oriented Methodology Framework.

Object oriented programming (OOP) this is the concept of objects which could contain data, in the form of fields which sometime called attributes. The codes of object oriented are in forms of procedure that can access and often modify the features of the objects that contain the data field. In OOP, computer programs are designed by making objects to interact with each other.

This work adopt object Oriented Programming approaches for developing the interactions and interfaces of the components of the pH neutralization Chemical Process in the proposed hybrid control algorithms. The simulations coefficients in the Matlab Simulink will be integrated into the OOP to produce the application for the system.

3.2 Analysis

System analysis is a process of investigating, analysis, design, installation and evaluation of information system either for change or modernization.

Comprehensive researches were done on the existing study, in order to identify the research gaps and challenges.

3.2.1 Analysis of the Existing Study

The analysis of the existing study is based on the mathematical Models describing the dynamics of a CSTR process and the model architecture.

3.2.5 Mathematical Model for the CSTR for pH Neutralization Process

The mathematical model of the considered CSTR is developed using energy balance. An exothermic irreversible reaction $A \rightarrow B$ takes place in the reactor, which is in turn cooled by a coolant that flows the jacket around the reactor. The feed material A is fed into the reactor. The Stirrer is used to blend the components of inputs to ensure uniformity of the reactant which is afterward extracted as the product (output) out of exit value. The jacket is taken to be perfectly mixed. Heat transfer takes place through the reactor wall into jacket. The main objective is to keep the temperature of the reacting mixture at desired value. The controlled or manipulated variable is the coolant temperature.

A diagram model of a pH neutralization process for wastewater treatment is shown in Figure 3.2. In this process, strong acid (HCl) and strong base (NaOH) of 1 molarity (Ram *et al.*, 2016) are used.

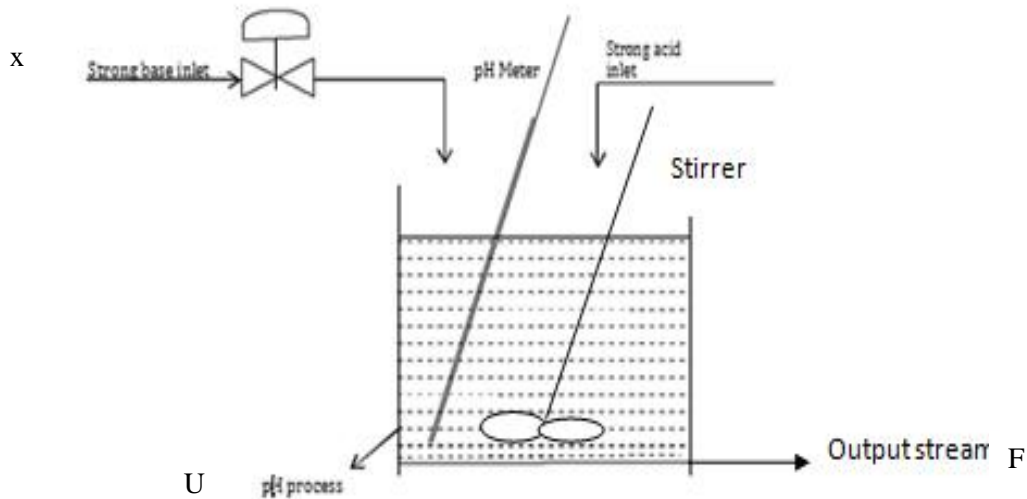


Figure 3.2: pH Process Model

Assuming no chemical reaction at given level, the equation for material balance can be expressed by:

$$\begin{aligned}
 &[\text{Rate of accumulation within vessel volume}] = [\text{Inflow rate to the pH process}] - [\text{Outflow rate from the pH process}] \\
 &[\text{Rate of accumulation within vessel volume}] = [\text{Inflow rate to the pH process}] - \\
 &[\text{Outflow rate from the pH process}] \quad (3.1)
 \end{aligned}$$

$$V \frac{dx}{dt} = U - Fx \quad (3.2)$$

where V is the volume of the mixture in the pH process, x is the state variable of the nonlinear pH process, U is the pH process input flow rate, F is the pH output flow rate. The back titration curve for process flow is given by:

$$T_{(\text{pH})} = X \quad (3.3)$$

In a chemical reaction, the components of the system for the back titration curve (TC) produce the nonlinearity of the process flow and can be represented by:

$$C_{\text{TC}} = \frac{A[\text{pH}] + \sum_{i=1}^n a_i [\text{pH}] C_i}{\sum_{i=1}^n a_i [\text{pH}] X_i} \quad (3.4)$$

where $a_i(\text{pH})$ is the acid-base weighting factor (-1 for Strong acid and + 1 for strong base), n is the number of ions present in the reactor, C_i ion concentration of the i^{th} kind of process flow, X_i is the concentration of the i^{th} kind of neutralization liquid.

The value of pH is defined as the negative logarithmic value of the concentration of Hydrogen $[\text{H}^+]$ ions.

$$\text{pH} = -\log(\text{H}^+) \quad (3.5)$$

For the neutralization process

$$A[\text{pH}] = 10[-\text{pH}_{\text{sv}}] - 10[\text{pH}_{\text{sv}} - 14] \quad (3.6)$$

where pH_{sv} is the setpoint (or desired) value

In the neutralization process, the difference value between the actual measured pH and the set point value in line with the nonlinear conversion is given by:

$$Y = (\text{pH}_{\text{sv}}) - (\text{pH}) \quad (3.7)$$

The First order differential equation is given by Equation (3.2) and since X is equal to $T(\text{pH})$, substituting into Equation (3.2) gives:

$$V \frac{d[\text{pH}]}{dt} = U - F[\text{pH}] \quad (3.8)$$

Taking the Laplace transform of Equation (3.8) assuming the initial condition to be zero, the Laplace transform is to reduce the differential equation into linearization and rearranging gives:

$$V\text{spH}(s) + F\text{pH}(s) = U(s) \quad (3.9)$$

Equations (3.1) – (3.9) are standard equations that describes pH process Model

Further rearrangement of Equation (3.9) gives:

$$\frac{pH(s)}{U(s)} = \frac{\frac{1}{F}}{\frac{V}{F}s + 1} \quad (3.10)$$

where V/F is equal to the process time constant τ and $1/F$ is the process gain, K .

The pH process is a first order system with time delay due to pipe line and detection process for the measuring instrument (sensor). Hence it takes the general transfer function model of First Order Plus Time Delay (FOPTD) given by:

$$G(s) = \frac{Ke^{-T_D s}}{\tau s + 1} \quad (3.11)$$

where $e^{-T_D s}$ is the time delay.

Substituting parameters obtained from real time experiments conducted in Dinesh and Deepika (2014) which was applied in (Ram *et al.*, 2016), with $K = 0.276$, $\tau = 3.2$, and $T_D = 5.005$, Equation (3.12) becomes:

$$G(s) = \frac{0.276e^{-5.005s}}{3.2s + 1} \quad (3.12)$$

3.2.6 State Space Equation

State space model is a linear time invariant (LTI) representation of a dynamic system in either continuous or discrete time. Expressing a model into state space form is the basis for many techniques in process dynamics and analysis of control systems. The general form of state space model is given by:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \quad (3.13)$$

such that states x and state derivative \dot{x} are in the set of a real number vectors of length n defined as $x \in \mathbb{R}_n$ and $\dot{x} \in \mathbb{R}_n$. Also the other variables, output y and input, u are defined as $y \in \mathbb{R}_p$ and $u \in \mathbb{R}_m$. where A is the state transition matrix, B is the input matrix, C is the output matrix, and D is the perturbation matrix. However, the matrix D is typically zero, the expression m , n , and p represents the dimensions of each matrix: m input, n states, and p output.

The transfer function in Equation (3.12) can be further expressed to describe the internal variable (which in this case is the concentration) of the pH process using state-space form. The state space expression is given by:

$$\begin{aligned}\dot{x} &= [-0.3125]x + [0.25]u \\ y &= [0.345]x + [0]u\end{aligned}\tag{3.14}$$

The MATLAB programme of the State space transformation from Laplace transform is given by:

```
num =[0.276];
>> den = [3.2 1];
>> sys =tf(num,den)
sys =
    0.276
-----
    3.2 s + 1
Continuous-time transfer function.
>> sys_ss = ss(sys)
sys_ss =
    a = x1
    x1 -0.3125
    b = u1
    x1 0.25
    c = x1
    y1 0.345
    d = u1
    y1 0
```

The representation describes further analysis of the existing transfer function in Matlab command window; fuzzy intelligent algorithm is required to track the continuous discrete values of the transfer function in Chemical Plant.

3.2.4 Study of the Existing System

In order to explore our research areas, an existing Continuous Stirred Tank Reactor (CSTR) systems with the application of control algorithms of an industrial chemical process, for treatment of waste water that have been studied so as to determine the weaknesses and strength of the works and then make sufficient contributions in to improve the general performance of the system in terms of rise time, settling time and percentage overshoot. The previous work done by (Ram *et al.* 2016) has been chosen for study and improvement.

3.3 Architecture of Existing Study

The existing system structure is shown in Figure 3.3. The architecture represents a continuous time control system that uses a PID algorithm to improve the response performance of the system. The structure shows a feedback continuous time PID control algorithm in the forward path with pH neutralization process (plant).

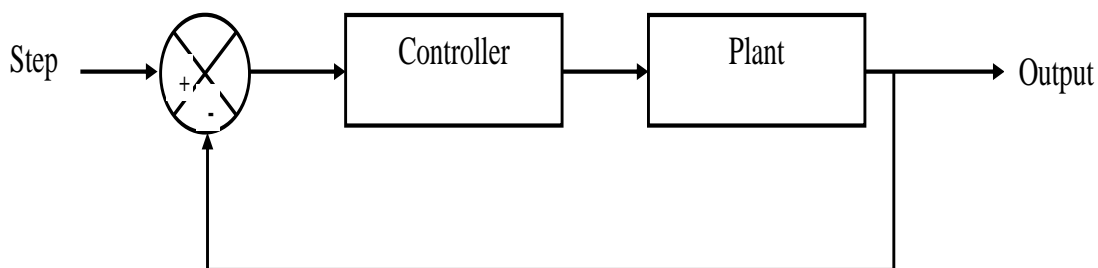


Figure 3.3: Existing System Architecture (Ram *et al.*, 2016).

The continuous time PID controlled pH neutralization control process in the Plant offers improved performance in terms of robustness. However, the problem with a PID algorithm is that its performance is seriously affected by perturbation and nonlinearity effect associated with industrial processes. For this reason, intelligent system are increase being proposed and demanded in industries because of their ability to overcome the effect of nonlinearity.

3.3.1 Challenges of the Existing Study

The existing studies has showed that PID controllers performances are affected by nonlinearity and disurbance or perturbation as indentified by (Ram *et al.*, 2016). For this reason, intelligent fuzzy model is being proposed to addressed this short coming of PID in industrial chemical plant design. The integration of Fuzzy and PID will help to overcome the effect of nonlinearity of the sytem with some set rules. There are challenges of low performance of the system due to the presences of multi variable in the complex dynamic system functionality, the Fuzzy train data set to study and handle the multiple variable.

3.4 Proposed System Architecture

The proposed control algorithm for an industrial chemical process which is a pH neutralization process in CSTR is given by the combination of fuzzy algorithm, PID algorithm and Fuzzy PID (FPID) algorithm. Initially, the parameters (or gains) of the PID control algorithm are obtained by the means of tuning using error signal. The gains of PID are used to tune fuzzy PID. Figures 3.4 shows the proposed system architecture of the proposed system.

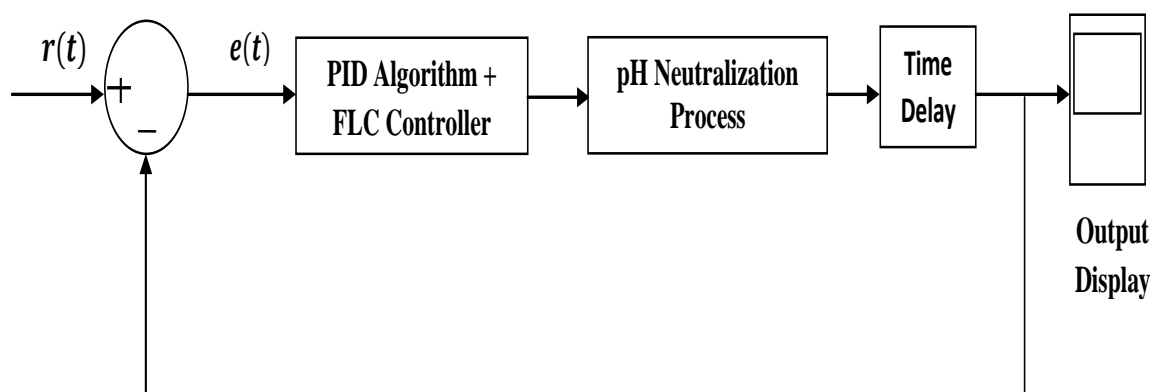


Figure 3.4: Proposed System Architecture

Initially, the parameters (or gains) of the PID control algorithm are obtained by the means of tuning using error signal. The gains of PID are used to tune fuzzy PID. The output and input data of fuzzy PID and PID is used to train the Fuzzy inference system (FIS) that is used to tune the intelligent control algorithm system. The Proposed architecture will use fuzzy inference system to regulate the

complexity of the system, nonlinearity of the system, and the time varying dynamic processes of industrial control loop. It can also be used as online gain tanner for developing the fuzzy PID

3.4.1 Proportional Integral and Derivative (PID) Algorithm

A Proportional Integral Derivative (PID) algorithm has been well implemented in almost every industrial chemical process due to its simplicity of design and robustness. This algorithm is largely common because it is able to provide good response characteristic for a closed loop system.

A PID controller offers a proportional action to control error, an integral action to control error and a derivative action to control error. Table 3.1 summarizes the functions of the three elements that make up a PID controller. Figure 3.5 is block diagram representation of a PID control algorithm.

Table 3.1: Summary of the Elements of PID Controller.

Control Action	Symbol	Function
Proportional Control	P	It implements the typical operation of increasing the control variable when the control error is large
Integral Control	I	It momentarily tracks the control error. It does this by relating to the past values of the control error and allows the reduction to zero of the steady state error when desired step input is applied
Derivative Control	D	It predicts the future values of the control error and has a great potential in improving the control performance as it can anticipate an incorrect trend of the control error and counteract for it.

Figure 3.5 shows how the three control algorithms are integrated as one called PID Controller.

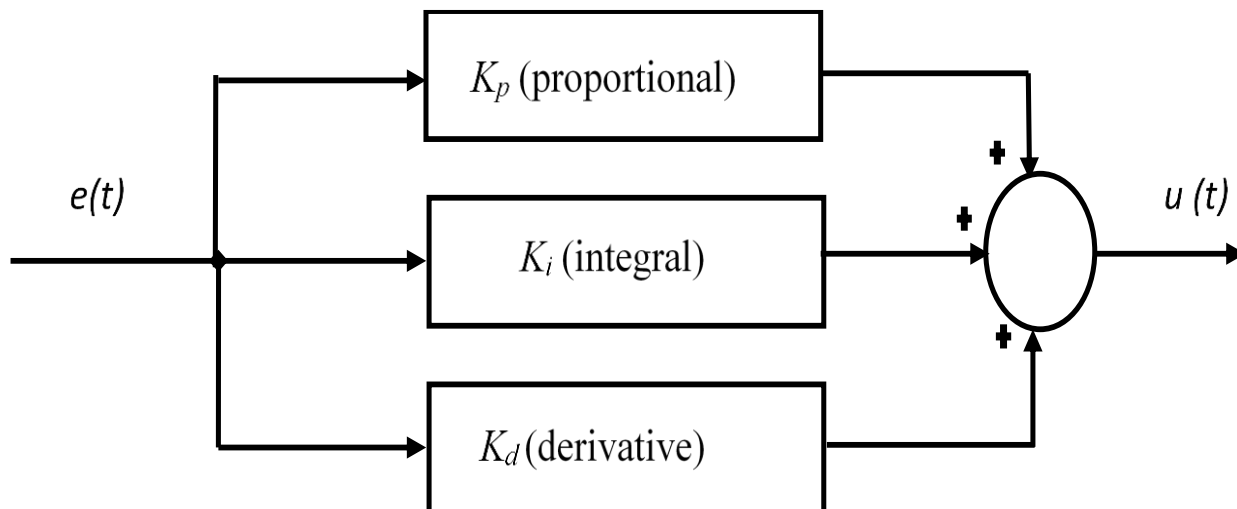


Figure 3.5: Block diagram of PID controller.

The Existing PID Algorithm

$$u(t) = k_p e(t) + k_i \int e(t) dt + k_d \frac{de(t)}{dt} \quad (3.15)$$

Or in Laplace transform as:

$$C(s) = \frac{U(s)}{E(s)} = k_p + k_i \frac{1}{s} + k_d s \quad (3.16)$$

In this work, a PID algorithm with low pass filter is developed and is given by:

$$C(s) = k_p + K_i \frac{1}{s} + K_d s \times \frac{1}{Tfs + 1} \quad (3.17)$$

3.4.2 Analysis of the Proposed System

In order to design the PID control algorithm for the proposed system, the problem of non linearity of the PID controller was first analysed in MATLAB, the following steps are used to address this problem.:

- i. The CSTR transfer function is generated in MATLAB command window as presented by the following command:

```
>> num = [0.276];
>> den = [3.2 1];
>> Gp =tf(num,den)
```

Gp =

$$\frac{0.276}{3.2 s + 1}$$

Continuous-time transfer function.

Where num means numerator, den means denominator, tf(num,den) means transfer function of numerator and denominator represented by Gp.

i. A PID tuner is opened as shown in Figure 3.6:

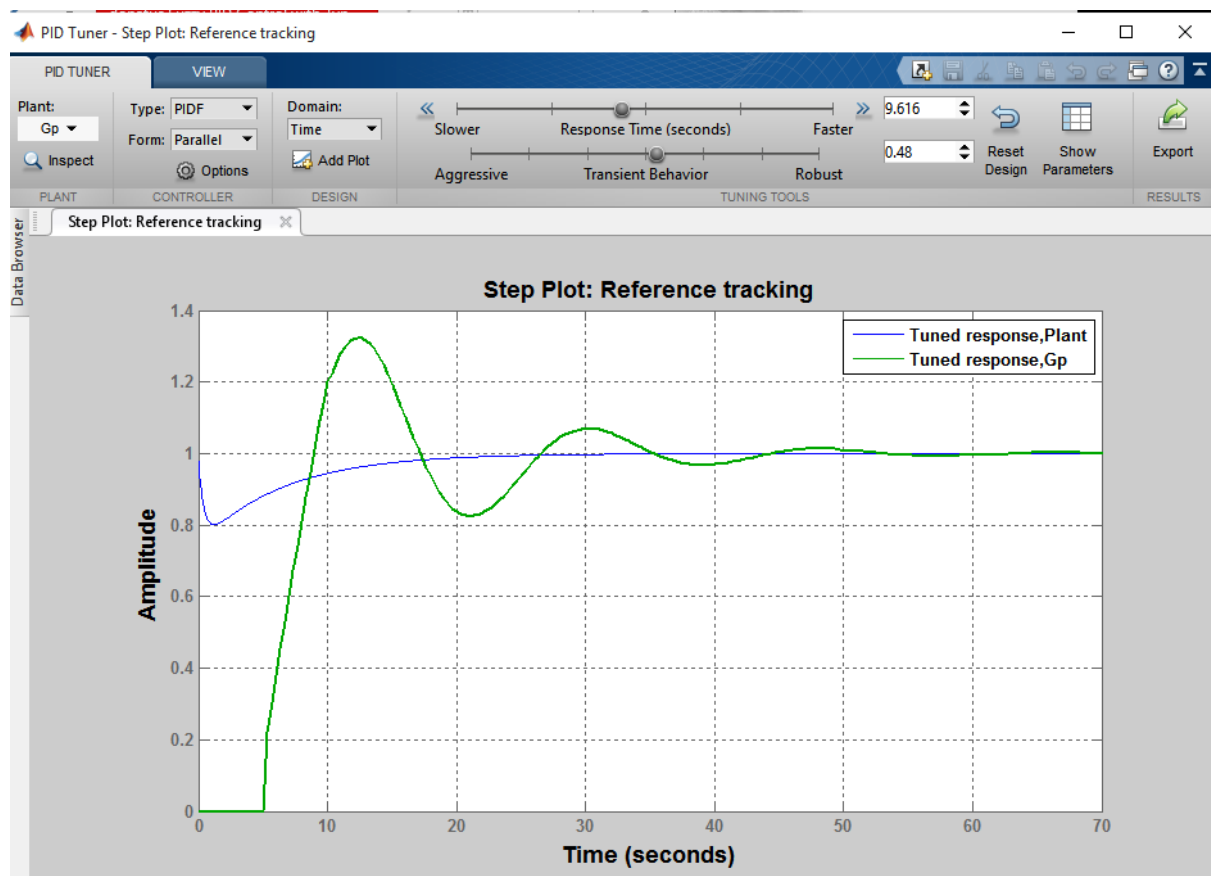


Figure 3.6: MATLAB PID GUI Tuner.

The transfer function Gp known as the plant is imported and then faster and robust tuning is carried out to obtain the appropriate gains $k_p = 3.46$, $k_i = 0.661$, and $k_d = 1.75$. The designed PID compensator with low pass filter, C is exported to the MATLAB command window as presented below:

```
>> C
C =
      Kp + Ki *  $\frac{1}{s}$  + Kd *  $\frac{s}{Tf*s+1}$ 
```

with $K_p = 3.46$, $K_i = 0.661$, $K_d = 1.75$, $T_f = 0.0408$

Continuous-time PIDF controller in parallel form.

Hence, the continuous-time conventional PID control algorithm in parallel form designed in this work is given by:

$$C(s) = 3.46 + \frac{0.661}{s} + \frac{1.75s}{0.0408s + 1} \quad (3.18)$$

The Simulink model of pH process compensated with PID algorithm is shown in Figure 3.7. The MATLAB programme created using the extension file (m-file) is given below:

This Matlab code simulates a PID controller for a first order time delay system
 % $G(s) = (K * e^{-\theta * s}) / (T * s + 1)$

```
clear all;
clc;
```

```
T = 3.2; K = 0.276; theta = 5.005; % System Parameters
```

```
refe = 1; % Reference to follow
```

```
Tf = 1000; % Simulation time
```

```
yold = 0; yold1 = 0;
```

```
yp = []; ys_prime = [];
```

```
er = refe; % Error (Initial error = Reference)
```

```
er1 = 0; % First derivative of error
```

```
er2 = 0; % Second derivative of error
```

```
eold = refe; eold2 = 0;
```

```
dt = 1;
```

```
for i=1:dt:Tfdtspan = [ii+dt];
```

```
    eold2 = eold;
```

```
    eold = er;
```

```
    er = refe - yold;
```

$$er2 = er + eold2 - 2*eold;$$

$$er1 = er - eold;$$

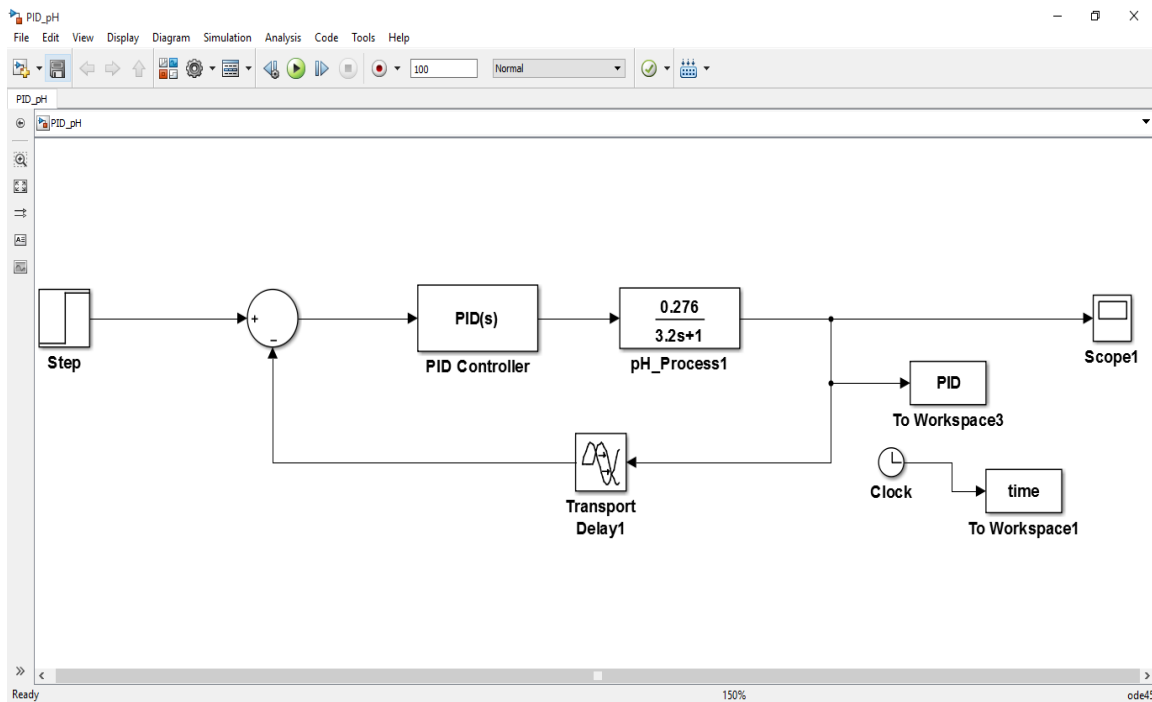


Figure 3.7 Simulink model of pH process with PID compensator.

In figure 3.7 is the block diagram of pH process modeled in Simulink, which is the model of PID controlled pH process. The first input to the summer or summing point (that is the block with +/- sign) is the set pH value (represented by the step block) and it is the desired or set pH value at which the pH process is expected to run to or achieved. The second input to the summer is the feedback signal, which is the current pH value at the output of the pH process which is measured or captured by a feedback sensor (in this case, for simulation purpose, the feedback sensor gain is taken as 1). Since the mathematical of pH process in Equation (3.11) showed that it is prone to time delay, a transport delay block is used in the Simulink model to represent the effect of delay on the pH process. It

can be seen in the block diagram that the set pH value entering the summer is positive (+) while the feedback signal coming from the output impacted by delay is negative. The difference between these two inputs is outputted by the summer and it is called error (that is, it represents the deviation of the output pH value from the expected or desired value). For instance, if the set pH value is 1 and the current or output pH value is 0.8, the error or deviation ($1 - 0.8$) is 0.2. This error is given to the PID controller that reads and manipulates the error signal and produces appropriate controller output called control input or command signal. The control input then reaches the pH process and caused it to adjust to the control input command. This way the pH process produces output pH value or response in accordance to the command signal from the controller. This process continues until the desired pH value is reached. As the pH value is reached, the error signal reduces to zero thereby stopping the pH control process provided the pH process continues to run at the expected value.

3.5 Fuzzy Logic Control Design

The steps highlighted below followed for developing of the Fuzzy Logic Control (FLC)

- i. The objectives and specifications are defined. This involves determine what to control, what to do to control the system and the kind of output (or response). Expected.

- ii. The input and output relationships are established and a minimum number of variables selected for input to the fuzzy logic system (for example, error and time derivative of error).
- iii. The problem to be controlled is broken down into a series of IF and THEN using the Fuzzy Logic rule-based structure. These rules defined the desired output response of the system for a given input condition.
- iv. The member functions of the fuzzy logic are created which help to define the values of input-output terms used in the rules.
- v. Simulation of the system is conducted after the construction of the fuzzy logic system.
- vi. The system is tested, the results evaluated, the rules and membership functions tuned and re-tested until satisfactory results were obtained.

A typical fuzzy logic control system has at least two inputs and output. Fuzzy inference System (FIS) maps inputs-outputs using the member functions. A membership function (MF) is a shape that defines how each point in the input space is mapped to a membership values (or degree of membership) between 0 and 1. In this design, the input and output is defined as follows: since Fuzzy logic control system needs at least two inputs and output, the first input will be the error (E) while the second input is the change in the error $(\Delta E(s))$ and the control output U(s) of the fuzzy output (Y(s)). The equation establishing these are expressed given by:

$$E(s) = R(s) - Y(s) \tag{3.19}$$

$$\Delta E(s) = sE(s) - E(0) \tag{3.20}$$

Assuming zero initial condition and substituting Equation (3.19) into Equation (3.20) gives:

$$\Delta E(s) = s(R(s) - Y(s)) \quad (3.21)$$

3.5.6 Fuzzy Logic Intelligent Framework

Having identified the problem of uncertainty among variables possessed on PID controller in most Industrial chemical plant design, the fuzzy logic is an intelligent technique that makes use of human expertise to monitor and control engineering and industrial processes. Fuzzy logic involves simple rule based that takes the form: “IF A and B then C” approach. Figure 3.8 shows the elements of a fuzzy logic system.

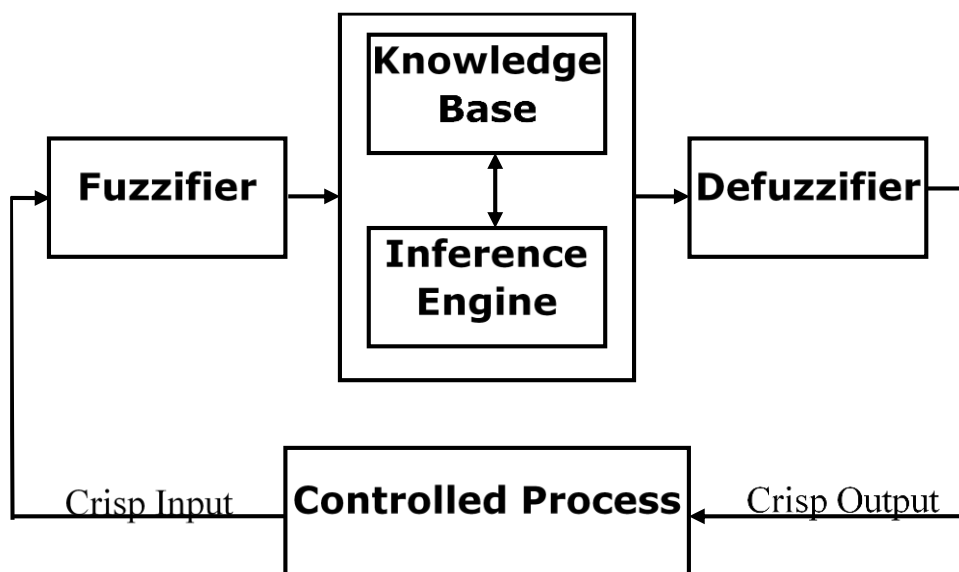


Figure 3.8: Elements of a Fuzzy Logic Control Algorithm.

The specific elements (or components) of a fuzzy logic controller consists of Fuzzifier, Defuzzifier, Knowledge base and Inference Engine (Decision making unit). The representation of the parameters involved in fuzzy logic is known as linguistic variables.

i. Fuzzification

Fuzzification is the process of transforming a crisp value into a fuzzy set, so that it can be processed by fuzzy inference mechanism. The inputs and output of the design as specified in table 3.2 are assigned linguistic variables and some degree of membership. The linguistic variables used in this research include: NB = Negative Large, NM = Negative Medium, NS = Negative Small, ZE = Zero

Error, PS = Positive Small, PM = Positive Medium and PL = Positive Large. Then the fuzzy rules in contain table 3.2 are formulated using a series of IF-THEN Statement, combined with AND/OR operator.

ii. Defuzzification

The transformation from a fuzzy set to a crisp value is known as defuzzification. Then the Crisp Output is obtain from the $\{\text{Sum}(\text{Membership Degree} * \text{Singleton Position})\} / \text{Membership degree}$. The defuzzification strategy is aimed at producing a non fuzzy control action that best represents the possibility of an inferred fuzzy control action.

a) Fuzzifier

The inputs real-life data or crisp measurement from linguistic variables the Fuzzifier converts the crisp input to degrees of membership using membership functions. This process is called fuzzification. The fuzzification block matches the input data with the conditions of the rule to determine.

- i. The first step in fuzzification is determined by inputs and outputs which has two inputs (e, \dot{e}) and one output (\dot{y}). The inputs are error (e) which measures the difference between desired and actual output position, and the change of error (\dot{e}) which measures the difference between desired and actual velocity and output is fuzzy equivalent torque.
- ii. The second step is chosen an appropriate membership function for inputs and output which, to simplicity in implementation because it is a linear function with regard to acceptable performance triangular membership function is selected.
- iii. The third step is chosen the correct labels for each fuzzy set which, in this research namely as linguistic variable. Based on experience knowledge the linguistic variables for error (e) are; Negative Big (NB), Negative Medium

(NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), Positive Big (PB), and based on.

b) Defuzzifier

The process by which the defuzzifier converts linguistic variables into crisp values that can be understood by controlled system is known as defuzzification this achieved in the fuzzy set crisps sets.

c) Knowledge Base

Knowledge base consists of a database and the rule base is used for defining set of rules. The collection of rules in a fuzzy logic system is called a knowledge base. The rules consist of two “IF” sides and the “THEN” side is known as the conclusion. Using more number of rules is recommended for achieving a very accurate system. Table 3.2 presents a fuzzy membership rule matrix table with 9 members.

d) Inference Engine

Decision making unit is responsible for taking decision based on given inputs and set of rules.

The fuzzy set and crisps sets

It has been observed that the main objective behind fuzzy logic is to represent and reason with some particular form knowledge expressed in linguistic form. Again, in a pH regulating System, whose pH values ranges from 0-14, where 0- 6.9 are acidic and pH value of 7 is said to be neutral and 7.1-14 values are refereed to be basic in a pH Scale. Hence using a language oriented approach for the knowledge representation, we can built a conceptual framework of the pH regulating system using the pH values.

Traditional crisps sets are based on two-value logic: Object is either members or not members. Every individual membership values μ may be either 1 or 0 that

discriminates between members and non members of the crisps set. Supposing a pH regulating Process has pH values of 0-14, the possible fuzzy set crisp values would be acidic, strong, medium and weak or Neural and also basic (Strong, Medium and Weak) is obtained by:

$$\mu_{\text{Acidic, (Strong, Medium, Low)}} \begin{cases} 0 & \text{if } \text{pH}_{\text{Acidic}} \leq \text{pH}_{\text{strong}} \leq \text{pH}_{\text{set value of Acid pH}} \\ \frac{\text{pH}_A - \text{pH}_s}{10} & \text{if } \text{pH}_{\text{acidic}} \leq \text{pH}_{\text{Medium}} \leq \text{pH}_{\text{Set value of Acid}} \\ 10 & \text{if } \text{pH}_{\text{Medium}} \leq \text{pH}_{\text{low}} \leq \text{pH}_{\text{set value of Acid}} \\ 1 & \end{cases}$$

However, the membership values are based on triangular membership functions.

In a pH of 5 in an acidic solution, the partial membership value has the crisp value of the degree as:

$$\mu_{\text{strong}} = \frac{6.90 - 5.00}{10} = \frac{1.90}{10} = 0.19$$

$$\mu_{\text{medium}} = \frac{5.00 - 3.45}{10} = \frac{1.55}{10} = 0.155$$

$$\mu_{\text{Low}} = 1 - (0.19 + 0.155) = 1 - 0.345 = 0.655$$

Therefore, the pH₅ have the crisps values of $\mu_{\text{strong}} = 0.19$, $\mu_{\text{Medium}} = 0.155$ and μ_{Low} as = 0.655

The same also apply to Basic solution pH regulating Value ranges from 7.1- 14 the crisp values

$$\mu_{\text{Basic, (Strong, Medium, Low)}} \begin{cases} 0 & \text{if } \text{pH}_{\text{Basic}} \leq \text{pH}_{\text{strong}} \leq \text{pH}_{\text{set value of Basic pH}} \\ \frac{\text{pH}_B - \text{pH}_s}{10} & \text{if } \text{pH}_{\text{Baisic}} \leq \text{pH}_{\text{Medium}} \leq \text{pH}_{\text{Set value of Basic}} \\ 10 & \text{if } \text{pH}_{\text{Medium}} \leq \text{pH}_{\text{low}} \leq \text{pH}_{\text{set value of Basic}} \\ 1 & \end{cases}$$

In a pH of 12 in a Basic solution, the partial membership value has the crisp value of the degree as:

$$\mu_{\text{strong}} = \frac{14 - 12}{10} = \frac{2.00}{10} = 0.2$$

$$\mu_{medium} = \frac{12.0 - 7.1}{10} = \frac{4.90}{10} = 0.49$$

$$\mu_{Low} = 1 - (0.20 + 0.49) = 1 - 0.69 = 0.31$$

Therefore, the pH₁₂ have the crisps values of $\mu_{strong} = 0.20$, $\mu_{Medium} = 0.49$ and μ_{Low} as = 0.31

Finally, in Neural Solution

$$\mu_{Neural} = \begin{cases} 0 & \text{if } pH_{Neural} = pH_{Neural} = pH_{set \text{ value of Neural } pH} \\ \frac{pH_N - pH_N}{10} & \text{if } pH_{Neural} = pH_{Neural} \leq pH_{Set \text{ value of Neural}} \\ 10 & \text{if } pH_{Neural} \leq pH_{Neural} \leq pH_{set \text{ value of Neural}} \\ 0 & \end{cases}$$

In a pH of 7 in a Neural solution, the partial membership value has the crisp value of the degree as:

$$\mu_{Neural} = \frac{7 - 7}{10} = \frac{0}{10} = 0.00$$

3.5.2 Fuzzy Inference System

As mention earlier, the intelligent is based on Fuzzy model. Fuzzy logic Toolbox software supports two types of Fuzzy Inference system which are Mamdani and Sugeno fuzzy Inference System. This work adopts Mamdani FIS for building the fuzzy rules for all membership functions for all uncertainties.

3.5.2.1 Mamdani Fuzzy Inference System

Mamdani fuzzy Inference System was first introduced as a method to create a control system by synthesizing a set of linguistic control rule obtained from experienced human operators. These output fuzzy sets are combined into a single fuzzy set using the aggregation method of the Fuzzy Inference System (FIS). Then, to compute a final crisp output value, the combined out fuzzy set is defuzzified using one of the method. The reason while Mamdani Model in fuzzy system is used because each individual rule in a Mamdani System is intended to

shape the final function somewhat locally, i.e. individual rule modify the input-output space without necessarily altering the relation on other region. Mamdani system can look particularly appealing because they are designed to incorporate expert knowledge in the form of IF-THEN rules expressed in natural language. This forms the attractive feature for modelling and simulation of social and complex system. The Mamdani system provides a function that complies with its generating set of IF- THEN rules, which is a different from finding the relation of the rules. The inference process of a Mamdani System is described in the Fuzzy inference Process.

The advantage of Mamdani system is that it is intuitive, well- suited to human input; more interpretable rule base and have a widespread acceptance.

3.5.2.2 Sugeno Fuzzy Inference Systems

Sugeno fuzzy inference also referred to as Takagi-Sugeno-Kang fuzzy inference. It uses a singleton output membership functions that are either constant or a linear function of the value of the input values. The defuzzification process for a Sugeno System is more computationally efficient compare to that of Mamdani system, since it uses a weighted average or weighted sum of a few data points dimensional area. Some of the advantages of the Sugeno system is that it is computationally efficient; work well with linear techniques such as PID Controller; it is also work well with optimization and adaptive techniques; it guarantee output of surface continuity and it well- suited to mathematical analysis.

3.5.3 Flow Diagram of Fuzzy Logic Design

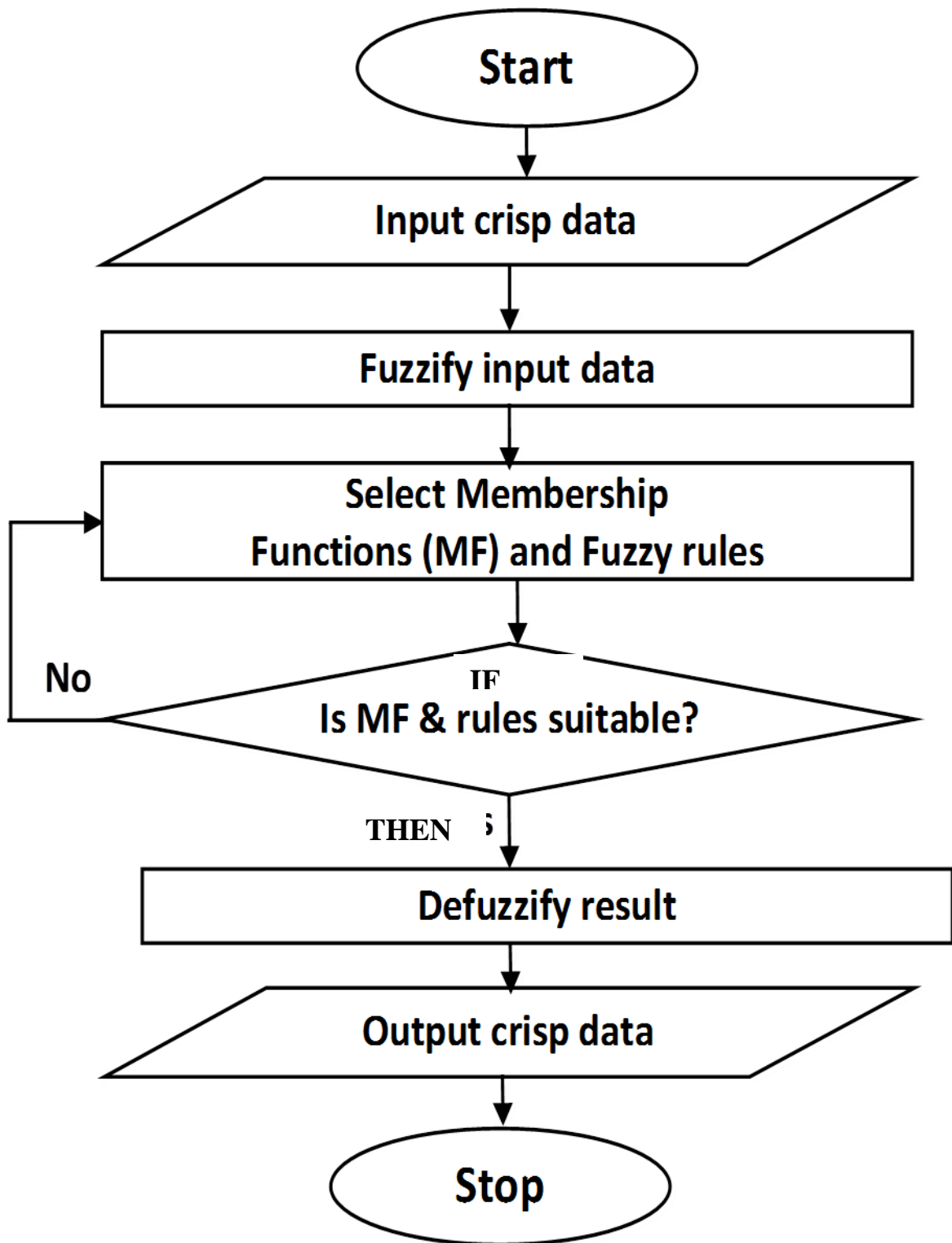


Figure 3.9: Flow Diagram of Fuzzy Logic Control Algorithm.

Figure 3.9 defines the fuzzy logic data flow. First the data input will be crisp as input, the fuzzier accept the crisped data input, where the crisp input to degrees of membership using membership functions notation by assigning appropriate

membership function with fuzzy rule. The fuzzy rules is being applied to all Membership Function (MF) the appropriate suitable MF will convert the linguistic variable into an understanding controlled value for the system. Finally, the output crisp data produces the results.

3.5.4 Advantages of Fuzzy Logic Control Algorithm

Fuzzy logic is not the only way to reason with ambiguous concepts but it seems to be the most apt to function approximation in control engineering. From the previous considerations, some of the most important advantages the use of fuzzy logic can entail to control system design are here detailed:

- i. Flexible, intuitive knowledge base design. Control and supervision speak the same language.
- ii. Convenient user interface
- iii. Easy computation. Widely available toolboxes and dedicated integrated circuits.
- iv. Learning. Linear in parameter systems (in most cases) make possible least squares, dead-zone learning algorithms and other results from adaptive control.
- v. Validation. Consistency, redundancy and completeness can be checked in rule bases (knowledge acquisition supervision). That could speed up automated learning and improve user interpretability.
- vi. Ambiguousness. Fuzzy logic is a natural way of expressing uncertain information. Research must be done in reasoning with incompleteness, i.e., concluding different actions depending on the possibility or necessity of certain plant situations. Some tools for it are already available.

- vii. Combine regulation algorithms and logic reasoning, allowing for integrated control schemas.
- viii. Its conceptual model (granulation, soft commutation) has been used in many other paradigms
- ix. FLC can incorporate a conventional design (PID, state feedback) and fine-tune it to certain plant nonlinearities due to universal approximation capacities.

3.5.5 Simulink Fuzzy Logic Tool

In this work, the designed fuzzy logic control algorithm is achieved using Simulink of MATLAB software. Simulink is a graphical user interface tool of MATLAB. It is used for modeling and simulation of mathematical models. Models can be built with Simulink blocks. The block for the design of the FLC algorithm is available at the directory in Simulink as shown in Figure 3.10. The Simulink model of fuzzy logic compensated pH process is presented in Figure 3.10.

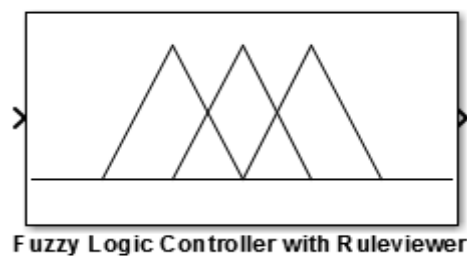


Figure 3.10 Fuzzy Logic Block Simulink

In designing a fuzzy logic control algorithm, a fuzzy inference system (FIS) of fuzzy logic tool is used. For the building, editing and monitoring fuzzy systems five basic tools representing the algorithm are used as shown in Figure 3.11 and outline as follows:

- a) Editor to determine the fuzzy inferential system FIS (called FIS editor)
- b) Member function (MF) editor
- c) Rule editor

- d) Rule viewer and
- e) Surface viewer

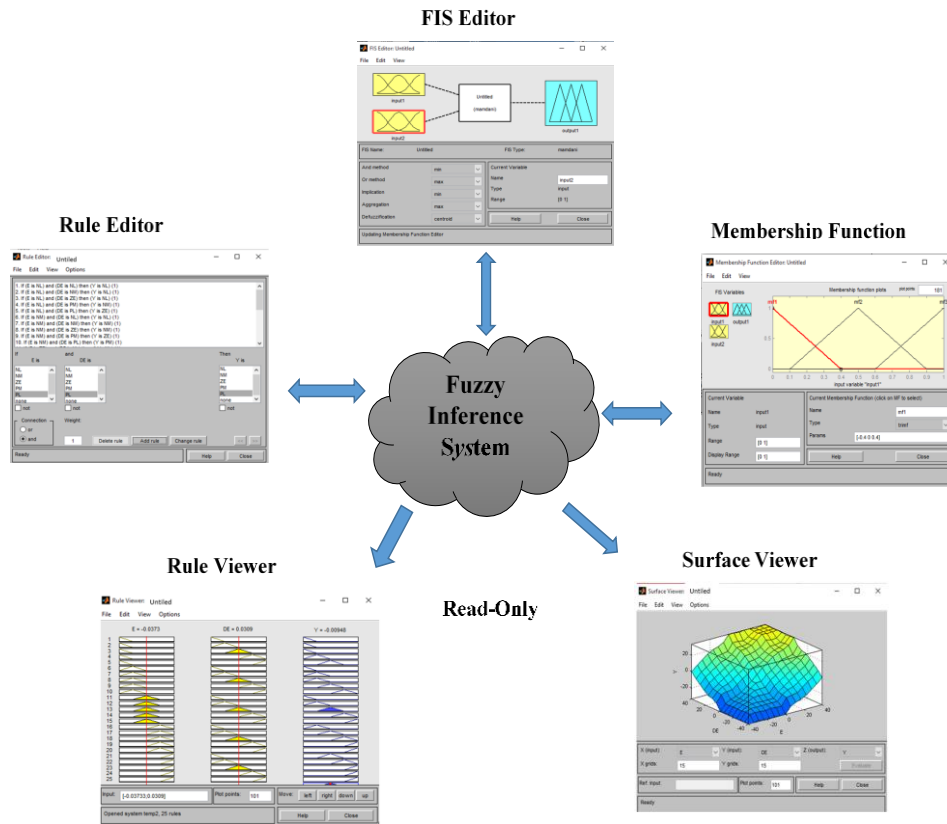


Figure 3.11 The Basic Tools in the FIS

3.6 Design of Fuzzy-PID Control Algorithm

Designing a fuzzy-PID control algorithm involves:

- i. Design a conventional PID control algorithm
- ii. Design an equivalent fuzzy PID control algorithm:
 - a. Use Mamdani style fuzzy FIS.
 - b. Use algebraic product for AND connective.
 - c. Normalize the range of both inputs (error, E and change in error, ΔE) to $[-1 \ 1]$.
 - d. The membership functions that make up the input sets are triangular and cross neighbour.
 - e. The output range is $[-1 \ 1]$.

- f. Use singletons as output, determined by the sum of the peak positions of the inputs
- g. Use the centre of gravity (COG) method for defuzzification.
- h. Simulate closed loop response in simulink

3.6.1 Integration Fuzzy Controller in pH Neutralization

The Fuzzy Logic Control (FLC) integrated in a pH industrial Neutralization process as presented in Figure 3.12. The design view of the fuzzy logic controller integrated with transfer function of the pH neutralization process as showned.

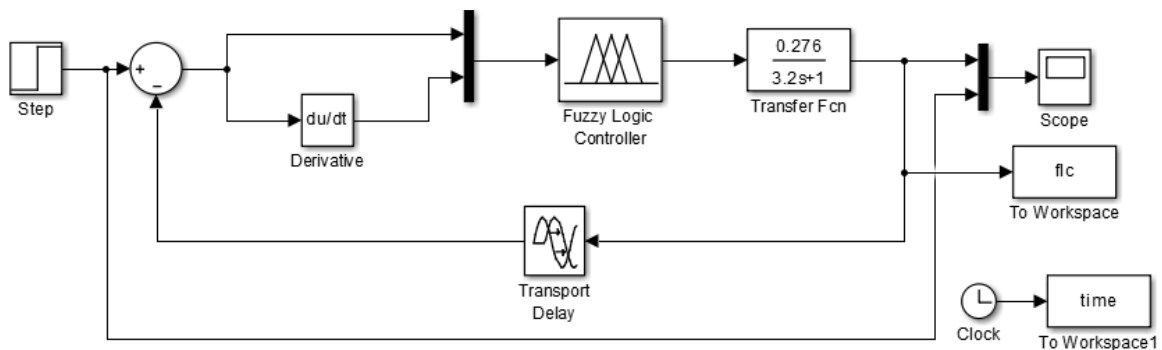


Figure 3.12: Fuzzy logic compensated pH process

The step input will accept signal from the Neutralization plant and then the fuzzy logic applies some sets of rules that will manipulate the step inputs to the transfer function using the transport or time delay to assess the input signals and the controlled or manipulated signals. In Figure 3.12, the inputs to the FLC are the error signal and the change or deviation in error signal with respect to time. These inputs are fed into the FLC which acts on them according to established set of rules and then produces a command signal that acts to regulate the pH process.

3.7 Detailed Architecture of the Proposed System

Eventually, the outputs of the fuzzy PID control algorithm and PID control algorithm are combined as hybrid intelligent fuzzy Inference System control algorithm. Figures 3.13 shows the detailed architecture of the proposed system high level model.

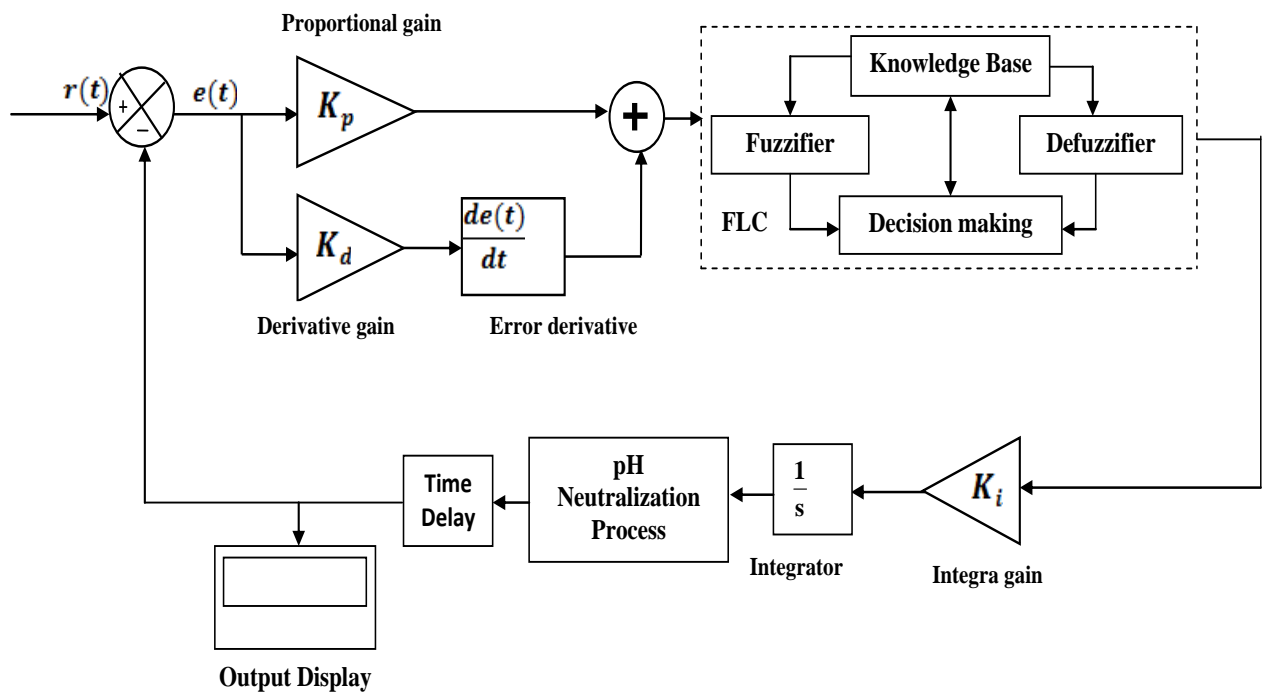


Figure 3.13: Detailed Architecture of the Proposed System.

Conventional PID controller does not give acceptable performance for systems with uncertain dynamics, time delays and non-linearity. Hence it is necessary to automatically tune the PID parameters so as to obtain satisfactory response. The PID gains are combined with the fuzzy logic. Based on expert knowledge a fuzzy logic system transforms a linguistic control strategy into an automatic control strategy.

The inputs to the controller are the error (e) and the rate of change of error (ΔE) while the outputs are controller gain K_p , K_i , and K_d . The structure of fuzzy-PID is a two input-one output structure as the range or the universe of discourse of the input and output membership functions the PID control algorithm has been tuned

using the MATLAB PID tuner graphical user interface (GUI) technique. From there the range of the input as well as output membership functions have been found. The linguistic variable levels are assigned as: negative big (NL), negative small (NS), zero (ZE), and positive small (PS) and positive big (PL). Similarly, the fuzzy set for error change (ΔE) is presented as NL, NS, ZE, PS, and PL. The ranges of these inputs are from -1 to 1 and -1 to 1 respectively.

3.7.1 Benefits of the Proposed System

The advantage offers by the proposed hybrid intelligent system are:

- i. It uses fuzzy inference system (FIS) to regulate the complexity of the system.
- ii. Minimized response time for pH Neutralization Process
- iii. Increase the level of precision of pH regulated values
- iv. The Fuzzy Inference System has the capability to manage nonlinearity of the system, and the time varying dynamic processes of industrial control loop.
- v. It can also be used as online gain tuner for developing the fuzzy PID

3.7.2 Design of the Proposed Fuzzy-PID Compensator

Conventional PID compensator does not give acceptable performance for systems with uncertain dynamics, time delays and non-linearity. Hence it is necessary to automatically tune the PID parameters for obtaining satisfactory response. The automatic tuning of PID compensator has been done using fuzzy logic. Based on expert knowledge a fuzzy logic system transforms a linguistic control strategy into an automatic control strategy.

In the Fuzzy-PID, the output membership functions of the fuzzy sets are obtained using the scaling of range corresponding to the formula given by (Chopra *et al*, 2014). For the output fuzzy sets the scaling of range has been done corresponding to the formulas given below:

$$K_{P'} = \frac{K_p - K_{P \min}}{K_{P \max} - K_{P \min}} \quad (3.22)$$

$$K_{P''} = \frac{K_i - K_{i \min}}{K_{i \max} - K_{i \min}} \quad (3.23)$$

$$K_d = \frac{K_d - K_{d \min}}{K_{d \max} - K_{d \min}} \quad (3.24)$$

The minimum and the maximum values of various gains have been obtained by analyzing the step response of the given process using Zeigler Nichols method (Chopra *et al.*, 2014). Equation (3.22) – (3.24) defines the various limits for scaling range of the PID gains into fuzzy sets out.

The inputs to the compensator are the error (e) and the rate of change of error (ΔE) while the outputs are controller gain K_p , K_i and K_d . The structure of fuzzy-PID controller is a two input-three output structure as shown in Figure 3.14, 3.15, 3.16, and 3.17 respectively. For finding the range or the universe of discourse of the input and output membership functions the PID compensator has been tuned using the conventional Zeigler Nichols method. From there the range of the input as well as output membership functions have been found. The linguistic variable levels are assigned as: Negative (NE), zero (ZE), and Positive (PO). Similarly, the fuzzy set for error change (DE) is presented as NE, ZE, and PO. The ranges of these inputs are from -1 to 1 and -1 to 1 respectively.

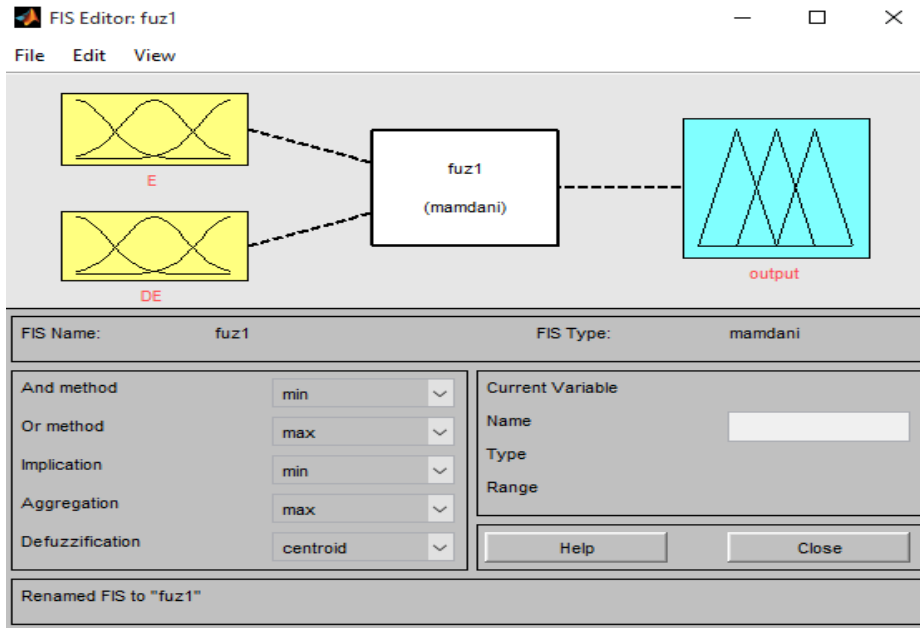


Figure 3.14 CSTR Fuzzy Inference System (FIS).

The error input on the fuzzy inference System is shown in Figure 3.24

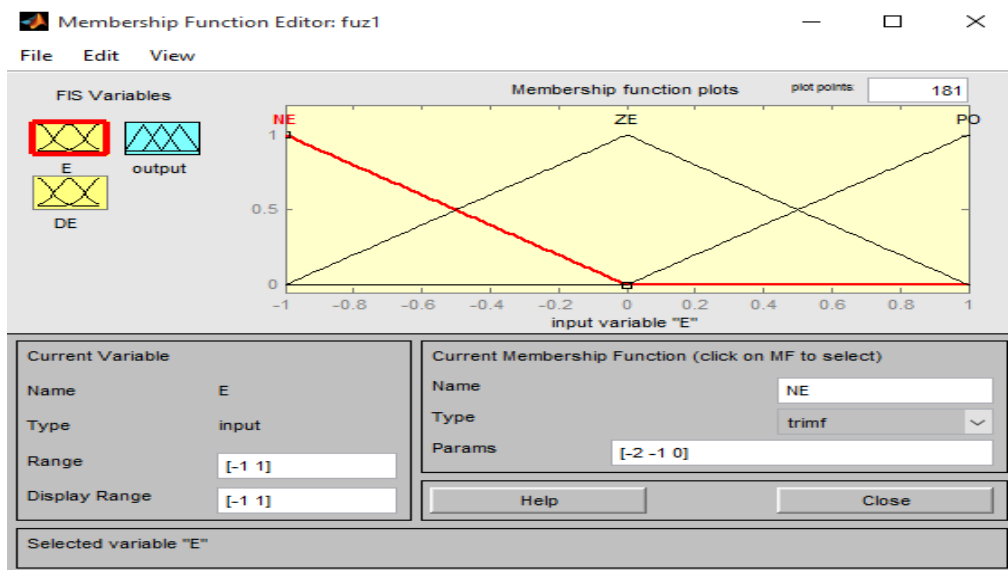


Figure 3.15 Error Input

As the error input is detected, Figure 3.16 show the change error

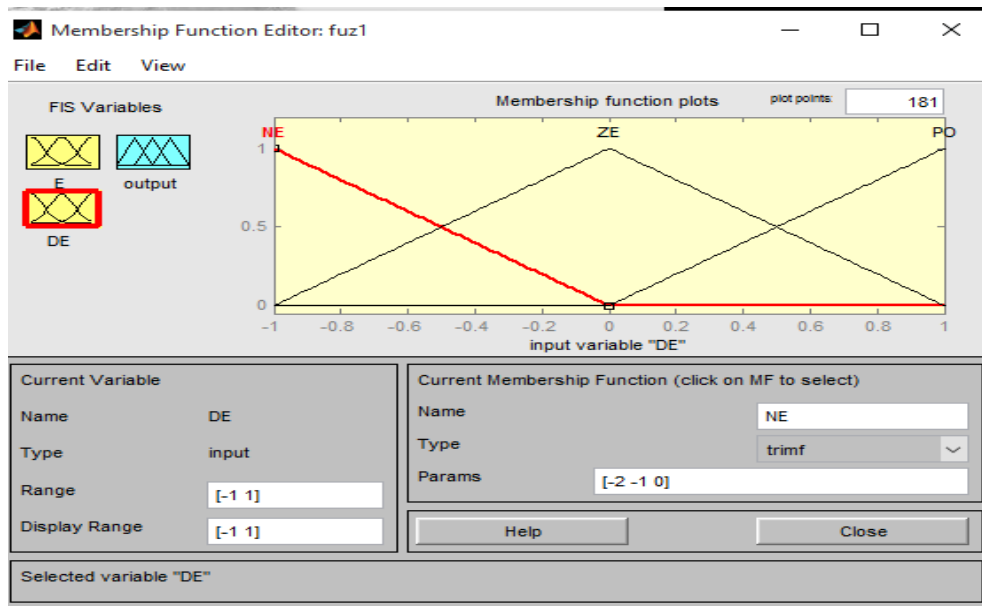


Figure 3.17 Change in Error

Changing in errors will remove interference, noise and then the Figure 3.18 is the Fuzzy Output.

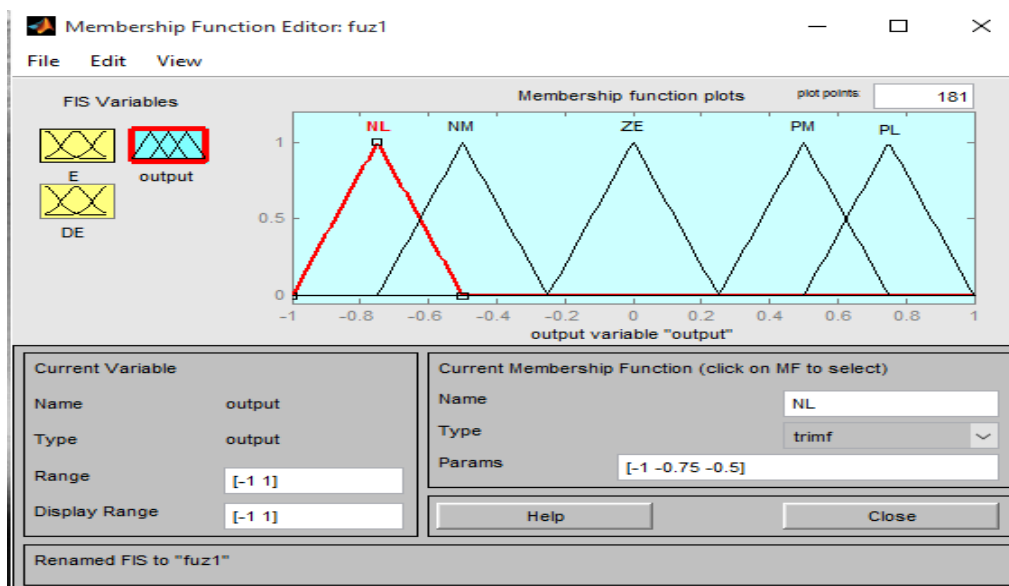


Figure 3.18 Fuzzy Output

In the designing the Fuzzy-PID, the inputs to the FLC still remains the error (E) and the time derivative of the error (or change in error), $\frac{dE}{dt}$ while the outputs in this case are the gain parameters K_p , K_i , and K_d . the structure of Fuzzy-PID

control system is a two input-three output Structure. For obtaining the range or the universe of discourse of the input and output membership functions, the PID control algorithm is tuned using the conventional Zeigler Nichols Method of the MATLAB PID tuner.

In pH concentration control process, the maximum error reached could be 100% (or unity) and accordingly the universe of discourse for the fuzzy input error (E) is fixed as: (-1, 1) and the change in error is also chosen with the same range, that is: (-1, 1). In order to integrate the PID with the fuzzy system, scaling factors are determined. The scaling factors GE, GCE, GIE and GU from K_p , K_i , and K_d gains used by the conventional PID compensator. Comparing the expression of the PID and the proposed Fuzzy-PID compensator, the following expressions are established:

$$GE = 1 \tag{3.25}$$

$$GCE = GE \times \frac{K_d}{K_p} \tag{3.26}$$

$$GIE = GE \times \frac{K_i}{K_p} \tag{3.27}$$

$$GU \times GE = K_p \tag{3.28}$$

Substituting the values of the PID gains into the equations above gives:

$$GCE = 1 \times \frac{1.75}{3.46} = 0.506$$

$$GIE = 1 \times \frac{0.661}{3.46} = 0.191$$

$$GU = \frac{3.46}{1} = 3.46$$

Where:

GCE = gain of change in error

GE = gain of error

GU = gain of control input (the reason for “U” here is because control input is represented with letter “u” in time domain or “U” in s-domain i.e. $u(t)$ or $U(s)$)

GIE = gain of integral error.

Having obtained these values, the fuzzy-PID model is implemented in MATLAB/Simulink environment. The Simulink model, representing the MATLAB/Simulink Graphical User Interface (GUI), developed for the purpose of carrying out simulation test in this work is shown in Figure 3.19.

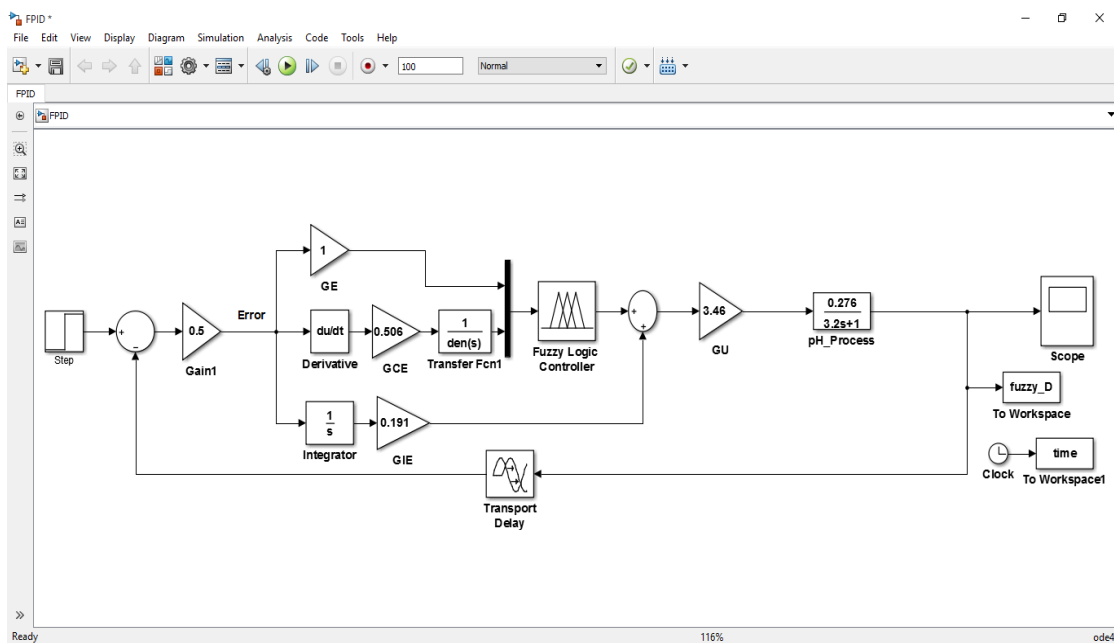


Figure 3.19: Simulink model of the proposed pH process with Fuzzy-PID

3.7.3 Algorithm of Proposed System

The MATLAB/Simulink extension files (fuzzy_D.slx) and the fuzzy inference system file (fuz1.fis) for the fuzzy-PID MATLAB programmes are given below:

```
clc;
```

```
clear;
```

```
close all;
```

```

%Create a FIS using mandanifis.
fis1 = sugfis;
%Add input variables to the FIS.
fis1 = addInput(fis1,[-1 1],'Name','E');
fis1 = addInput(fis1,[-1 1],'Name','DE');
%Add three uniformly distributed overlapping triangular membership functions
(MFs) to each
%input. The MF names stand for negative (NE), zero (ZE), and positive (PO).
fis1 = addMF(fis,'E','trimf',[-2 -1 0],'Name','NE');
fis1 = addMF(fis,'E','trimf',[-1 0 1],'Name','ZE');
fis1 = addMF(fis,'E','trimf',[0 1 2],'Name','PO');
fis1 = addMF(fis,'DE','trimf',[-2 -1 0],'Name','NE');
fis1 = addMF(fis,'DE','trimf',[-1 0 1],'Name','ZE');
fis1 = addMF(fis,'DE','trimf',[0 1 2],'Name','PO');
%Plot the input membership functions.
figure
subplot(1,2,1)
plotmf(fis,'input',1)
title('Input 1')
subplot(1,2,2)
plotmf(fis,'input',2)
title('Input 2')
fis1 = addOutput(fis1,[-1 1],'Name','U');
fis1 = addMF(fis1,'U','trimf',[-1 -0.75 -0.5],,'Name','NL');
fis1 = addMF(fis1,'U','trimf',[-0.75 -0.5 -0.25],,'Name','NM');
fis1 = addMF(fis1,'U','trimf',[-0.25 0 0.25],,'Name','ZE');
fis1 = addMF(fis1,'U','trimf',[0.25 0.5 0.75],,'Name','PM');
fis1 = addMF(fis1,'U','trimf',[0.5 0.75 1],,'Name','PL');

```

The Fuzzy rules = [..."E==NE & DE==NE => U=NB"; ..."E==ZE & DE==NE => U=NM"; ..."E==PO & DE==NE => U=ZE"; ..."E==NE & DE==ZE => U=NM"; ..."E==ZE & DE==ZE => U=ZE"; ..."E==PO & DE==ZE => U=PM"; ..."E==NE & DE==PO => U=ZE"; ..."E==ZE & DE==PO => U=PM"; ..."E==PO & DE==PO => U=PL" ...];

The proposed fuzzy rules are summarized in table 3.2 The membership functions of these inputs fuzzy sets are shown in Figure 3.20 The linguistic variable levels are assigned as: negative (NE), negative large(NL), zero (ZE), and positive large (PL) and positive Medium (PM). Similarly, the fuzzy set for error change (ΔE).

In designing the rule base as contained in Table 3.2 these rules often are designed to mimic a conventional PID Controller in the velocity mode so as to incorporate information about process nonlinearities, constraints and uncertainty. Because the input variables belong to the fuzzy set, they produce non-zero membership values for a number of different rules.

Rule 1: if E is ZE and if ΔE is ZE, then the control action is ZE

Rule 2: if E is PS and if ΔE is ZE, then the Control is PS

Table 3.2: Fuzzy rules.

E/ ΔE	NE	ZE	PO
NE	NL	NM	ZE
ZE	NM	ZE	PM
PO	ZE	PM	PL

The rule evaluation focuses on operation in the antecedent of the fuzzy rules in fuzzy sliding mode controller

```
fis1 = addRule(fis1,rules);
```

```
gensurf(fis1)
```

title('Control surface of FIS')

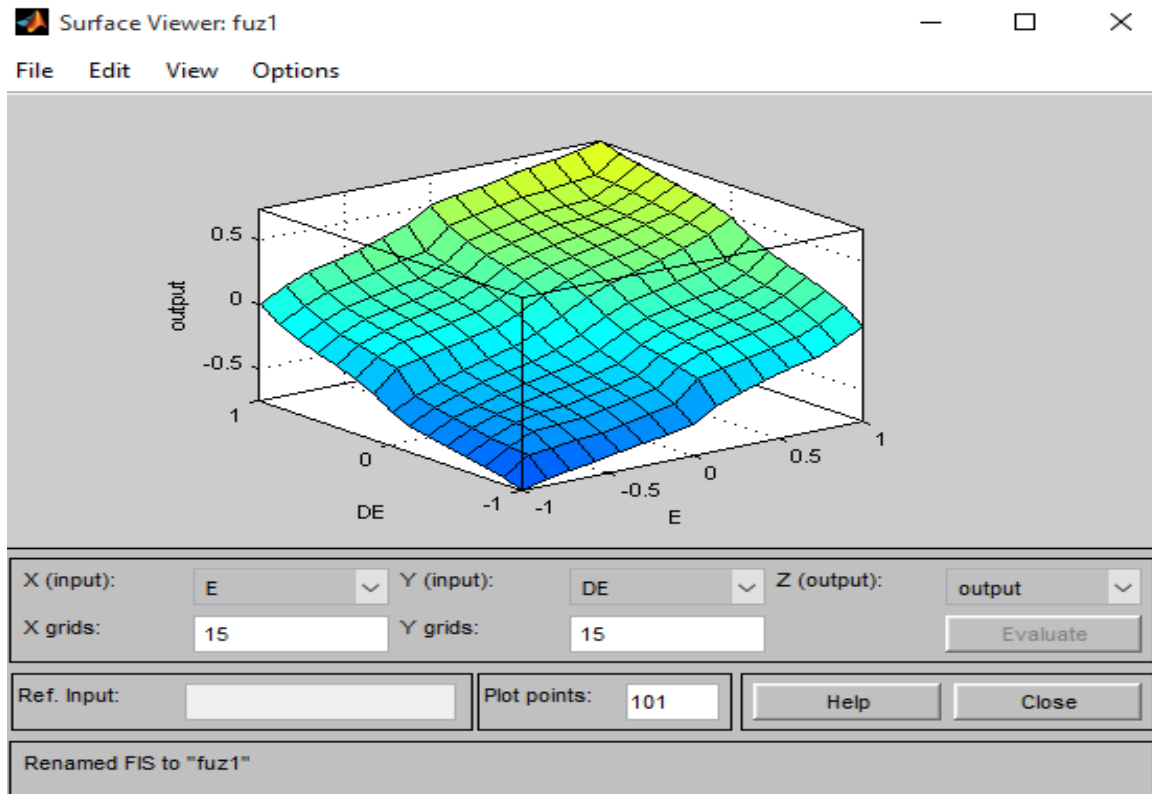


Figure 3.20: Fuzzy inference System Surface view

% Fuzzy Inference System (FIS) File:

fuz1.fis

[System]

Name='fuz1'

Type='mamdani'

Version=2.0

NumInputs=2

NumOutputs=1

NumRules=9

AndMethod='min'

OrMethod='max'

ImpMethod='min'

```

AggMethod='max'
DefuzzMethod='centroid'
[Input1]
Name='ERROR'
Range=[-1 1]
NumMFs=3
MF1='NE': 'trimf', [-2 -1 -0]
MF2='ZE': 'trimf', [-1 0 1]
MF3='PO': 'trimf', [0 1 2]
[Input2]
Name='DE-ERROR'
Range=[-1 1]
NumMFs=3
MF1='NE': 'trimf', [-2 -1 0]
MF2='ZE': 'trimf', [-1 0 1]
MF3='PO': 'trimf', [0 1 2]
[Output1]
Name='OUTPUT'
Range=[-1 1]
NumMFs=5
MF1='NL': 'trimf', [-1 -0.75 -0.5]
MF2='NM': 'trimf', [-0.75 -0.5 -0.25]
MF3='ZE': 'trimf', [-0.25 0 0.25]
MF4='PM': 'trimf', [0.25 0.5 0.75]
MF5='PL': 'trimf', [0.5 0.75 1]
[Rules]
1 1, 1 (1) : 1
1 2, 2 (1) : 1
1 3, 3 (1) : 1

```

2 1, 2 (1) : 1

2 2, 3 (1) : 1

2 3, 4 (1) : 1

3 1, 3 (1) : 1

3 2, 4 (1) : 1

3 3, 5 (1) : 1

3.8 Unified Modeling Language (UML)

The UML has become the standard language in object-oriented analysis and design. It is widely used for modeling software systems and is increasingly used for high designing non-software systems and organizations. The UML of this System consist of activities diagram, sequence diagram, Class diagram, Use case diagram and collaboration diagram. A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. The activities diagram is shown in figure 3.21

Activities Diagram

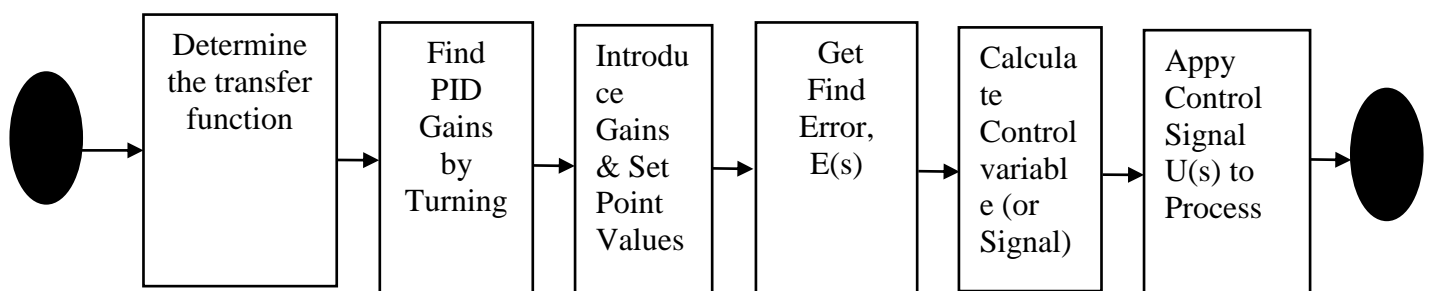


Figure 3.21: Activities Diagram

The activities diagram demonstrated a preview from the system transfer function to PID the turn, then use Fuzzy to obtained the errors and compute the state variable setpoint signal to gets control output.

The Use Case diagram for the proposed system is shown in figure 3.22 demonstrated

Use Case Diagram

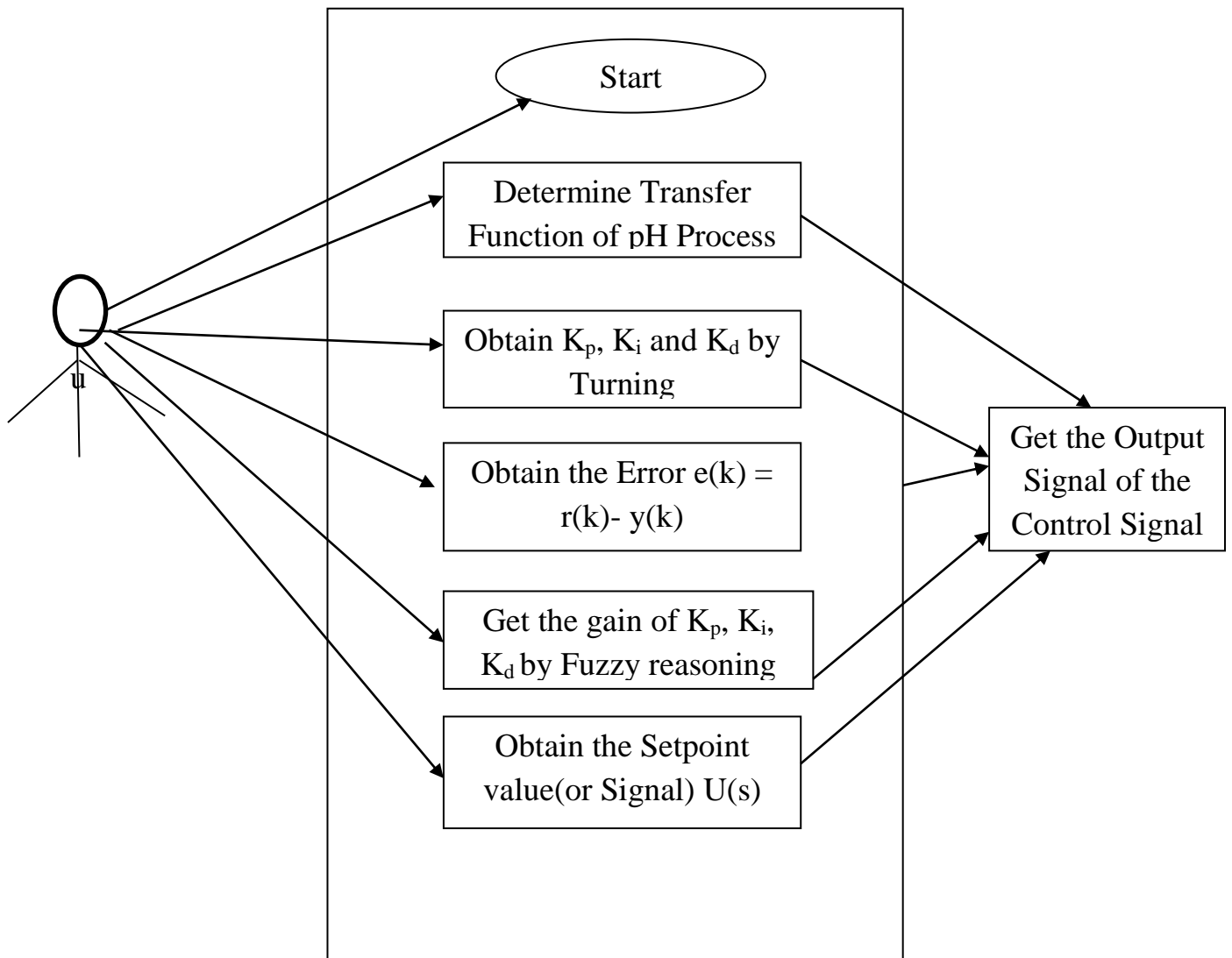


Figure 3.22: Use Case Diagram for the Proposed System

The use case diagram described the user and how the system operations are being defined the pH obtained the transfer function, the turn the PID to the respective gains. Then the fuzzy reasoning the setpoint variables and obtain the control setpoint value for the pH, the obtain the results.

Sequence Diagram

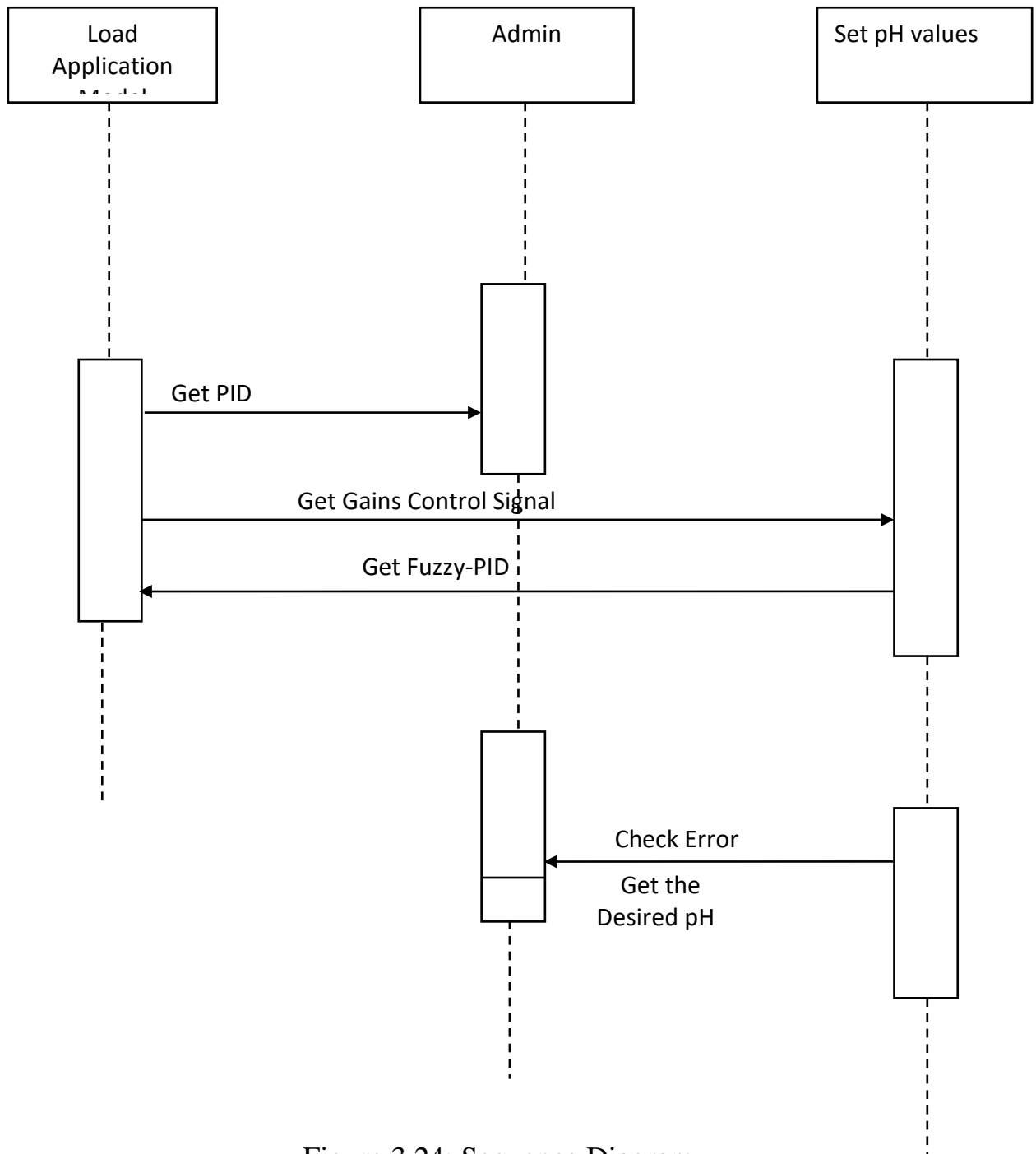


Figure 3.24: Sequence Diagram

The sequence diagram shows where the user loads the pH application. Then the admin Admin create the authentication by assigning user name and password. The system gets PID controller and then the gain. The hybrid Fuzzy-PID computes the set values of pH by checking for errors and produces the desired pH for the system.

Collaboration Diagram

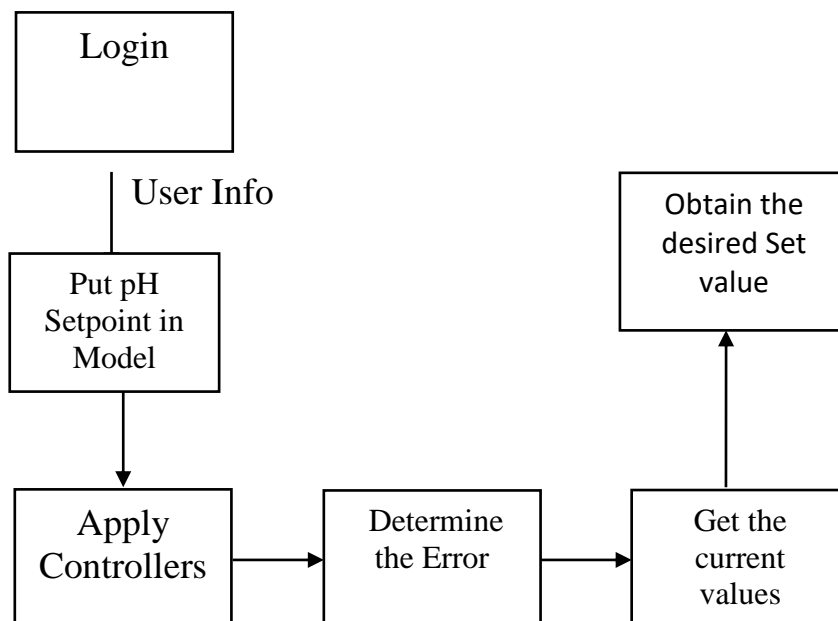


Figure 3.25: Collaboration Diagram

3.8.1 Flowchart for the Proposed System

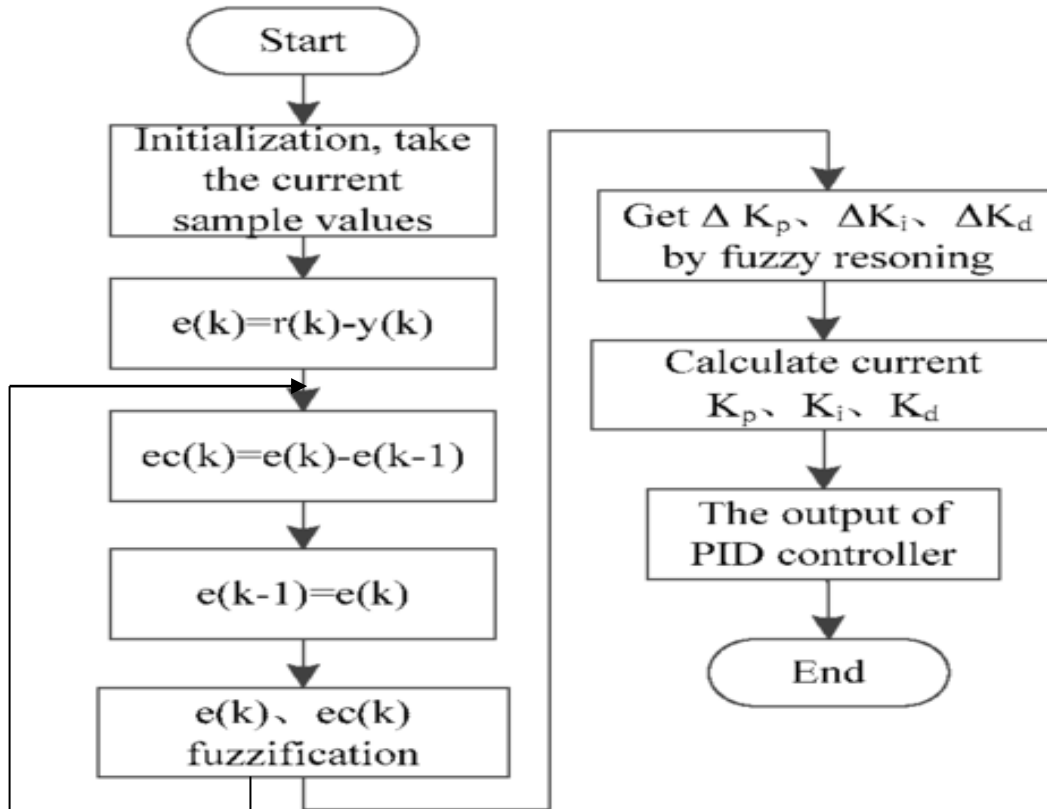


Figure 3.26: Proposed System Flowchart

Figure 3.26 shows the proposed system flowchart, initially, the current state sample values will evaluate the error difference several times and then carry out fuzzification process. The fuzzification apply the linguistic reasoning on the gains proportional, derivatives and integral K_p , K_i , and K_d , the reasoning eliminate the degree of the uncertainty, nonlinearity and linearized the parameters to the PID controller and then to the Plant.

3.9 Integrated Development Environment Design Tool

Matlab-Simulink

Matlab stands for Matrix Laboratory. It is a very powerful tool for Science and Engineering. It is typically used for Math Computation, Algorithm development, Modeling, Simulation and Prototyping, Data analysis, exploration and visualization, Scientific and Engineering Graphics as well as application development. The simulink is a time based software packages that is included in Matlab and its main task is to solve Ordinary Differential Equation (ODE) numerically. The need for the numerical solution comes from the fact that there is not an analytical solution for all differential Equation (DE) especially for those that are nonlinear.

Simulink Library

Simulink is a MATLAB-based graphical programming environment for modeling, simulating and analyzing multidomain dynamical systems. Its primary interface is a graphical block diagramming tool and a customizable set of block libraries which is used in design of the chemical plants. The Matlab Simulink IDE is as shown in Figure 3.27

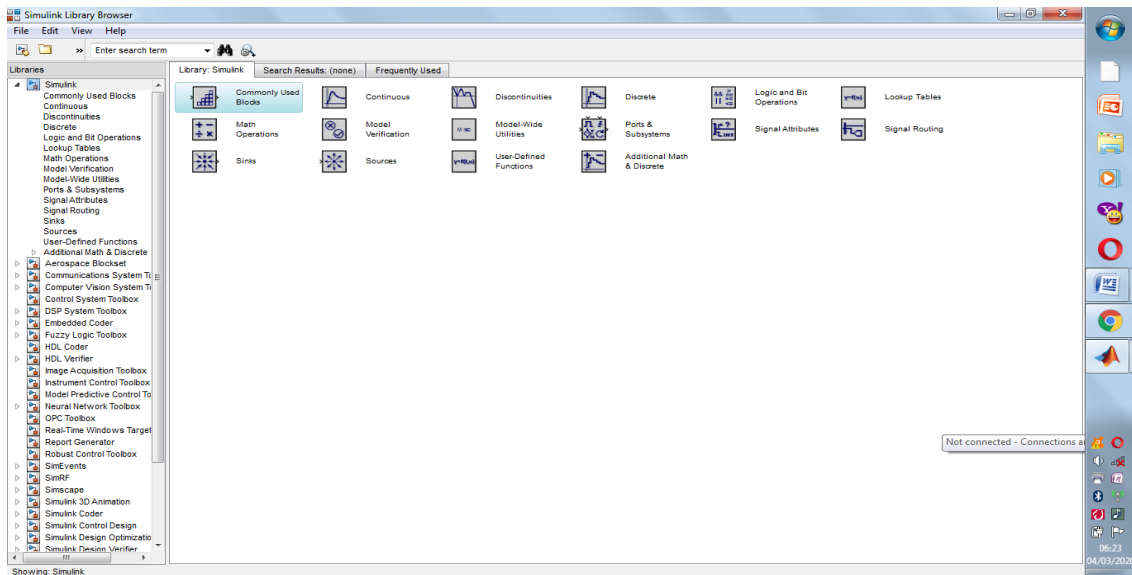


Figure 3.27: MATLAB/Simulink GUI environment for simulation

3.9.1 Netbeans Integrated Development Environment (IDE)

The NetBeans IDE provides support for several languages (PHP, C/C++, Java Scripts etc) and frameworks. NetBeans Development tools is an open source project dedicated to provide solid rock on software development product. The NetBean and NetBean Platform address the needs for developers. The NetBeans IDE as shown in Figure 3.28 will be used in conjunction with JAVA in the developing the application for the pH neutralization Process.

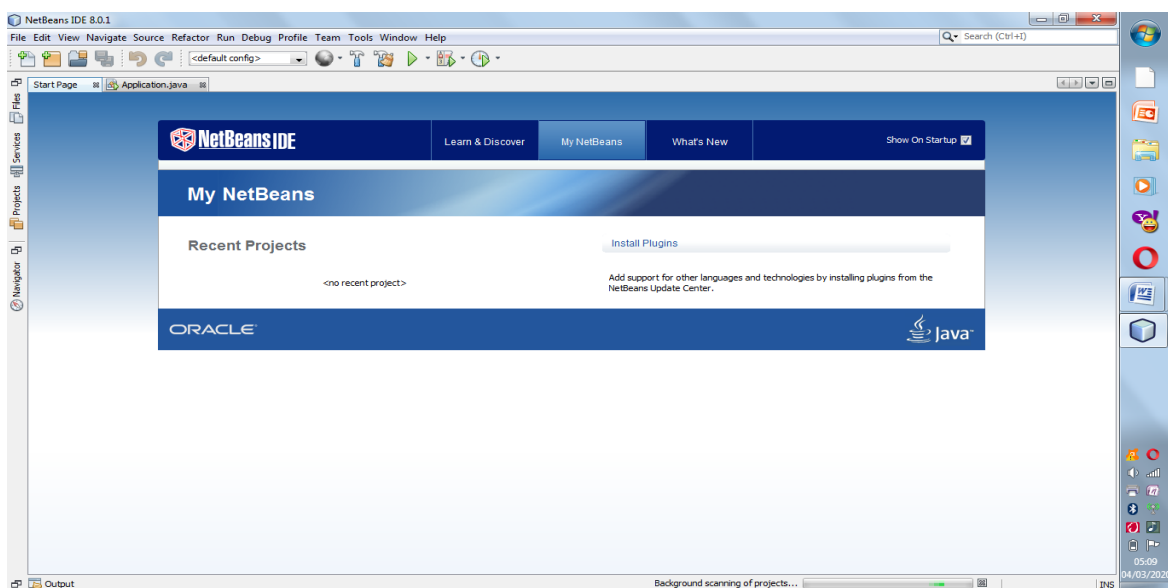


Figure 3.28: Netbeans IDE

3.9.2 pH Data for Water treatment

Excessively high and low pHs can be detrimental for the use of water. High pH cause a bitter taste, water pipes and water using appliances becomes encrusted with deposits, and it depresses the effectiveness of the disinfection of chrine, thereby causing the need for additional chlorine when pH is high. Low pH water will corrode or dissolve metals and other substance. Figure 3.29 demonstrated the data values for water treatment.

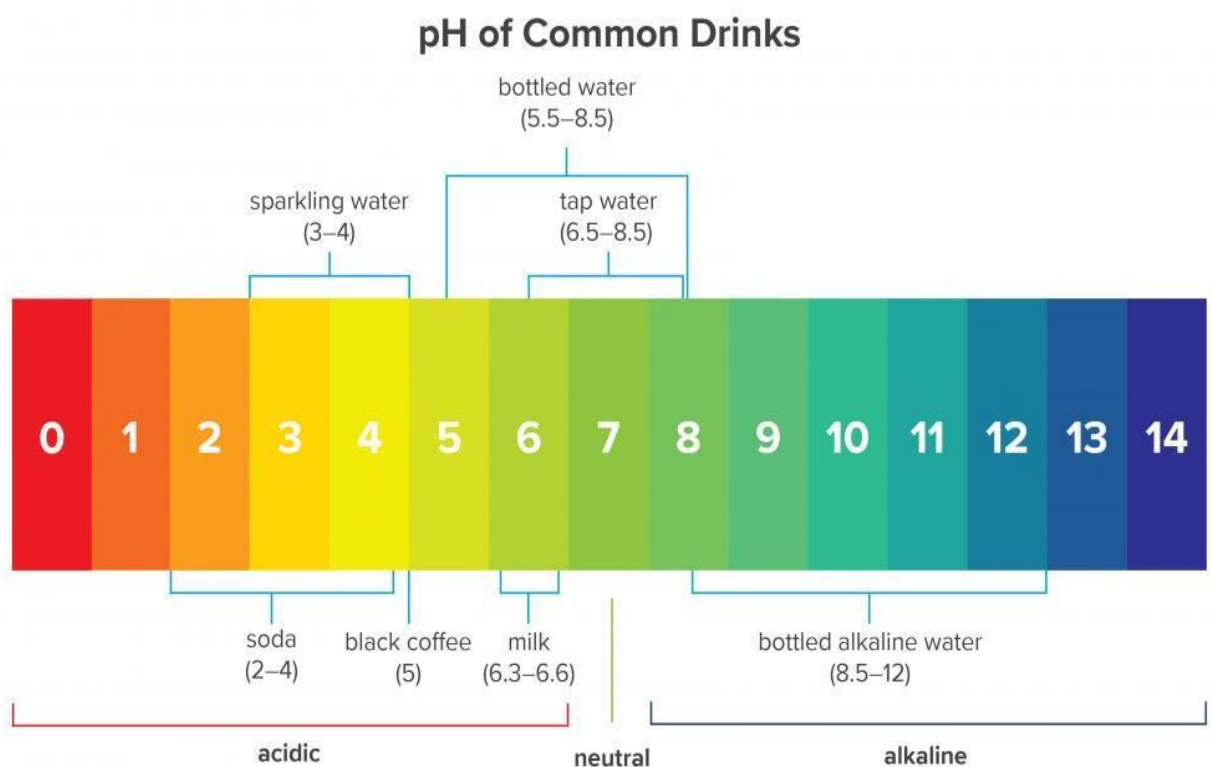


Figure 3.29: pH of drinking water (<https://www.healthline.com/health/ph-of-drinking-water>, 2019)

Most acidic water which are sparkling water with pH of (3-4), soda (2-4), Black coffee (5), milk (6.3-6.6), but bottle water (5.5-8.5) and tap water (6.5-8.5) are between acidic, neutral or alkaline. Also bottled alkine water (8.5-12) is alkaline. This sets of investigated data are use to train the model data for 5 acidic, 7 neutral and 9 Alkaline.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Implementation

The implementation defining how the information system was built using the MATLAB Simulink and the application was developed using JAVA, Netbean IDE 8.0.1.

4.2 Programming Language Used

The implementation of this work involved simulation and application development. The simulation result was obtained using MATLAB, and JAVA Kits. Netbean 8.0.1 version was used as Integrated Environment (IDE)

4.2.1 System Requirements

The basic requirement necessary to installed and deploy the application include

4.2.1.1 Hardware Specification

The hardware specifications are as follows:

- i. An enhanced keyboard, F1-F12
- ii. Intel Core i7, Core i5 or Core i3 Processor
- iii. A Minimum of 4 to 8GB of RAM
- iv. At least 500GB to 1 Terabyte of Hard Disk Drive (HDD)
- v. Mouse Type: PS2 or USB enabled
- vi. Display Device 15”to 19”inch Monitor

4.2.1.2 Software Specification

The following the software specification requirement

Operating System: Window 7, 8 or 10

Development Kit: JDK, Netbean and Matlab, Excel Spreadsheet

4.3 Results

The results of the simulations carried out in MATLAB/Simulink environment for pH concentration control system whose mathematical models, representing the dynamic of the process, are combined with Proportional Integral Differential (PID) control algorithm (compensator), and Fuzzy Logic Fuzzy-PID Control algorithm (compensator) are presented in this chapter. The results are presented in terms of unit step responses. A step response represents the output of a desired value of an industrial or chemical process expressed as unity (or 100%) such that the control algorithm when implemented as part of the system is expected to ensure that the output of the process reaches close or track this predetermined value and maintained with negligible error. The step responses to a unit step input (representing 100% output efficiency) are presented below:

1. The step response of pH concentration process without a compensator or control algorithm.
2. The step response of pH concentration process compensated with PID control algorithm.
3. The step response of pH concentration process compensated with Fuzzy-PID control algorithm.

4.3.1 Unit Step Response of Uncompensated pH Process

The unit step response performance of pH concentration process without a compensator was investigated by simulating pH process using MATLAB/Simulink developed model shown in Appendix I. The unit step response plot of the result is shown in Figure 4.1.

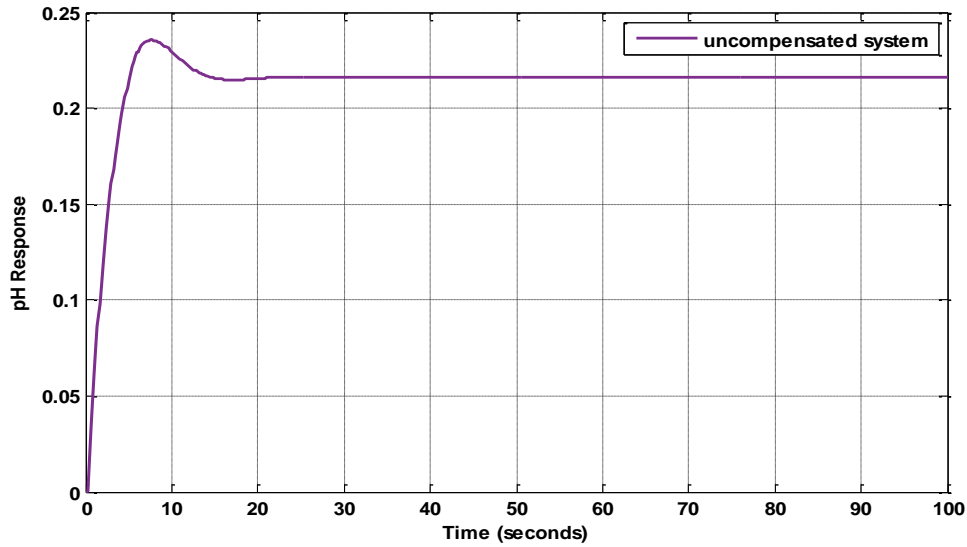


Figure 4.1 Unit step Response of uncompensated pH process.

Figure 4.1 the results shows for the performance of uncompensated pH regulating process that is, without a control model. It is showed that the response performance is not suitable in the pH process plant due to the fact that is highly affected by uncertainty, nonlinearity and perturbation.

4.3.2 Unit Step Response of PID Compensated pH Process

The unit step response of pH process compensated with tuned PID algorithm simulated using MATLAB/Simulink model in Figure 3.7 of chapter three is shown in Figure 4.2.

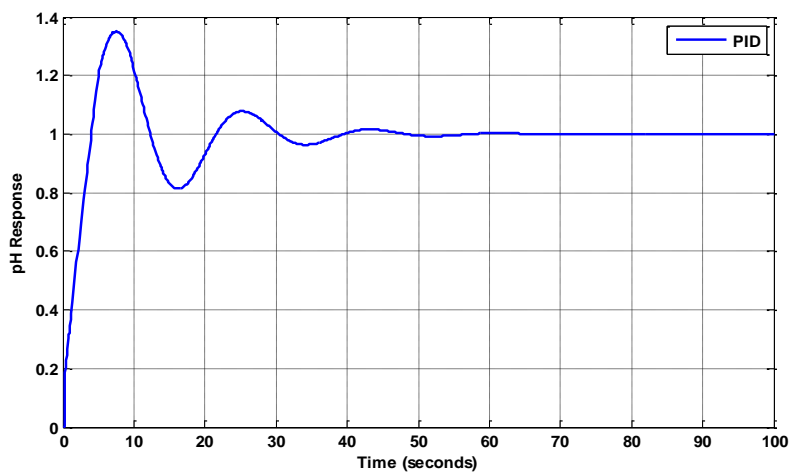


Figure 4.2 Step response of PID compensated pH process.

Figure 4.2 shows the step response of a PID Control Model for a compensator. It is observed that the step responses are dynamical responding to the set point signal. Here the uncertainty and perturbation reduces due to the PID Controller, though the respond signal is still slightly affected by noise and other factors in the system.

4.3.3 Unit Step Response of Fuzzy-PID Compensated pH process

The step response of Fuzzy-PID compensated pH process simulated using the Simulink model in Figure 3.19 of chapter three is shown in Figure 4.3.

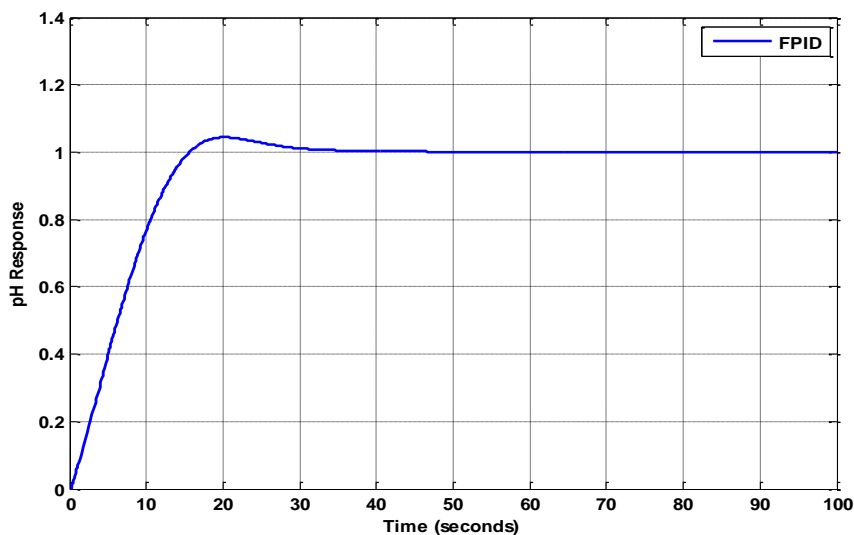


Figure 4.3 Step response of Fuzzy-PID compensated pH process.

The result in Figure 4.3 shows that the Fuzzy-PID known as FPID hybrid control model was able to address and handle the challenges of nonlinearity and perturbation. The system dynamic response was better suitable for the pH control plant to function.

4.4 Performance of Analysis in Terms Unit Step Input

This subsection presents the performance analysis of the various simulations considering Figure 4.1, Figure 4.2 and Figure 4.3. The step response performance

parameters are given in continuous time domain and are presented in Table 4.1. A remark for each simulation condition is presented in Table 4.2.

Table 4.1 Step response Performance Parameters

System state	Rise time (s)	Peak Time (s)	Settling time (s)	Overshoot (%)	SSE
Uncompensated	3.65	7.70	12.37	8.90	0.76
PID	3.28	7.60	37.04	35.20	0.19
Fuzzy-PID	11.06	20.22	26.88	4.34	0.1

SSE means Steady State Error.

The result obtained in table 4.1 can be further analyzed into various state responses

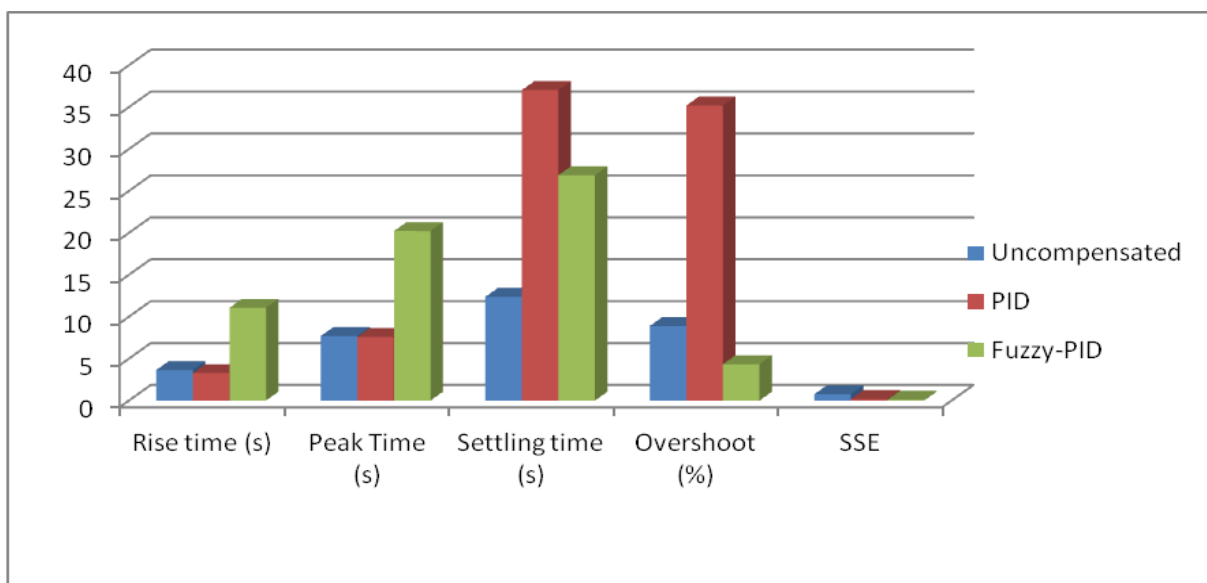


Figure 4.4: Analyzed Response of System State

Figure 4.4 show that with the use of fuzzy-PID hybrid model produces an overshoot of 4.34% which was better than the existing model of 32.21%, and PID model used as 35.20% and for uncompensated was 8.90% as well as the Settling Time for Fuzzy-PID was 26.88s, 37.04s for PID and 12.37s for uncompensated medium. In all cases the fuzzy-PID was better in all the evaluation parameters for

peak Time, Settling Time, Overtime and Steady Square Error, expect in rise time where the value of the hybrid model is highly.

Table 4.2 Remark for each of the Simulation result

System state	Remark with respect to unit step input
Uncompensated	The system could not reach the desired pH value (that is, 1 or 100%). Hence, it is unsatisfactory in this case
PID	The system was able to track or reach the desired pH value but showed high possibility of instability due to high overshoot. Hence it is unsatisfactory.
Fuzzy-PID (proposed system)	The system was able to track or reach the desired pH value with high stability showing very little (or no steady state error).

4.4.1 Performance Indices

The performance of a control system can be measured in terms of error indices such as Integral Square Error (ISE), Integral Absolute Error (IAE) and Integral Time Absolute Error (ITAE). The performance of the the proposed system is examined in terms of ISE, IAE and ITAE. These performance indices are defined mathematically in eq. (4.1), eq.(4.2) and eq. (4.3)

$$ISE = \int_0^{\infty} e^2(t) dt \quad (4.1)$$

$$IAE = \int_0^{\infty} |e(t)| dt \quad (4.2)$$

$$ITAE = \int_0^{\infty} t |e(t)| dt \quad (4.3)$$

The ISE focuses on the square of the error function and deals with both the positive and negative values of the error. IAE utilizes the size of the error and increases for either positive or negative error. ITAE deals with prolong transients and offers performance index that is more selective compare to ISE or IAE.

4.5 Performance Responses for pH Values in Medium using PID and FPID Control Model

Table 4.3 Shows the responses for pH values for acid with pH -5, Base with pH- 9 and Neural Medium with pH- 7 using the analysis tools on PID, FPID, the error analysis performance was based on ITAE (Integral Time Absolute Error), ISE (Integral Square Error) and IAE (Integral Absolute Error)

Table 4.3 Error Performance Responses for pH value-5

Acid pH- 5

Control Model	ITAE	ISE	IAE
PID	39.61	164.70	51.58
FPID	28.86	71.75	39.31

The results of Table 4.3 are represented in the figure 4.5

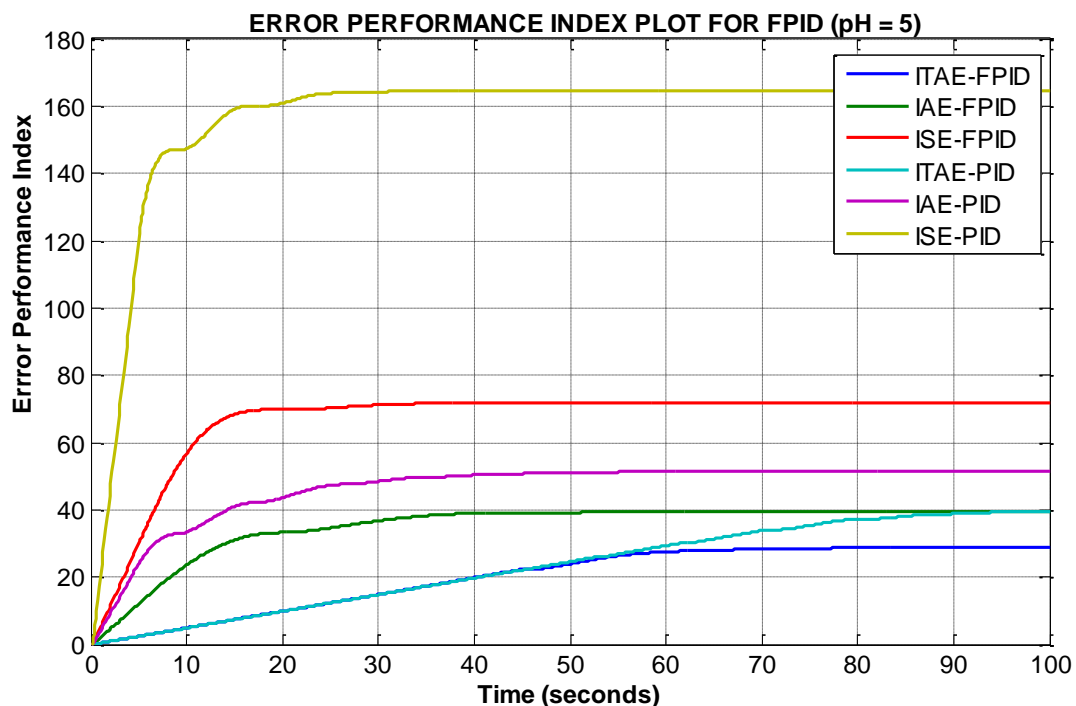


Figure 4.5: Error Performance Index PID, and FPID, pH-5

Figure 4.5 shows the responses error performance in index using the error analysis tools of ITAE, ISE and IAE. It was observed that ITAE for FPID has

an error margin of 28.86s and in PID was 39.61. The ISE produces larger error values of 164.70s and 71.75 in PID and FPID respectively.

Table 4.4: Error Performance Responses for pH value-7

Neural Medium pH value – 7

Control Model	ITAE	ISE	IAE
PID	41.37	322.90	72.20
FPID	34.11	141.80	56.47

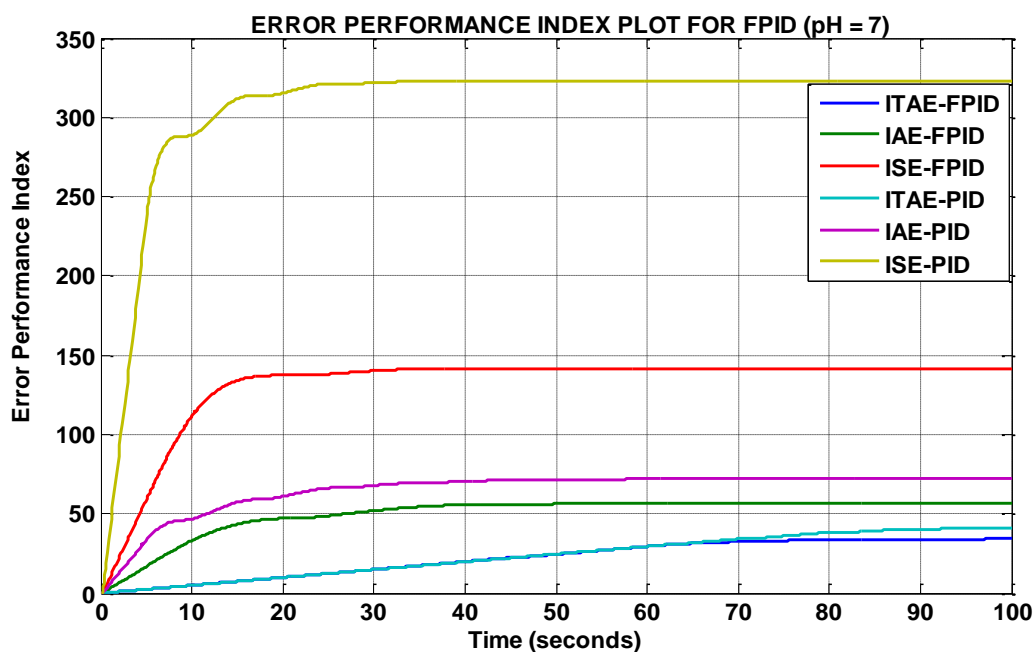


Figure 4.6: Error Performance Index PID, and FPID, pH-7

Figure 4.6 shows the error performance index for PID, FPID for pH value of 7. It was observed that FPID produces least error of 34.11 in ITAE (Integral Time Absolute Error), followed by the ITAE for PID with 41.37

Table 4.5: Error Performance Responses for pH value-9

Base Medium pH value – 9

Control Model	ITAE	ISE	IAE
PID	42.75	533.70	92.83
FPID	36.53	234.80	73.09

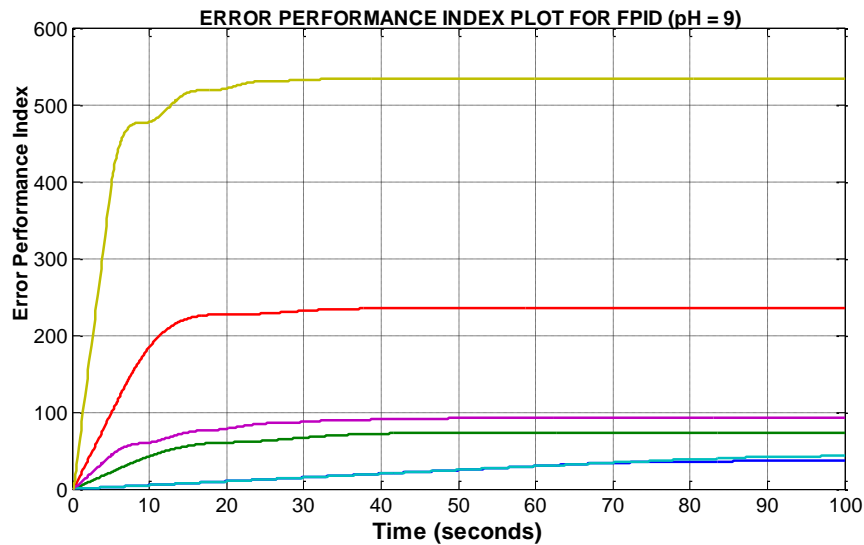


Figure 4.7: Error Performance Index PID, and FPID, pH-9

In the results obtained above it is clearly showed that ITAE produces better responses with less error in both PID of 42.75 and FPID 36.53 respectively. But FPID produces least errors with ITAE. The next followed by IAE and then ISE that has highest error Margin.

In the performance indices of the above shows in the three media considered, the error performance indices of indices of FPID control pH neutralization process were better than that of PID control system. This is obvious looking at the numerical values of ISE, IAE and ITAE in the various media of acid, neutral and base for PID controlled and FPID controlled pH neutralization process. That is, least error performance indices were obtained using FPID controller to regulate the pH neutralization process. Hence, the FPID provided more efficient and robust performance in terms of error handling which is a measure of the deviation of actual pH value from set or reference pH value.

4.6 Validation of Fuzzy-PID Unit Step Response for varying pH value

In order to validate the performance of the proposed Fuzzy-PID control algorithm, simulations were further carried out setting the desired pH value at 5, 7, 9 which represent acid, neutral, and base medium respectively. The choice of these values was necessitated because pH above 9 and pH below 4 are considered to be harmful to the surroundings (Ram *et al.*, 2016). The essence of this was to study the effectiveness of the proposed system in maintaining effective output at any desired pH value.

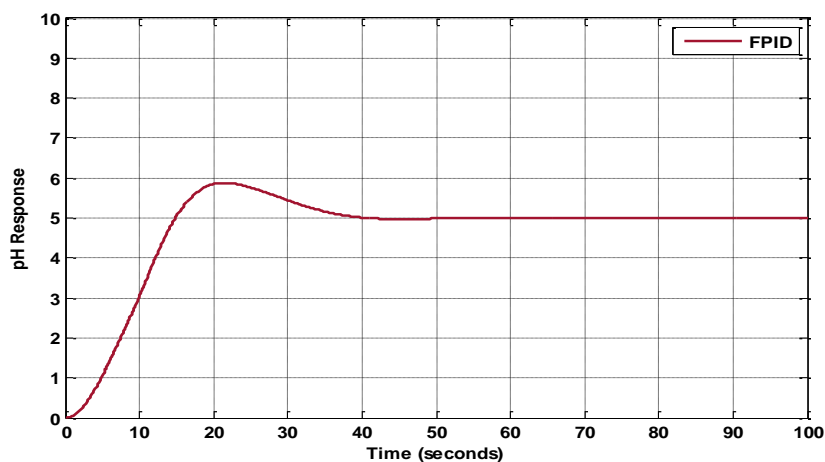


Figure 4.8: Step response at pH setpoint 5 (Acid medium)

Figure 4.8 shows the step responses time for Fuzzy-PID in an acid medium of pH value of 5. It was observed that the FPID was able to track the desire set signals input of 5 at its settling time

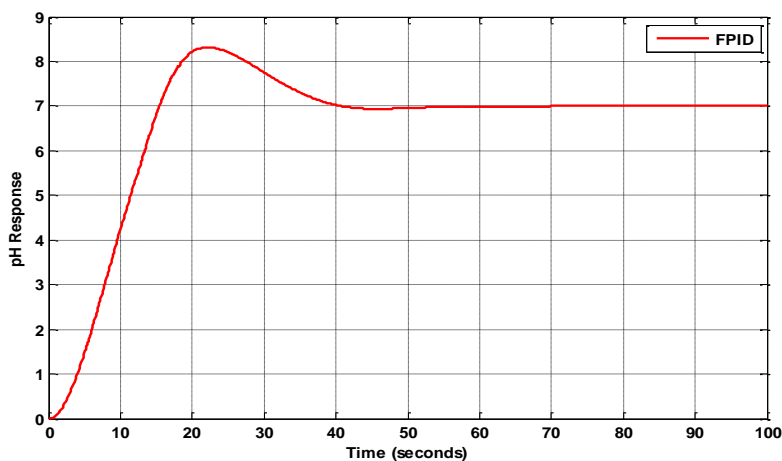


Figure 4.9: Step response at pH setpoint 7 (Neutral medium)

Figure 4.9 shows the step responses time for Fuzzy-PID in an Neutral medium of pH value of 7. It was observed that the FPID was able to track the desire set signals input of 7 at its settling time.

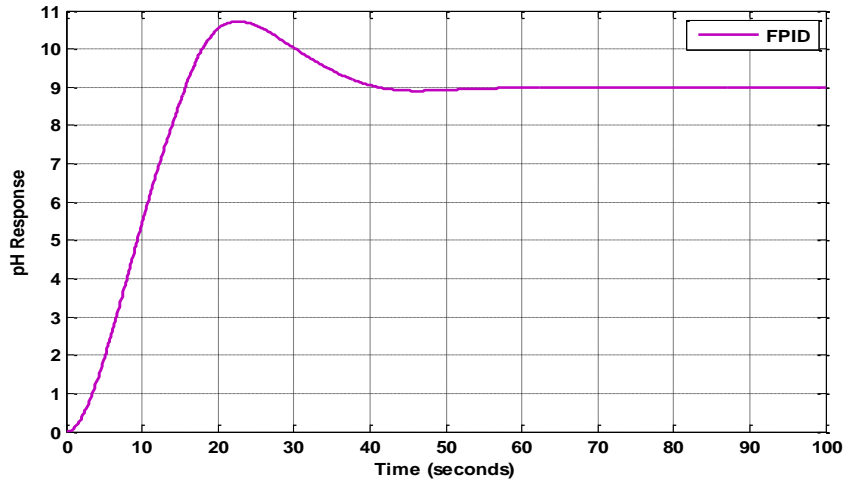


Figure 4.10: Step response at pH setpoint 9 (Base medium)

Figure 4.8, 4.9 and 4.10 shows the response of track input for pH values of 5 (acidic), 7 (Neutral) and 9 for Basic medium the hybrid model was able to track a steady and better performance in the pH Solution.

Table 4.6: Step responses performance parameters at varying pH values

System state (Fuzzy-PID)	Rise time (s)	Peak Time (s)	Settling time (s)	Overshoot (%)	SSE
Acid medium	10.11	21.36	36.57	17.6	0.46
Neutral medium	10.67	22.2	37.50	18.8	0.67
Base medium	10.84	22.7	38.02	19.1	0.83

Table 4.6 shows the that the results contain in figure 4.11, it was observed that level of the pH neutralization process was accepted at the Neutral Medium, the result obtained was recommed for water treatment process. It is showed that Base mediun is high due to high value in the pH scale.

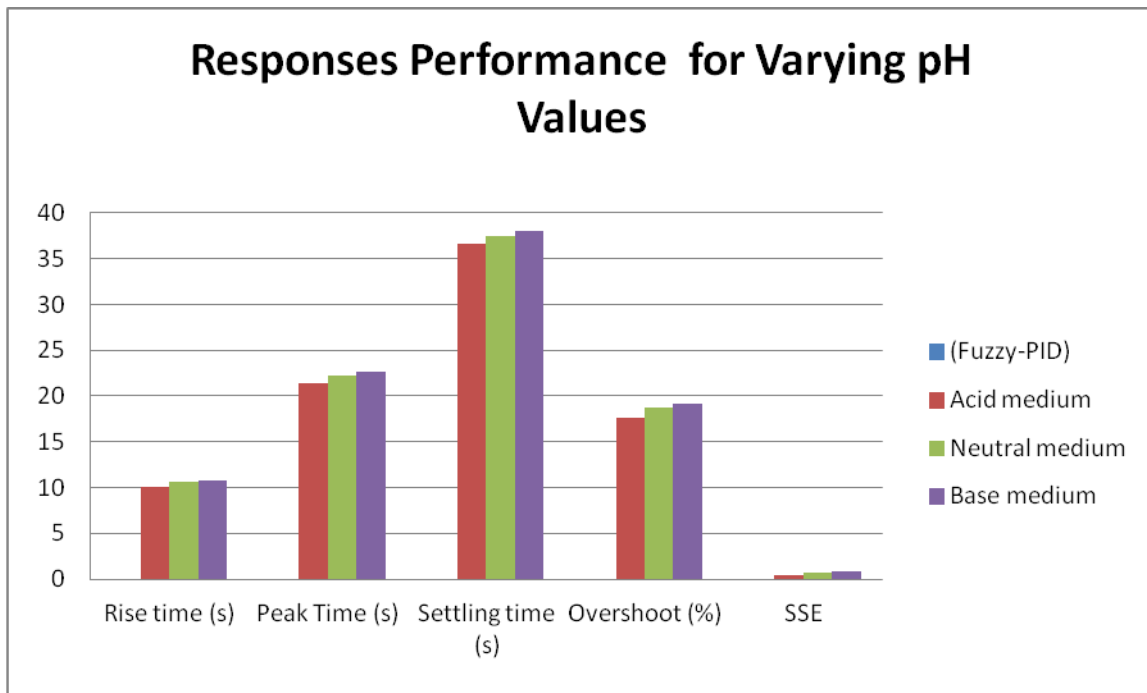


Figure 4.11: pH responses of Acid Base and Neutral Point

The comparison of the performance of FPID and PID is shown on Figure 4.12

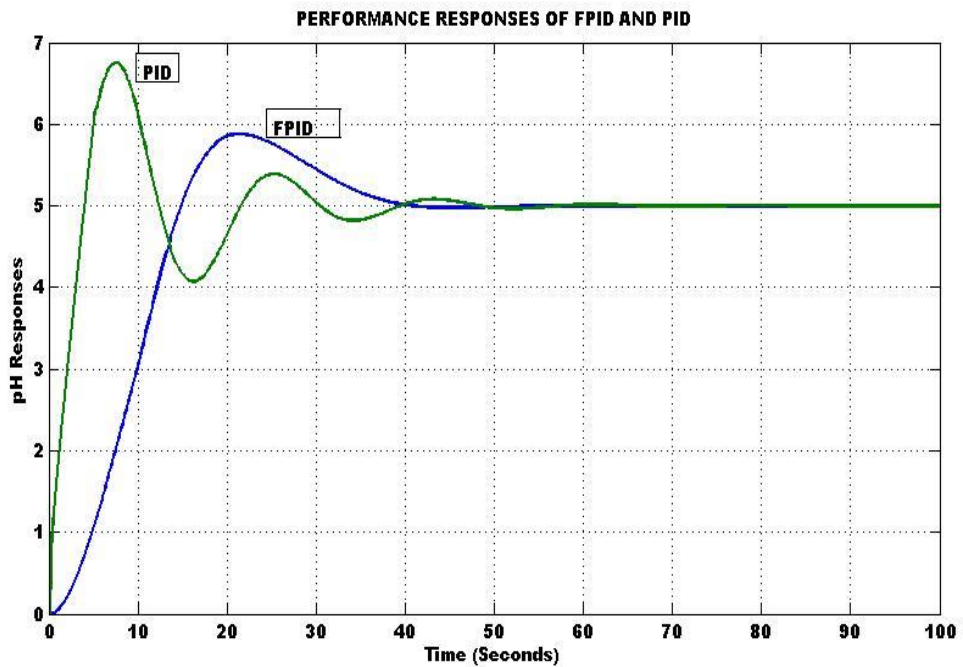


Figure 4.12: Performance Responses for FPID and PID for pH value of 5(Acid Medium)

4.7 Discussion of Results

As shown in Table 4.1, the system rise time was 3.64 seconds, the time to peak (or peak time) was 7.70 seconds, settling time was 12.37 seconds, and the percentage overshoot was 8.9% when the system has not been compensated that is, no control algorithm has been integrated into the pH process to hence the neutralization of the waste water treatment. Despite the fact that the time domain performance parameters of the system seems to be promising, the system at this condition was not capable of tracking or maintaining a desired pH value with 100% effectiveness. Hence, the need for a compensator that will ensure that the system attains a setpoint pH target with improved output or response performance.

When a PID compensator was introduced into the system, the rise time becomes 3.28 seconds, time to peak was 7.60 seconds, the settling time was 37.04 seconds, and the percentage overshoot was 35.2%. In this condition, the introduction of PID was able to ensure that the desired pH value was reached or tracked with respect to unit step input, which represents 100% efficient tracking. Despite the fact that the introduction of PID ensured that the desired pH value for unit step input was attained, it can be seen from Figure 4.2 that it introduced a high overshoot which may affect the performance of the system as it continuous to run and this can cause instability in the process.

In order to address the limitation of the PID, that is the high percentage overshoot, an intelligent algorithm using fuzzy logic technique in combination with PID parameters was developed and introduced into the system. As can be seen in Table 4.1, the system rise time was 11.06 seconds, time to peak was 20.2 seconds, settling time was 26.88 seconds, and the percentage overshoot was 4.34% when the Fuzzy-PID was introduced. It is obvious that the addition of the proposed Fuzzy-PID compensator ensured that the desired pH value maintained

at 100% (that is unit step input) was reached or tracked and the same time reduced the overshoot value to 4.38%. This indicated an efficient and stable system with satisfactory performance.

Furthermore, to validate the step response performance of the Fuzzy-PID, simulations were carried out by setting the pH value at a desired level of 5, 7 and 9 for acid, neutral and base medium respectively. It can be seen as shown in Figures 4.4, 4.5 and 4.6 that the Fuzzy-PID was able to guarantee that the setpoint target was met with improved stability for the various media.

4.8 Interface of Application for pH Neutralization

The software application front end interface for the stand alone intelligent Application as show in figure 4.13 being the login interface window



Figure 4.13: login Interface Window for pH Neutralization Process

The output of the stages of the pH Neutralization Process is described in the Figure 4.14



Figure 4.14: Stages of Neutralization Process of Water Treatment Plant

Stage I: At this phase, the researcher visits Federal University of Technology, Owerri (FUTO), Pure Water Factory to collect some sample and data on the process of water purification and Neutralization Starting from the point of water entering into the Tank from the Water borehole to the final Neutralization process"

Stage II: At this phase, Water from the bore hole is stored inside the water storage tank.

Stage III: At this phase, the power machine is used to increase the pressure of the water entering into the water storage tank inside of the factory

Stage IV: At this phase, the micro thread filter is used to further carryout purification process "

Stage V: At this phase, the ultraviolet water stablizer provides 95% purification of the water before the water can be moved to the next phase"

Stage VI: At this phase, the water now enters inside the three-in-one machine, which has the ability to wash, fill water, and cap water battles.

Stage VII: At this phase, the battle waters after filling and capping by the 3-in-1 machine, then label and package the water battles for distribution and use.

In the Water treatment production Process the result obtained shows the accepted level of NAFDAC approved from 6.8 to 7.2 level of pH Neutralization Process. The result shows in figure 4.15.

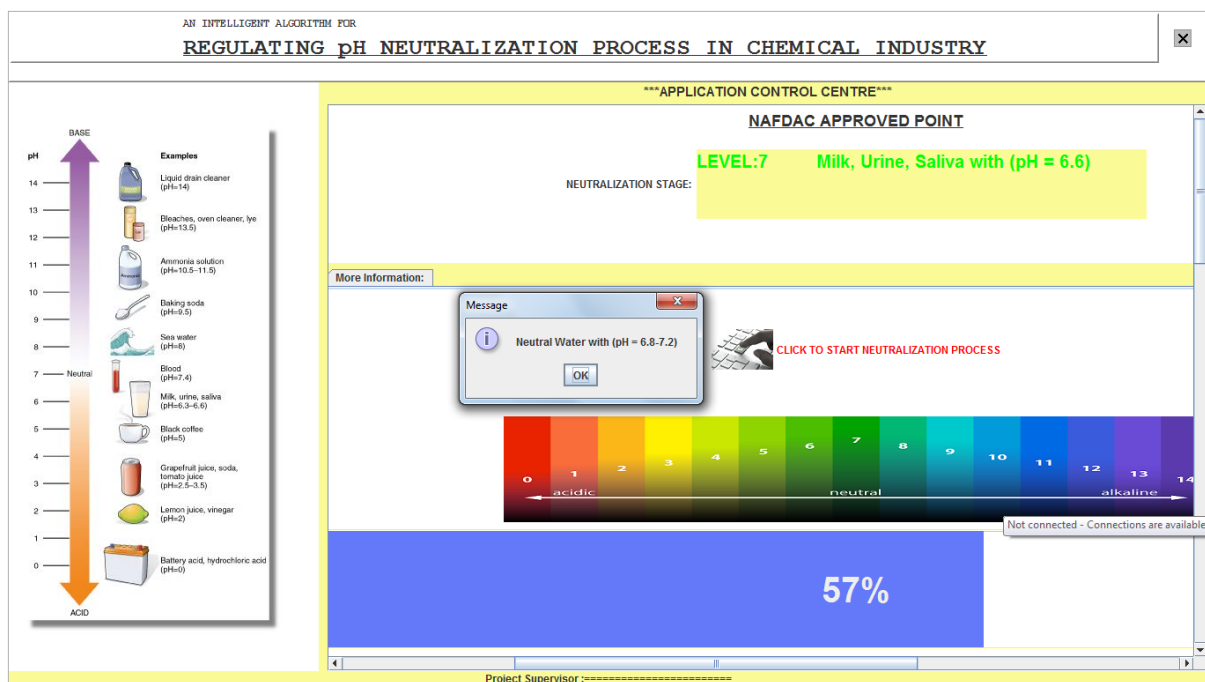


Figure 4.15: Neutralization Process of Water Treatment Plant

The application scanned through the pH neutralization meter and continue to search for the level of approved standard for production process if found the application move the value to the production for more simulation in order to satisfied the production requirement.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

This research work has presented an application of intelligent model in performance improvement of industrial process for pH regulation process. A mathematical model of a pH neutralization process for wastewater management was obtained. The mathematical equations of pH neutralization process were further modelled in MATLAB/Simulink environment to study the dynamic performance response of the process when no computational algorithm has been introduced. Simulation was initially carried out at the uncompensated condition. The result showed that step response performance to unit step input was largely unsatisfactory as the response obtained was far less than the desired pH value. Hence, a computational algorithm for a compensator that combines proportional integral and derivative (PID) actions was developed using the MATLAB/Simulink PID tuning tool. Simulation was conducted at the time a PID computational compensation algorithm was introduced to enhance the process. The result obtained indicated improvement from the uncompensated process with the PID compensator ensuring that the output attained the desired pH value with respect to unit step input. However, the PID performance needed to be improved with respect to the overshoot performance. A compensator algorithm that combined PID computational gains and Fuzzy Logic intelligent algorithm called Fuzzy-PID was proposed. A Fuzzy Inference System (FIS) of Fuzzy Logic tool of MATLAB/Simulink software was used to develop a Fuzzy Logic algorithm. The PID gains were integrated with the developed Fuzzy Logic algorithm to optimize the performance of pH process. With the introduction of the Fuzzy-PID compensation algorithm into the pH process, simulation was conducted and the result indicated that the proposed compensation technique improved the response

performance of pH process with satisfactory overshoot and adequate stability. Since the Fuzzy-PID compensator was able to attain the desired value in terms of a step response, a validation simulation was carried out to ascertain the performance of the proposed system in different pH media. The results of the simulations conducted in the three media namely, acid neutral and base, indicated that the Fuzzy-PID technique ensured that the desired pH value was attained in each of the media respectively.

Generally, the effects of introducing Fuzzy-PID compensation technique into the system were assessed based on time domain performance parameters using MATLAB as simulation tool. The obtained results from the system step response and validation simulation experiments showed remarkable improvement in the performance of the Fuzzy-PID compensated system over PID compensated and the uncompensated system with efficient attainment in desired pH value. Hence, using this system as a part of a pH neutralization process will ensure that good wastewater treatment is achieved at every desired pH value.

5.2 Recommendations

The research work has studied different processes in enhancement of the compensators this is capable of achieving a given level of performance when used in automated process, some improvement and further research directions are recommended below:

1. In this research, a compensated pH process has been tested using computer simulation. However, further work can be carried out by checking the performance capacity of the Fuzzy-PID through hardware implementation by developing a prototype.

2. Uncertain disturbance or changes can be introduced into the compensated pH process to ascertain the ability of the proposed system to maintain the pH accurately.
3. A combination of MATLAB and embedded programming or coding can be carried out to make sure that the system becomes fully automated to the core.

5.3 Contributions to Knowledge

Having studies extensively on several work done by authors, this research developed a Fuzzy-PID compensated model for pH regulation process, the following distinctive features makes the present work unique:

1. A hybrid dynamic model called Fuzzy-PID has been developed to enhance the pH regulating process that handles nonlinearities and uncertainties, among variables.
2. The research developed a discrete-continuous real time stimulation and application system for the improvement of industrial based processes for better performance.

REFERENCES

- Aboelela M.A.S., Hennas, R.H.M., and Dorrah, H.T. (2015). Design and Implementation of a PID Controller for a Continuous Stirred Tank Reactor (CSTR) System using Particle Swarm Algorithms Conference 1-12
<https://www.researchgate.net/publication/281068326> accessed on
- Ang G. J. N., Wei Z. L and Lim C. M. (2018). A Proposed Two-Input-Output Self-turning Control Scheme. 2018 IEEE Symposium on Computer Application & Industrial Electronic (ISCAIE), 102-107
DIO: 10.1109/ISCAIE.2018.8405452,
- Alejandro, R.M., Rosano, C.M.P., and Juan, B.G. (2010). Problem Based Learning (PBL): Analysis of Continuous Stirred Tank Chemical Reactors with a Process Control Approach. International Journal of Software Engineering and Application 1(14), 54-73
- Alshammari O., Muhammad N. and Mahyuddin H. H. (2018) A survey on Control Techniques of a BenchMarked Continuous Stirred Tank Reactor. Journal of Engineering Science and Technology 13, 10, 3277-3296
- Ahmed, S. I.,(2011) Modified Mathematical Model for Neutralization System in Stirred Tank Reactor, *BCREC, ISSN* , 1978-2993, 2011.
- Antsaklis, P. J., and Koutsoukos, X. D., (2003). Hybrid systems: review and recent progress, in *Software-enabled Control: Information Technology for Dynamical Systems* edited by Samad, T., and Balas, G., IEEE Press : Wiley-Interscience, 256-268.
- Allwin, S., Biksha, S.N., Abirami, S., Kala, H. and Udhaya A.P. (2014). Engineering Application of Artificial Intelligence in Electrical Engineering International. Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 3(9), 11934-11941.

- Alwan, G.M. (2008). pH-Control Problems of Wastewater Treatment Plants. *Al-Khwarizmi Engineering Journal*, 4(2), 37- 45.
- Ahn, H., Choi, C., and Kim, K., (1993)“Iterative learning control for a class of nonlinear systems, *Automatica*, 29, 6,1575-1578.
- Aitbayevich, S. B. Alkhaidarovna S.L. and Batyrbekovich S. A. (2015) Intelligent and Hybrid Systems of Process Control: Theory, Methods and Applications. *Mediterranean Journal Social Sciences*, 6(1S1),627. DIO:10.5901/mjss.2015.v6n1s1p627
- Artin, H. and Remoundaki, E.(2018). Bioreactors for Metal Bearing Waste, Treatment (online). Available and access on: 14th February, 2018 <http://www.metal.ntua.gr/orpkousi/e-learning/bioreactors/page 04.htm>
- Artur, W. (2016). Modeling and Control of Continuous Stirred Tank Reactor with PID Controller. *Ecological Engineering*, 49, 195-201.
- Antonelli R. and Astolfi A. (2003) Continuous Stirred Tank Reactors: Easy to Stabilise. *Automatica* 39, 1817-1827
- Anna V. and Alajos M. (2022). Cascade Fuzzy Control of Tubular Chemical Reactor. *Computer Aided Chemical Engineering*, 51(12), 1021-1026 <https://doi.org/10.1016/B978-0-323-958879-0.5017-5>
- Bequette, B.W. (2003). *Process Control: Modelling, Design and Simulation*, Prentice Hall Professional.
- Bharathi S. K, Radheysham A., Rajakeelpakkam C. (2013). Continuous Stirred Tank Reactor Based Blender Controlled Application using PLC *International Journal of Engineering Research & Technology*. 2(2), 1-6.
- Bequette, B. W. (1991). Nonlinear Control of Chemical Processes: A Review. *Industrial and Engineering Chemistry*, 30, 1391-1413

- Bustamante J. A., Castaneda M., Puebla H. and Martine E. H.(2017) Robust Cascade Control for Chemical Reactors: An Approach based on Modelling Error Compensation. International Journal of Chemical Reactor Engineering. 15(6), 20170082
<https://dio.org/10.1515/ijcre-2017-0082>
- Chen, C. T., & Chang, W.D. (2009). A feed forward neural network with Function Shape Auto Tuning. Neural Networks, 9, 627–641.
- Chopra, V., Singla, S.K., and Dewan L. (2014). Comparative Analysis of Tuning a PID Controller Using Intelligent Methods. Aota Polytechnica Hungarica, 11(8), 235-249
- Danish, M., Almesfer, M. K. and Rashid, M. (2015). Effect of Operating Conditions on CSTR performance: An Experimental Study. International Journal of Engineering Research and Applications, 5(2), 74-78.
- D'Azzo, J. J., Houpis, C. H., and Sheldon, S. N.(2003) Linear Control System Analysis and Design with MATLAB, CRC, 72-91
- Darius, M., and Sivagmasundari, S. (2016). Design and implementation of Controller for a CSTR Process. International Journal of Emerging Technology in Computer Science and Electronics 23(1), 175-183
- Dostal, P., Jiri, V. and Vladimir B. (2015) Simulation of Adaptive Control of a Continuous Stirred Tank Reactor. Proceedings of the 23rd European Conference on Modelling and Simulation @ECMS Javier Otamendi, Andrzej Bargiela Jose luis 9-6
- Deepa, S., Anipriya, N., and Subbulakshmy, R. (2015). Design of Controllers for Continuous Stirred Tank Rector. International Journal of Power Electronics and Derive System, 5(4), 576-582.

- Debnath, S., and Tripathi, V.K. (2016). A case study to optimized Design in linearization of Non-linear CSTR Using Multiple Model Predictive Control Approach. *International Journal of Engineering Science and Computing*, 6(7), 8682-8686
- Diarmuid O. D. and Donald E. Ramirez (2016) "Mitigation Collinearity in linear regression models using ridge, Surrogate and Raised Estimator, *Cogent Mathematics*, 3:1, 1144697, DOI: 10.1080/23311835.2016.1144697, 1-9
- Farzin P., Mohammad R. Rashidian, M. S. and Sadeq A. (2012) Effect of Rule Base on the Fuzzy-Based Tuning Fuzzy Sliding Mode Controller: Applied to 2nd Order Nonlinear System, *International Journal of Advanced Science and Technology* 46(9), 39- 70
- Gao, R., O'Dwyer, A., and Coyle E., (2003). Model Predictive Control of CSTR Based on Local Model Networks *Proceedings of the Inshi Signals and Systems Conference*, 397-402
- Ghatule, M.P. (2017). Control of Neutralization Process in Continuous Stirred Tank Reactor (CSTR). *International Journal of Research in Advent Technology*, 5(12), 24-28.
- Hoang N. H., Couenne F., Gorrec Y.L., Chen C. L and Ydstie B. E (2013). Passivity- Based Nonlinear Control of CSTR via asymptotic Observer. Author Manuscript, Published in *Annual Review on Control*. 1-33
- Iannic, A. and Rossi, N. (2021) SIR-Proportional Integer Derivative Controller for COVID-19 Outbreak Containment. *Physics*, 3, 459-472 <https://doi.org/10.3390/physics 3030031>
- Ikutegbe C. A., Long O. E., Alege M.O., Oduola M.O., and Adetunji A.R (2015) Modeling Analysis of Stirred Tank Reactors. *Current Journal of Applied*

Science and Technology, 9(2) 191-199.
<https://dio.org/10.9734/BJAST/2015/16685>

- Ishaq M. N. and Ingmar P. (2001). Dynamic Factor Demand Models and Productivity. A Chapter in New Developments in Productivity Analysis 103-172
- Jasvir K. and Gursewak S. B. (2017) Review on PID Controller with Intelligent System, International Journal of Advance Research, Ideas and Innovations in Technology: 12(5), 75-79
- Jose, T., Antony, R., and Isaac, S. (2013). pH Neutralization in CSTR Using Model Reference Neural Network and Fuzzy Logic Adaptive Controlling Schemes. International Journal of Advancements in Research & Technology, 2(1), 1-6.
- Jorge A. Romero- Bustamante Jazeal G. Moguel-Castaneda Hector Pueblea Eliseo Hernandez-Martinez (2017). Control en cascada de reactores químicos vía compensación de error de modelado. International Journal of Chemical Reactor Engineering (15)6, 1- 15
- Jorge A. Miguel R. Carlos C. and Vicente J. (2013) "MDD-Approach for Developing Paervative Systems Basced on Service-Oriented Multi-Agent Systems. ADCAIL: Advances in Intelligence Journal, Salananca, 2(3), 1154-1163
- Jarinah M. A., Hussain M.A., Moses O.T., and Jie Z. (2015) Artificial Intelligence Techniques applied as estimator in Chemical Process Systems; Expert Systems with Application- Journal homepage:www.elsevier.com 2015 5916-5931
- Jhon, E.R., Victor, H.G. and Jorge, E.C. (2018). A Tuning Proposed for direct fuzzy PID Controllers Oriented to industrial Continuous Processes.

Preprints of the 3rd IFAC Conference on Advances in Proportional –
Integral Derivative Control

Jimisha K. and Shinu M.M. (2016) Analysis of Conventional Controllers for Pressure rated Modified CSTR System. International Advanced Research Journal in Science, Engineering and Technology. 3(6), 274-277. DIO:10.17148/IARJSET.2016.3653

Malhotra R. and Sodhi A. (2017) Boiler flow control Using PID and Fuzzy Logic Controller. International Journal of Computer Science and Engineering Technology 1(6), 315-319

Mansouri, D., and Rezaie, B. (2012). Nonlinear Model for Controlling CSTR System. 3rd International Conference on Research and innovation in Computer, electronics and manufacturing Engineering (RICEME- 17), 121-123.

Muqheet, M. A., Najeeq, M. A., Akbar F. and Ali Sadra (2015) “PLC and SCADA Based Control of Continuous Stirred Tank Reactor (CSTR). International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering, 3(12), 185-189
DIO 10.17148/IJIREEICE.2015.31238.

Manikandan, P., Geetha, M., Jubi, K., and Jerome, J. (2013). Fault Tolerate Fuzzy Gain Scheduling Proportional- Integral Derivative Controller for Continuous Stirred Tank Reactor. Australian Journal of Basic and Applied Sciences, 7(13), 84-93.

Maitraye, S., Ravendra, S., and Robit, R. (2004). A Hybride MPC-PID Control System Design for the Continuous Purification and Processing of Active Pharmaceutical Ingredients Processes, 2 392-418.

- Maheshwan, N. Jain, N., Jingar, A., and Suthehar M. (2016). Design and Analysis of PID Controller for CSTR Process. *International Journal of Science Engineering and Technology Research* 5(2), 600-603.
- Manoj V., Varunkumar S., Manoranjith R. and Radhik V. (2015) Control and Monitoring of pHProcess using Fuzzy Based Controller. *International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Enigneering.* 3(1), 119-122
DIO:10.17148/IJREEICE.2015.3125.
- Ming, L., and Zhao, J. (2017). Review on Chemical Process Fault Detection and Diagnosis. 6th International Symposium on Advanced Control of Industrial Processes 457-462.
- Maayedukondalu, T. S. (2015), Steady State Simulation of Continuous Stirred Tank Reactor (CSTR) System Using Aspen. Biju Patnalk Central Library National Institute of Technology Rourkela, Odisha-769008, 1-46
- Mujahed .I.A. D. (2022). Construction and Evaluation of a Control Mechanism for Fuzzy Fractional-Order PID. *Applied Sciences*, 12, 1-12, 6832. <https://doi.org/10.3390/app12146832>
- Nabi, M.M. (2015). Neural Model Based Advanced Control of Chylla-Haase Reactor. A dissertation for award of degree of Doctor in Philosophy, Doctoral thesis, Liverpool John Moores University..
- Navid Y., Mehdi S., Masoud A., and Saeed P. A., (2015) Design of Nonlinear CSTR Control System Using Active Disturbance Rejection Control Optimized by Asexual Reproduction Optimization. *Science and Education Publishing from Scientific Research Knowledge*, (3), 2, 24364-24375

- Nemani K.M.P (2015). Simulation of Continuous Stirred Tank Reactors (CSTR) Using Orthogonal. Research Thesis, Department of Chemical Engineering, National Institute Technology, Rourkela-769008, 123-136
- Kanishna D., Suryanarayana, K., Aparana, G., and Padmasree, R. (2012). Tuning of PID Controllers for Unstabled Continous Stirred Tank Reactors. International Journal of Applied Science and Engineering, 10(1), 1-18.
- Kambale, S. D., George S., and Zope R.G. (2015). Controllers used in pH Neutralization Process: A review. inters. J. Eng. Technol., 2:2667-2670.
- Kaur A. and Kaur A. (2012) Comparison of Fuzzy Logic and Neuro Fuzzy Algorithms for Air Condition System. International Journal of Soft Computing and Engineering(IJSCE) ISSN: 2231-2307, 2(1), 417-420
- Karl, B. Anders R. and Erik K. (2014) "Rao-Blackwellized Particle filters with out-of-sequence Measurement Processing: IEEE Transactions on Signal Processing, 62(24) 6454-6467
- Kennedy J. and Eberhart (1995) "Particle Swarm Optimization" in Proc. IEEE int. Conf. Neural Networks, Perth, Australia 4, 1942-1948
- Kumar, V., Rana, K.P.S. and Sinha, A.K. (2011). Design Performance, and Stability Analysis of a Formula-Based Fuzzy PI Controller. International Journal of Innovation Computing, Information and control, 7(7B), 4291-4308.
- Khanduja, N. (2015). CSTR Control by using Model Reference Adaptive Control and PSO International Journal of Mechanical, industrial and Aerospace Science, 8(12), 2144-2149

- Kishore B., Rosdiazli I., Mohd N. TranDuc C., and Sabo M.H., (2018) Optimal PID Control of pH Neutralization Plant. Arabian Journal for Science and Engineering 43(15), 2687–2701
- Krishna, D., Suryanarayana, K., Aparajitha, G., and Padmasree R. (2012). Tuning of PID Controllers for Continuous Stirred Tank Reactors. *Indian Chemical Engineer*, 54(3), 157-179
- Liu, J.F., Munoz de la Penia, D., and Christofides P.D. (2010). Distributed Model Predictive Control of nonlinear Systems subject to Asynchronous and delayed measurements. *Automatic*, 46(1), 52-61
- Luyben, W.L., and Tyreus, B.D., (1998). *Process Control*, New York, USA: McGraw. Hill pp 548-596
- Liberzon D. (2003) *System with Sensor or Actuator Constraints Control: Foundation and Application* Birkhauser, Boston, MA
<https://doi.org/10.007/978-14612-007,8-5>
- Lunze, J. (2000) *Process supervision by means of qualitative models*. Technische Universität Hamburg-Harburg, Germany, 1758-1762.
- Lloyds R. G., and Ahmed A. (2018) Modified Series Cascade Control Strategy for Integrating Processes. *Indian Control Conference (ICC)*, January 4-8, IIT Kanpur, India 252-257
- Oriabo, T.K., and Igbokwe, P.K. (2015). Proportional Integral Derivative Control (PID) of a Continuous Stirred Tank Reactor (CSTR). *The International Journal of Engineering and Science*, 4(10), 66-73.
- Ortega C., Colina E., and Chacon E. (2007) Intelligent supervisory control designs framework for fault exposed processes. *WSEAS, International*

Journal of Circuits, Systems and Signal Processing, Cambridge University, USA,1-4,

- Ojiabo T. K and Igbokwe P. K.(2015) Proportional Integral Derivative Control (PID) of Continuous Stirred Tank Reactor (CSTR). *International Journal of Engineering and Science (IJES)*. 4(10), 66-73
- Oumair, N. R, Atif, A. and Khan, R. (2013) Hybrid Fuzzy Logic and PID Controller for pH Neutralization Pilot Plant, *International Journal of Fuzzy Logic Systems (IJFLS)*, 13(3), 425-436
- Park D., Let T-L., Quynh N. V., Long N. K. and Hong S. K. (2021) Online Turning of PID Controller Using a Multilayer Fuzzy Neural Network Design for Quadcopter Attitude. Tracking Control. *Front. Neurobit* 14:1619935.dio
- Pathak, K., Markana, A., Parekh, N. (2010). Optimal Control of CSTR, *Electroanalytical Chemistry*, 3, 199-296
- Pablo, A. Lopez P., and Ricardo, A. L. (2009). Dynamic Non linear Feedback for Temperature Control of Continuous Stirred Reactor with Complex Behaviour. *Journal of Applied Research and Technology*, 7(2),202-217.
- Prabhu, K., and Bhastram, V.M. (2014). Controller Design for Continuous Stirred Tank Reactor Using Adaptive Control. *Research Journal of Applied Science*, 9(8), 489-495
- Poorarasan, J., Kayalvizhi, R. and Pongianan, R.K. (2014). Design of Fractional Order PID Controller for a CSTR Process. *International Referenced Journal of Engineering and Science*, 3(1), 08-14.

- Poongodi, P., and Sudhanan, R.M. (2015). Simulation of Temperature Control Methodologies for Chemical Reactor. *Journal of Chemical Pharmaceutical Research*, 7(9), 682-689.
- Prabhu, K., and Bhaskaran, V.M. (2014). Controller Design for Continuous Stirred Tank Reactor Using Adaptive Control. *Research Journal of Applied Sciences*, 9(8), 489-495
- Puren O. (2005) Hybrid Intelligent Machine Systems: Design, Modeling and Control, Thesis Submitted to the College of Graduate Studies and Research In Partial Fulfillment of the Requirements For the Degree of Doctor of Philosophy In the Department of Mechanical Engineering University of Saskatchewan Saskatoon, Canada 1-11.
- Rahmat, M.F., Yazdami, A.M., Movehad, and Mahmoud Z. S. (2011). Temperature Control of a Continuous Stirred Tank Reactor by means of Two Differential Intelligent Strategies, *International Journal of Smart Sensing Intelligent System*, (6)2, 244-267.
- Rakesh P. Borase, D. K. Maghade, and S.N Pawar (2020), A review of PID Control turning Methods and Application. *International Journal of Dynamics and Control* 9(6), 818-827
- Ramasubramanian M., and Thirumarimurugan M. (2022). Spider- Based FOPID Controller design for Temperature Control in Aliminium extraction Process 41, 493-504; <https://doi.org/10.1515/htmp.2022-0052>
- Ramli, L.,Izzuddin, M., Lazim H. I. and Jaafor Z. M.(2020). Modelling and Fuzzy Logic Control of Tower Crane System. *Applications of Modelling and simulation* (4), 1-11
- Ram, S.S., Kumar, D.D. and Meenakshipriya, B. (2016). Designing of PID Controllers for pH Neutralization Process. *Indian Journal of Science and Technology*, 9(12),1-5.

- Russell, S. and Norvig, P. (1995) Artificial Intelligence: a modern approach. Prentice Hall International Editions, Rich, E., & Artificial intelligence. London: International Student Edition, McGraw-Hill. 91-97
- Raju S. S., Harish P.K., Nikhila G., Doodeswar P. And Meghana (2021) Design and Implementation of Fuzzy PID Controller for Speed of Robotic Vehicle System in MATLAB. 2nd International Conference on IoT Based Control Networks and Intelligent Systems (ICICNIS 2021), 1-8
- Saad, M., Albagul A., and Obiad D. (2013) Modeling and Control of Continuous Stirred Tank Reactor System. <https://researchgate.net/publication/280039754>, 28/09/2013
- Seborg, D.E., Edgar, F.T., and Mellichamp, D.A. (2004) Process Dynamics and Control, John Willy and Sons, USA.
- Seraji H. (1998). Nonlinear and Adaptive Control of Force and Compliance in Manipulators. International Journal of Robotics Research, 17(5), 467-484
- Saravanan V. and Kumar R.B (2019) Mathematical modelling and Controller design using electromagnetic techniques for sugar industry process Received 08 Jan 2019, Accepted 29 Apr 2019, Published online: 16 Aug 2019 <https://doi.org/10.1080/00051144.2019.1653667>
- Sakthiya R.S., Dinesh K. and Meenakshipriya, B. (2016) Designing of PID Controllers for pH Neutralization Process Indian Journal of Science and Technology, 9(12), DOI: 10.17485/ijst/2016/v9i12/89940, March 2016, ISSN (Print) : 0974-6846, ISSN (Online) : 0974-5645
- Seki, H., and Naka, Y. (2008). Optimizing Control of CSTR/Distillation Column Processes with one Material Recycle Industrial and Engineering Chemistry Research, 47(22), 8741-8753.

- Seki H., Ooyama, S., and Ogawa, M. (2004). Nonlinear Model Predictive Control Using Successive Linearization – Application of Chemical Reactors. *Trans of the Society of Instrument and Control Engineers*, 3(1), 66-72.
- Soheilirad, M.S., Ghasab, M.A.J., Sefidgar S., and Saberian, A. (2012). Turning of PID Controller for Multi Area load Frequency control by using imperialist Competitive Algorithm. *J. Basic Applied Sci. Res.*, 2, 3461-3469
- Soukkou A., Khellaf., Leulmi S. and Boudeghdegh (2008). Optimal Control of a CSTR Process, *Brazilian Journal of Chemical Engineering*. 25(4), 799-812
- Stephanopoulos, G., and Han, C. (1996). Intelligent Systems in Process Engineering: A review. *Computers and Chemical Engineering*, 20, 743,791.
- Shrivatava P. (2012) Modeling and Control of CSTR Using Model based Neural Network Predictive Control. *Computer Science, Physic*, Corpus ID: 17552907
- Snehal D. K., Sebastain G. and Zope R.G. (2015). Controllers used in pH Neutralization Process: A Review. *International Research Journal of Engineering and Technology (IRJET)* 2(3), 354-361
- Stogested, S. (2004). Control Structure Design for Complete Chemical Plants. *Computer and Chemical Engineering*, 28 (1-2), 219-234.
- Sivan S., Filo O., and Siegelman H. (2007). Application of Expert Networks for Predicting Protein Secondary Structure in Artificial Intelligence. *Science Direct, Biomolecular Engineering* 24(2007), 237-234

- Suvire R., Serrano M., Vega J., Ortiz O., and Scagila G. (2017) Nonlinear Controller for Trajectory Tracking of a Continuous Tank Reactor. *Control Engineering and Applied Information* 19(4), 112-121
- Thulasidharan, S., Kavyarasan, K., and Bagyaveereswaran, V. (2017). Turning of PID Controller Using Optimization Techniques for a MIMO Process. *IPO Conference Series: Materials Science and Engineering*, 263, 052019.
- Thulasi D. S., Kavyarasan, K., and Bagyaveereswaran, V. (2017). Turning of PID Controller using Optimization Techniques for a MIMO Process. *IOP Conference Series: Materials Science and Engineering*, 263 052019, 1-16.
- Tippet, M.J., and Bao, J. (2015). Distributed Control of Chemical Process Networks. *International Journal of Automation and Computing*, 12(4), 368-381.
- Updadhay R. and Singla R. (2010) Analysis of Temperature Control with Adaptive and PID Controller (A Comparative Study). *International Journal of Engineering Technology*. 2(5), 1793-8236. DIO: 10.7763/IJET.2010.V2.164
- Upadhye P. A., Sebastina G. , Mobeen S. A (2015) Control of pH Neutralization Process Using Fuzzy logic: A review. *International Research Journal of Engineering and Technology* 2,(3)
- Vishnoi, V., Padhee, S., and Kaur, G. (2012). Controller Performance Evaluation for Concentration Control of Isothermal Continuous Stirred Tank Reactor. *International Journal of Scientific and Research Publications*, 2(6), 1-7

- Vojtesek J., and Dostal P.(2015). Nonlinear versus Ordinary Adaptive Control of Continuous Stirred- Tank Reactor, The Scientific World Journal, Article ID 389273, 10. <https://doi.org/10.1155/2015/389273>
- Vijayakumar, K., and Magnigandan, T. (2015). Design of Soft Computing Based Nonlinear Model Predictive Controller for the CSTR Process. World Engineering and Applied Sciences Journal 6(1), 32-40
- Xin-She, Y. (2021), Nature- Inspired Optimization Algorithm (Second Edition), Chapter 8, 111-121
- Yadav, I. and Trivedi, A. (2014). Temperature Control of CSTR a Comparative Approach, International Journal of Engineering Research and Reviews. 2(3), 200-208, ISSN 2348-699X
- Yong Z., Lijun Z. and Zhiliang D.(2019). MEA-Turning Method for Design of the PID Controller, Mathematical Problems in Engineering Article ID 1378783, 11 <https://doi.org/10.1155/2019/1378783>
- Zhao, D., Zhu, O., and Dubbeldam, J. (2015). Terminal Sliding Mode Control for Continuous Stirred Tank Reactor. Chemical Engineering Research and Design, 94, 266-274.
- Zhang S. and Wang X. (2013) "Study of Fuzzy-PID Control in MATLAB for Two-Phase Hybrid Stepping Motor, Proceedings of the 2nd International Conference on Systems Engineering and Modeling (ICSEM-13), 804-811

APPENDIX A

SOURCE CODES

```
import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Component;
import java.awt.Dialog;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Graphics;
import java.awt.GridBagConstraints;
import java.awt.GridBagLayout;
import java.awt.GridLayout;
import java.awt.Insets;
import java.awt.Toolkit;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;
import java.awt.event.FocusListener;
import java.awt.event.FocusEvent;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;
import static java.lang.Thread.currentThread;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.List;
import javax.swing.AbstractListModel;
import javax.swing.BorderFactory;
import javax.swing.DefaultListModel;
import javax.swing.ImageIcon;
import javax.swing.JButton;
```

```

import javax.swing.JComboBox;
import javax.swing.JDialog;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JTextArea;
import javax.swing.JList;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JProgressBar;
import javax.swing.JScrollPane;
import javax.swing.JTabbedPane;
import javax.swing.JTextField;
import javax.swing.ListCellRenderer;
import javax.swing.ListModel;
import javax.swing.border.BevelBorder;
import javax.swing.border.EtchedBorder;
import java.util.Random;
import java.util.Calendar;
import java.text.SimpleDateFormat;
import Com.iprosoft.tech.*;
/**
 *
 * @author DONATUS TECH
 */
public class Application extends JFrame implements Runnable{
    JPanel problempanel;
    JPanel causepanel,leftpanel,p1,p2,p3,p4,p30,p31,p32;
    JPanel solutionpanel ;
    JLabel cdr,cssr,ccsr,csfr,hosr,hofr,rscp,status,analizer;
    JTextArea status1,status2,status3,status4,status5,status6,status7;
    JProgressBar bar1;
    int i,j,k,l =0;
    double ik=0.0;
    Thread t1,t2,t3,t4;
    private JButton addButt,callbutt,databutt;
    public Calendar calenda;
    public SimpleDateFormat format;

```

```

    Thread t,tt,ttt,tttt;
JProgressBar bar,barr,barr,analyzer1,analyzer2,analyzer3,analyzer4,analyzer5,analyzer6,analyzer7;
JList list;
private JPanel searchpanel,addpanel;
private JTextArea searchtxt;
private JPanel np;
private JLabel info1Lbl,info2Lbl;
private final CPanel cp=new CPanel();
// private ApplicationController contr;
private JComboBox modulesCmb;
private JList TrainingList;
private JList staffList;
private JList leaveList;
    JButton up;
private JScrollPane scroll, scroll1;
private JTextArea txtcause, txtsol;
// JProgressBar bar = new JProgressBar();
Application(){
    super("NUETRILIZATION OF WATER");
    setIconImage(new ImageIcon(getClass().getResource("/newlog1.png")).getImage());
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setUndecorated(true);
    setSize(Toolkit.getDefaultToolkit().getScreenSize());
    setLayout(new GridLayout(1,1));
    getRootPane().getContentPane().setBackground(new Color(250,250,150));
    info2Lbl=new JLabel();
    JButton exit=new JButton(new ImageIcon(getClass().getResource("/Exit.png")));
    exit.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent e){
            System.exit(0);
        }
    });
    exit.setBorderPainted(false);
    exit.setContentAreaFilled(false);
    np=new JPanel(new BorderLayout(1,10)){
        public void paintComponent(Graphics g){
            super.paintComponent(g);

```

```

        // g.drawImage(new ImageIcon(getClass().getResource("/bg.png")).getImage(), getWidth()-
503, 0, this);
    }
    public Dimension getPreferredSize(){
        return new Dimension(100,80);// upper size of the app. the place where the banner in placed
    }
};
//np.setBounds((int)getBounds().getX(),(int) getBounds().getY(),np.getWidth() , np.getHeight());
//np.setOpaque(true);
addpanel=new JPanel(new BorderLayout());
JButton about=new JButton(new ImageIcon(getClass().getResource("/lo.jpg")));
about.setBorderPainted(false);
about.setContentAreaFilled(false);
about.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){
        JDialog dialog=new JDialog();
        dialog.setSize(350, 220);
        dialog.setLayout(new GridLayout(6,1));
        dialog.setLocationRelativeTo(null);
        dialog.setIconImage(Application.this.getIconImage());
        dialog.setTitle("DONTEX TECHNOLOGIES NIG.");
        JLabel l=new JLabel("DONTEX TECHNOLOGIES NIG.");
        l.setFont(new Font(Font.MONOSPACED,Font.BOLD,20));
        l.setForeground(new Color(255,255,255));
        JPanel p=new JPanel();
        p.setBackground(new Color(200,50,255));
        p.add(l);
        dialog.add(p);

        l=new JLabel("Mobile : +2348036824158");
        l.setFont(new Font(Font.MONOSPACED,Font.BOLD,15));
        l.setForeground(new Color(240,20,200));
        dialog.add(l);
        l=new JLabel("Email : njokudonatus1@gmail.com");
        l.setFont(new Font(Font.MONOSPACED,Font.BOLD,15));
        l.setForeground(new Color(240,20,200));
        dialog.add(l);
    }
});

```

```

        l.setFont(new Font(Font.MONOSPACED,Font.BOLD,15));
        l.setForeground(new Color(240,20,200));
        dialog.add(l);
        l.setFont(new Font(Font.MONOSPACED,Font.BOLD,10));
        l.setForeground(new Color(240,20,200));
        dialog.add(l);
        p=new JPanel();
        p.setBackground(new Color(200,50,255));
        dialog.add(p);
        dialog.getContentPane().setBackground(new Color(240,240,100));
        dialog.setVisible(true);
    }
});

```

```

JLabel lbl=new JLabel("<html>AN INTELLIGENT ALGORITHM FOR<div align=\"
center\"><b><u><span style=\"font-size:20px;font-weight:bold;\"> REGULATING pH
NEUTRALIZATION PROCESS IN CHEMICAL INDUSTRY</span></u></div></html>");

```

```

    lbl.setOpaque(true);
    lbl.setFont(new Font(Font.MONOSPACED,Font.BOLD,12));
    lbl.setHorizontalAlignment(JLabel.CENTER);
    lbl.setBackground(new Color(255,255,255));
    lbl.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));

```

```

JPanel p=new JPanel(new BorderLayout());
p.setBackground(new Color(255,255,255));
//p.add(about,BorderLayout.WEST);
p.add(lbl,BorderLayout.CENTER);
p.add(exit,BorderLayout.EAST);

```

```

np.add(p,BorderLayout.NORTH);
JPanel lp=new JPanel(new GridLayout(4,1,1,5));
lbl=new JLabel("Name:NJOKU DONATUS");
lbl.setFont(new Font(Font.MONOSPACED,Font.BOLD,15));
lp.add(lbl);
lbl=new JLabel("MatNo:=====");
lbl.setFont(new Font(Font.MONOSPACED,Font.BOLD,15));
lp.add(lbl);

```

```

lbl=new JLabel("Department: COMPUTER SCIENCE");
lbl.setFont(new Font(Font.MONOSPACED,Font.BOLD,15));
lp.add(lbl);
lbl=new JLabel("Institution: FEDERAL UNIVERSITY OF TECHNOLOGY, OWERRI");
lbl.setFont(new Font(Font.MONOSPACED,Font.BOLD,15));
lp.add(lbl);
JPanel bannerpanel=new JPanel();
lp.setBackground(new Color(250,250,150));
// np.add(lp,BorderLayout.WEST);
np.setBackground(new Color(250,250,150));
np.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
JPanel pp=new JPanel(new BorderLayout());
pp.setBorder(BorderFactory.createEtchedBorder(EtchedBorder.RAISED));
pp.add(np,BorderLayout.NORTH);
pp.add(cp,BorderLayout.CENTER);
lbl=new JLabel("Project Supervisor :=====");
lbl.setHorizontalAlignment(JLabel.CENTER);
JLabel banner=new JLabel(new ImageIcon(getClass().getResource("/banners.jpg")));
pp.add(lbl,BorderLayout.SOUTH);
// np.add(banner,BorderLayout.EAST);
np.setBackground(new Color(255,255,255));
lp.setBackground(new Color(255,255,255));

add(pp);
setLocationRelativeTo(null);
txtcause=new JTextArea();
txtsol=new JTextArea();
list = new JList();
list.setBackground(new Color(200,200,125));

// list.add(jsp);
list.setFont(new Font(Font.MONOSPACED, Font.BOLD, 20));

up = new JButton("WATER NEUTRALIZATION PROCESS");
list.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
txtcause.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
txtsol.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));

```

```

p1.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
p2.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
p3.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
p4.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
list.setForeground(new Color(250,250,150));
    scroll = new
JScrollPane(txtcause,JScrollPane.VERTICAL_SCROLLBAR_ALWAYS,JScrollPane.HORIZONTAL
_SCROLLBAR_ALWAYS);
    scroll1 = new
JScrollPane(txtsol,JScrollPane.VERTICAL_SCROLLBAR_ALWAYS,JScrollPane.HORIZONTAL_S
CROLLBAR_ALWAYS);

//causepanel.add(mainbanner);
//solutionpanel.add(txtsol);

txtcause.setBackground(new Color(200,200,125));
txtsol.setBackground(new Color(200,200,125));
txtsol.setFont(new Font(Font.SANS_SERIF,Font.BOLD,30));
txtcause.setFont(new Font(Font.SANS_SERIF,Font.BOLD,30));
    problempanel.add(list);
    // for connection to database and list
    pp.add(addpanel,BorderLayout.SOUTH);
    //addpanel.add(bar,BorderLayout.NORTH);
    addpanel.add(lbl,BorderLayout.SOUTH);
    addpanel.setBackground(new Color(250,250,150));
    // up.setBackground(new Color(250,250,150));
    setVisible(true);
}
private class CPanel extends JPanel implements Runnable{
    private JPanel mainpan,p2_1,p2_2,p2_3,p2_4,p2_5,p2_6,p2_7,statuspan;

CPanel(){
    setLayout(new BorderLayout(10,2));
    setBackground(new Color(250,250,150));
    // contr=new ApplicationController();
    // modulesCmb=new JComboBox(contr.getAllStaff().toArray());
    p2_1=new JPanel();

```

```

p2_2=new JPanel();
p2_3=new JPanel();
p2_4=new JPanel();
p2_5=new JPanel();
p2_6=new JPanel();
p2_7=new JPanel();
statuspan=new JPanel(new GridLayout(2,1,10,10));
searchtxt= new JTextArea(3,30);
JTabbedPane tp=new JTabbedPane();
callbutt=new JButton("CALL");
databutt=new JButton("DATA PACKET");
JTextArea phonetxt=new JTextArea(3,30);

JLabel phonelabel= new JLabel("Enter Phone N0:");
phonelabel.setHorizontalAlignment(JLabel.CENTER);
phonelabel.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));

JLabel J= new JLabel("STATUS");

J.setHorizontalAlignment(JLabel.CENTER);
J.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
J.setForeground(Color.red);
cdr=new JLabel("Call Drop Rate (CDR)");
cssr=new JLabel("Call Setup Success Rate (CSSR)");
ccsr=new JLabel("Call Success Rate (CSR)");
csfr=new JLabel("Call Retainability (CR)");
hosr=new JLabel("HOSR");
hofr=new JLabel("HOFR");
rscp=new JLabel("RSCP");
JLabel call1=new JLabel("User 1:");
JLabel call2=new JLabel("User 2:");
JTextField call1txt= new JTextField(20);
JTextField call2txt= new JTextField(20);
status=new JLabel("STATUS");
analizer=new JLabel("ANALYZER");
status2=new JTextArea(3,30);
status3=new JTextArea(3,30);

```

```

status4=new JTextArea(3,30);
status5=new JTextArea(3,30);
status6=new JTextArea(3,30);
status7=new JTextArea(3,30);
status1=new JTextArea(3,30);

status2.setBackground(new Color(250,250,150));
status2.setEditable(false);
status1.setBackground(new Color(250,250,150));
status1.setEditable(false);
status3.setBackground(new Color(250,250,150));
status3.setEditable(false);
status4.setBackground(new Color(250,250,150));
status4.setEditable(false);
status5.setBackground(new Color(250,250,150));
status5.setEditable(false);
status6.setBackground(new Color(250,250,150));
status6.setEditable(false);
status7.setBackground(new Color(250,250,150));
status7.setEditable(false);

analyzer1 = new JProgressBar();
analyzer2 = new JProgressBar();
analyzer3 = new JProgressBar();
analyzer4 = new JProgressBar();
analyzer5 = new JProgressBar();
analyzer6 = new JProgressBar();
analyzer7 = new JProgressBar();
mainpan=new JPanel(new BorderLayout());
leftpanel=new JPanel(new BorderLayout());
p1=new JPanel(new BorderLayout());
p2=new JPanel(new GridLayout(1,2,10,10));
p30=new JPanel(new GridLayout(7,1));
p31=new JPanel(new GridLayout(7,1));
p32=new JPanel(new GridLayout());
p3=new JPanel(new BorderLayout());
p4=new JPanel(new BorderLayout());

```

```

problempanel = new JPanel(new BorderLayout());
causepanel = new JPanel(new BorderLayout());
solutionpanel = new JPanel(new BorderLayout());
mainpan.setBackground(new Color(255,255,255));
problempanel.setBackground(new Color(200,200,125));
causepanel.setBackground(new Color(200,200,125));
solutionpanel.setBackground(new Color(200,200,125));
leftpanel.setBackground(new Color(255,255,255));
p1.setBackground(new Color(250,250,150));
p2.setBackground(new Color(255,255,255));
p3.setBackground(new Color(250,250,150));
p4.setBackground(new Color(250,250,150));
JPanel phonepanel=new JPanel(new GridLayout(1,2));
add(mainpan,BorderLayout.CENTER);
//mainpan.add(problempanel);
//mainpan.add(causepanel);
//mainpan.add(solutionpanel);
JPanel p1_1=new JPanel(new GridLayout(2,1));
JPanel p1_2=new JPanel();
JLabel mainbanner=new JLabel(new ImageIcon(getClass().getResource("/ph1.jpg")));//center

```

image

```

mainpan.add(leftpanel);
leftpanel.add(mainbanner);

//leftpanel.add(p1, BorderLayout.NORTH);
//leftpanel.add(p2, BorderLayout.CENTER);
//leftpanel.add(p3, BorderLayout.WEST);
//leftpanel.add(p4, BorderLayout.SOUTH);

p1.add(p1_1, BorderLayout.NORTH);
p1.add(callbutt,BorderLayout.EAST);
// p1.add(databutt,BorderLayout.EAST);
p1_1.setBackground(new Color(250,250,150));
p1_1.add(call1);
p1_1.add(call1txt);
p1_1.add(call2);

```

```

p1_1.add(call2txt);
call1.setHorizontalAlignment(JLabel.RIGHT);
call2.setHorizontalAlignment(JLabel.RIGHT)
callbutt.addActionListener(new ActionListener(){// adding to database
    public void actionPerformed(ActionEvent ae){
        tttt.start();
    }
});
databutt.addActionListener(new ActionListener(){// adding to database
    public void actionPerformed(ActionEvent ae){
        tttt.start();
    }
});
analyzer1.getModel();
analyzer1.setForeground(new Color(0,250,250));
analyzer1.setBackground(Color.white);
analyzer1.setBorder(BorderFactory.createLineBorder(Color.red));
// bar.setBorder(BorderFactory.createCompoundBorder());
analyzer1.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
analyzer1.setBorder(BorderFactory.createEtchedBorder(Color.white, Color.yellow));
analyzer1.setStringPainted(true);
analyzer1.setMinimum(0);
analyzer1.setMaximum(100);
analyzer1.setFont(new Font(Font.SANS_SERIF,Font.BOLD,40));
analyzer2.getModel();
analyzer2.setForeground(new Color(0,250,250));
analyzer2.setBackground(Color.white);
analyzer2.setBorder(BorderFactory.createLineBorder(Color.red));
// bar.setBorder(BorderFactory.createCompoundBorder());
analyzer2.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
analyzer2.setBorder(BorderFactory.createEtchedBorder(Color.white, Color.yellow));
analyzer2.setStringPainted(true);
analyzer2.setMinimum(0);
analyzer2.setMaximum(100);
analyzer2.setFont(new Font(Font.SANS_SERIF,Font.BOLD,40));

analyzer3.getModel();

```

```

analyzer3.setForeground(new Color(0,250,250));
analyzer3.setBackground(Color.white);
analyzer3.setBorder(BorderFactory.createLineBorder(Color.red));
// bar.setBorder(BorderFactory.createCompoundBorder());
analyzer3.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
analyzer3.setBorder(BorderFactory.createEtchedBorder(Color.white, Color.yellow));
analyzer3.setStringPainted(true);
analyzer3.setMinimum(0);
analyzer3.setMaximum(100);
analyzer3.setFont(new Font(Font.SANS_SERIF,Font.BOLD,40));

analyzer4.getModel();
analyzer4.setForeground(new Color(0,250,250));
analyzer4.setBackground(Color.white);
analyzer4.setBorder(BorderFactory.createLineBorder(Color.red));
// bar.setBorder(BorderFactory.createCompoundBorder());
analyzer4.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
analyzer4.setBorder(BorderFactory.createEtchedBorder(Color.white, Color.yellow));
analyzer4.setStringPainted(true);
analyzer4.setMinimum(0);
analyzer4.setMaximum(100);
analyzer4.setFont(new Font(Font.SANS_SERIF,Font.BOLD,40));

analyzer5.getModel();
analyzer5.setForeground(new Color(0,250,250));
analyzer5.setBackground(Color.white);
analyzer5.setBorder(BorderFactory.createLineBorder(Color.red));
// bar.setBorder(BorderFactory.createCompoundBorder());
analyzer5.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
analyzer5.setBorder(BorderFactory.createEtchedBorder(Color.white, Color.yellow));
analyzer5.setStringPainted(true);
analyzer5.setMinimum(0);
analyzer5.setMaximum(100);
analyzer5.setFont(new Font(Font.SANS_SERIF,Font.BOLD,40));

analyzer6.getModel();
analyzer6.setForeground(new Color(0,250,250));

```

```

analyzer6.setBackground(Color.white);
analyzer6.setBorder(BorderFactory.createLineBorder(Color.red));
// bar.setBorder(BorderFactory.createCompoundBorder());
analyzer6.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
analyzer6.setBorder(BorderFactory.createEtchedBorder(Color.white, Color.yellow));
analyzer6.setStringPainted(true);
analyzer6.setMinimum(0);
analyzer6.setMaximum(100);
analyzer6.setFont(new Font(Font.SANS_SERIF,Font.BOLD,40));

analyzer7.getModel();
analyzer7.setForeground(new Color(0,250,250));
analyzer7.setBackground(Color.white);
analyzer7.setBorder(BorderFactory.createLineBorder(Color.red));
// bar.setBorder(BorderFactory.createCompoundBorder());
analyzer7.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
analyzer7.setBorder(BorderFactory.createEtchedBorder(Color.white, Color.yellow));
analyzer7.setStringPainted(true);
analyzer7.setMinimum(0);
analyzer7.setMaximum(100);
analyzer7.setFont(new Font(Font.SANS_SERIF,Font.BOLD,40));

p2.add(p30);
p2.add(p31);

p2_1.add(cdr);
p2_1.add(analyzer1);
p2_1.add(statuspan);
statuspan.add(J);
statuspan.add(status1);
p30.add(cssr);
p30.add(analyzer2);
p30.add(ccsr);
p30.add(analyzer3);
p30.add(csfr);
p30.add(analyzer4);
p30.add(hosr);

```

```

p30.add(analyzer5);
p30.add(hofr);
p30.add(analyzer6);
p30.add(rscp);
p30.add(analyzer7);

p31.add(status1);
p31.add(status2);
p31.add(status3);
p31.add(status4);
p31.add(status5);
p31.add(status6);
p31.add(status7);

add(new FrontPage(),BorderLayout.EAST);
setVisible(true);
ttt = new Thread(){
public void run(){
    try{

        for (i= 0; i<=100; i++){
            analyzer2.setValue(i);
            Thread.sleep(200);
            repaint();
            if (i ==30) {

                status1.setText(" Current Status 30%");
                status1.setForeground(Color.white);
                status1.setBackground(Color.red);
                status1.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
            }if(i ==56){
                status1.setText("Current Status 56% ");
                status1.setBackground(Color.blue);
                status1.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
                // Thread.sleep(500);
            }if(i==78){
                status1.setText(" Current Status 78%");

```

```

        status1.setBackground(Color.green);
        status1.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));

    }

}

for (i= 0; i<=100; i++){
    analyzer3.setValue(i);
    Thread.sleep(200);
    repaint();
    if (i ==30) {

        status2.setText(" Current Status 30%");
        status2.setForeground(Color.white);
        status2.setBackground(Color.blue);
        status2.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    }if(i ==56){
        status2.setText("Current Status 56%");
        status2.setBackground(Color.green);
        status2.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        // Thread.sleep(500);
    }if(i==78){
        status2.setText(" Current Status 78%");
        status2.setBackground(Color.green);
        status2.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));

    }

}

for (i= 0; i<=100; i++){
    analyzer4.setValue(i);
    Thread.sleep(200);
    repaint();
    if (i ==30) {

```

```

        status3.setText(" Current Status 30%");
        status3.setForeground(Color.white);
        status3.setBackground(Color.blue);
        status3.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    }if(i ==56){
        status3.setText("Current Status 56%");
        status3.setBackground(Color.green);
        status3.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        // Thread.sleep(500);
    }if(i==78){
        status3.setText(" Current Status 78%");
        status3.setBackground(Color.green);
        status3.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));

    }
}
for (i= 0; i<=100; i++){
    analyzer5.setValue(i);
    Thread.sleep(200);
    repaint();
    if (i ==30) {

        status4.setText(" Current Status 30%");
        status4.setForeground(Color.white);
        status4.setBackground(Color.blue);
        status4.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    }if(i ==56){
        status4.setText("Current Status 56%");
        status4.setBackground(Color.green);
        status4.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        // Thread.sleep(500);
    }if(i==78){
        status4.setText(" Current Status 78%");
        status4.setBackground(Color.green);
        status4.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));

    }
}
for (i= 0; i<=100; i++){

```

```

        analyzer6.setValue(i);
        Thread.sleep(200);
        repaint();
        if (i ==30) {

            status5.setText(" Current Status 30%");
            status5.setForeground(Color.white);
            status5.setBackground(Color.blue);
            status5.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        }if(i ==56){
            status5.setText("Current Status 56%");
            status5.setBackground(Color.green);
            status5.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
            // Thread.sleep(500);
        }if(i==78){
            status5.setText(" Current Status 78%");
            status5.setBackground(Color.red);
            status5.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));

        }
    }
    for (i= 0; i<=100; i++){
        analyzer7.setValue(i);
        Thread.sleep(200);
        repaint();
        if (i ==30) {

            status6.setText(" Current Status 30%");
            status6.setForeground(Color.white);
            status6.setBackground(Color.blue);
            status6.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        }if(i ==56){
            status6.setText("Current Status 56%");
            status6.setBackground(Color.green);
            status6.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
            // Thread.sleep(500);
        }if(i==78){
            status6.setText(" Current Status 78%");

```

```

        status6.setBackground(Color.gray);
        status6.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));

    }}

    } catch(InterruptedException ea){}

};

};

}

public void run() {
    throw new UnsupportedOperationException("Not supported yet.");
}
}

public class ApplicationInformation extends JPanel implements Runnable{
    public JButton addButt=new JButton("CLICK TO START PROCESSING",new
    ImageIcon(getClass().getResource("/add.png")));
    public JButton backButt=new JButton("Clear",new
    ImageIcon(getClass().getResource("/bk.png")));
    private JLabel progresstext=new JLabel("Wifi Network Undergoing Optimization Process.....It
    takes some time..");
    private JLabel networkname=new JLabel("Wifi Name/Location:");
    private JLabel signal=new JLabel("Optimized Wifi Signal Strength:");
    private JLabel datarate=new JLabel("Network Data Rate:");
    private JLabel latency=new JLabel("Wifi Latency Rate/Delay:");
    private JLabel bandwidth=new JLabel("Network BandWidth:");
    private JLabel noisefloor=new JLabel("Noise Floor Level:");
    private JLabel Time=new JLabel("Transmission Time:");

    private JButton view_result=new JButton("Click To View Results in Chart:");

    JLabel n1=new JLabel(new ImageIcon(getClass().getResource("/n1.jpg")));
    JLabel n2=new JLabel(new ImageIcon(getClass().getResource("/h1.jpg")));
    JLabel n3=new JLabel(new ImageIcon(getClass().getResource("/n3.jpg")));
    String info="pH is a determined value based on a defined scale, "
    + "similar to temperature. This means that pH "

```

+ "of water is not a physical parameter that \n "
 + "can be measured as a concentration or in a "
 + " quantity. Instead, it is a figure between"
 + " 0 and 14 defining how acidic or basic a \n "
 + "body of water is along a logarithmic scale¹. "
 + "The lower the number, the more acidic "
 + "the water is. The higher the number, \n "
 + "the more basic it is. A pH of 7 is "
 + "considered neutral. The logarithmic scale "
 + " means that each number below 7 is 10 times \n "
 + "more acidic than the previous number when "
 + "counting down. Likewise, when counting "
 + " up above 7, each number is 10 times \n "
 + "more basic than the previous number² "
 + "The logarithmic scale of pH means that "
 + "as pH increases, the H⁺ concentration will"
 + " decrease by a power of 10. Thus at a pH of 0,"
 + " H⁺ has a concentration of 1 M. At a pH of 7,\n "
 + "this decreases to 0.0000001 M. At a pH of 14,"
 + " there is only 0.00000000000001 M H⁺.\n" +
 "pH stands for the “power of hydrogen”³. The numerical value of pH is "
 + "determined by the molar concentration of\n "
 + " hydrogen ions (H⁺)³. This is done by "
 + "taking the negative logarithm of the H⁺ "
 + "concentration (-log(H⁺)). For example, if"
 + " a solution has a H⁺ concentration of 10⁻³ M,\n "
 + " the pH of the solution will be -log(10⁻³), "
 + "which equals 3.\n" +
 "This determination is due to the effect of "
 + "hydrogen ions (H⁺) and hydroxyl ions (OH⁻) on pH.\n "
 + "The higher the H⁺ concentration, the lower the pH, "
 + "and the higher the OH⁻ concentration, the higher the pH."
 + " At a neutral pH of 7 (pure water), the concentration of\n "
 + "both H⁺ ions and OH⁻ ions is 10⁻⁷ M. Thus the ions H⁺ and OH⁻ "
 + "are always paired – as the concentration of one increases, \n "
 + "the other will decrease; regardless of pH, the sum of "
 + "the ions will always equal 10⁻¹⁴ M². Due to this influence, \n "

```

        + "H+ and OH- are related to the basic definitions of acids and bases."
        + "Acid-base pairs can neutralize each other like H+ and OH- do \n"
        + "in this equation.\n";
private JTextField signalratio=new JTextField(10);
// private JTextField networkname=new JTextField(10);
private JTextField networktype=new JTextField(10);
private JTextField networknametxt=new JTextField(10);
public JTextField signaltxt=new JTextField(10);
private JTextArea dataratetxt=new JTextArea();
private JTextField latencytxt=new JTextField(10);
private JTextField timetxt=new JTextField(10);
private JTextField bandwidthtxt=new JTextField(10);
private JTextField noisefloortxt=new JTextField(10);
// private int size=contr.getAllStaff().size();

public ApplicationInformation(){
    setLayout(new BorderLayout(1,5));

    networkname.grabFocus();
    addButt.setBorderPainted(false);
    addButt.setContentAreaFilled(false);

    addButt.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent e){
            }
    });

    backButt.setBorderPainted(false);
    backButt.setContentAreaFilled(false);
    view_result.setBorderPainted(false);
    view_result.setContentAreaFilled(false);
    backButt.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent e){
            networktype.setText("");
            networkname.setText("");
            }
    });
}

```

```

JPanel data=new JPanel(new GridLayout(3,1));
JPanel data1=new JPanel(new BorderLayout(1,1));
JPanel data2=new JPanel(new GridLayout(2,1));
JPanel data3=new JPanel(new GridLayout(2,1));
JPanel data3_1=new JPanel(new GridLayout(1,7));
JPanel data3_2=new JPanel(new GridLayout(1,7));
data.setBackground(new Color(255,255,255));
    barr = new JProgressBar();
    JLabel result= new JLabel("");

addButt.addActionListener(new ActionListener(){// adding to database
    public void actionPerformed(ActionEvent ae){
        tt.start();
    }
});
add(data,BorderLayout.CENTER);

progresstext.setHorizontalAlignment(JLabel.CENTER);
progresstext.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
progresstext.setForeground(Color.green);
addButt.setForeground(Color.red);
networkname.setForeground(Color.BLUE);
signal.setForeground(Color.BLUE);
datarate.setForeground(Color.BLUE);
latency.setForeground(Color.BLUE);
bandwidth.setForeground(Color.BLUE);
Time.setForeground(Color.BLUE);
noisefloor.setForeground(Color.BLUE);
networkname.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
signal.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
datarate.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
latency.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
bandwidth.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
Time.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
noisefloor.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
networkname.setHorizontalAlignment(JLabel.CENTER);
signal.setHorizontalAlignment(JLabel.CENTER);

```

```

datarate.setHorizontalAlignment(JLabel.CENTER);
latency.setHorizontalAlignment(JLabel.CENTER);
bandwidth.setHorizontalAlignment(JLabel.CENTER);
Time.setHorizontalAlignment(JLabel.CENTER);
noisefloor.setHorizontalAlignment(JLabel.CENTER);
data2.setBackground(new Color(255,255,255));
data3_1.setBackground(new Color(255,255,255));
data3_2.setBackground(new Color(255,255,255));
data1.setBackground(new Color(255,255,255));
// networknametxt.setBackground(new Color(255,255,255));
networknametxt.setEditable(false);
signaltxt.setEditable(false);
dataratetxt.setEditable(false);
latencytxt.setEditable(false);
bandwidthtxt.setEditable(false);
timetxt.setEditable(false);
noisefloortxt.setEditable(false);

networknametxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
signaltxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
dataratetxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
latencytxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
bandwidthtxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
timetxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
noisefloortxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));

dataratetxt.setBackground(new Color(200,200,200));
networknametxt.setBackground(new Color(200,200,200));
signaltxt.setBackground(new Color(200,200,200));
latencytxt.setBackground(new Color(200,200,200));
bandwidthtxt.setBackground(new Color(200,200,200));
timetxt.setBackground(new Color(200,200,200));
noisefloortxt.setBackground(new Color(200,200,200));

dataratetxt.setForeground(new Color(0,0,0));
networknametxt.setForeground(new Color(0,0,0));
signaltxt.setForeground(new Color(0,0,0));

```

```

latencytxt.setForeground(new Color(0,0,0));
bandwidthtxt.setForeground(new Color(0,0,0));
timetxt.setForeground(new Color(0,0,0));
noisefloortxt.setForeground(new Color(0,0,0));
JTextArea ar= new JTextArea(info);
    ar.setFont(new Font(Font.SANS_SERIF,Font.BOLD,15));
data.add(data2);
data2.add(ar);
data2.add(n1);
data1.add(view_result, BorderLayout.WEST);
    barr.getModel();
barr.setForeground(new Color(0,250,250));
barr.setBackground(Color.white);
barr.setBorder(BorderFactory.createLineBorder(Color.red));
// barr.setBorder(BorderFactory.createCompoundBorder());
barr.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
barr.setBorder(BorderFactory.createEtchedBorder(Color.white, Color.yellow));
barr.setStringPainted(true);
barr.setMinimum(0);
barr.setMaximum(100);
barr.setFont(new Font(Font.SANS_SERIF,Font.BOLD,40));
    setVisible(true);
    tt = new Thread(){
public void run(){
    try{
        for (i= 0; i<=100; i++){
            barr.setValue(i);
            Thread.sleep(200);
            repaint();
            if (i == 100){
                double Signalmulti=(double)(10.0*Math.random())-5.00;//Signal Strength
multiplying factor(booster)
                double TransmitRate=225.00;//Transmitting multiplying factor
                double ReceivRate=225.00;//Receiving Multiplying factor
                double signalstrength=(double)(10.0*Math.random())-63.00+Signalmulti;//
                double datrate=(double)(10.0*Math.random()+1000.00;
                double recDatarate=(double)(10.0*Math.random()+821.00+ReceivRate;

```

```

double TranmitDatarate=(double)(10.0*Math.random()+732.00+TransmitRate;
double latency=(double)(10.0*Math.random()-23.00;
double bandwidth=(double)(2*Math.random()+2;
double noisefloor=(double)(10.0*Math.random()-83.00;//

signaltxt.setText(""+ signalstrength +" dBm");
signaltxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));

String status="WIFI OPTIMIZED";
searchtxt.setText(""+ signalstrength +" dBm"+"\\n"+"STATUS:"+status);
searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
searchtxt.setForeground(Color.red);
dataratetxt.setText("Transmit:" + TranmitDatarate +" bps" +"\\n"+"\\nReceiving:" +
recDatarate +"bps" );
dataratetxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
latencytxt.setText(""+ latency +"Mbps");
latencytxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));

bandwidthtxt.setText(""+ bandwidth +" GHz");
bandwidthtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
noisefloortxt.setText(""+ noisefloor +"Noise Floor");
noisefloortxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));

networknametxt.setText(""+ natworkname +"\\nLocation: "+"Along Sameke
Road");
networknametxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));

calenda= Calendar.getInstance();
format=new SimpleDateFormat ("HH: mm: ss");
timetxt.setText(format.format(calenda.getTime()));
timetxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
    }
}
} catch(InterruptedException ea){}

};
};
}

```

```

public void run() {
    throw new UnsupportedOperationException("Not supported yet.");
}
}

public class ApplicationController extends JPanel implements Runnable{
    public JButton addButt=new JButton("CLICK TO START NEUTRALIZATION PROCESS",new
ImageIcon(getClass().getResource("/d.jpg")));
    public JButton backButt=new JButton("Clear",new
ImageIcon(getClass().getResource("/bk.png")));
    private JLabel progresstext=new JLabel("Wait Water Neutralization process
ongoing.....");
    public JLabel s=new JLabel(new ImageIcon(getClass().getResource("/ph2.png")));

    private JLabel networkname=new JLabel("Wifi Name/Location:");
    private JLabel signal=new JLabel("Current Wifi Signal Strength:");
    private JLabel datarate=new JLabel("Network Data Rate:");
    private JLabel noiserate=new JLabel("Noise Floor Rate:");
    private JLabel latency=new JLabel("Wifi Latency Rate:");
    private JLabel bandwidth=new JLabel("Network BandWidth:");
    private JLabel Time=new JLabel("Transmission Time:");

    private JButton view_result=new JButton("Click To View Results in Chart:");

    private JTextField signalratio=new JTextField(10);
    // private JTextField networkname=new JTextField(10);
    private JTextField networktype=new JTextField(10);
    private JTextField networknametxt=new JTextField(10);
    public JTextField signaltxt=new JTextField(10);
    private JTextField dataratetxt=new JTextField(10);
    private JTextField latencytxt=new JTextField(10);
    private JTextField timetxt=new JTextField(10);
    private JTextField bandwidthtxt=new JTextField(10);
    private JTextField noiseratetxt=new JTextField(10);
    // private int size=contr.getAllStaff().size();
    public ApplicationController(){

```

```

setLayout(new BorderLayout(1,5));
networkname.grabFocus();
addButt.setBorderPainted(false);
addButt.setContentAreaFilled(false);
addButt.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){

        }
});

backButt.setBorderPainted(false);
backButt.setContentAreaFilled(false);
view_result.setBorderPainted(false);
view_result.setContentAreaFilled(false);

backButt.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){
        networktype.setText("");
        networkname.setText("");
    }
});

JPanel data=new JPanel(new BorderLayout());
JPanel dataa=new JPanel(new GridLayout(5,1));
JPanel data1=new JPanel(new GridLayout(1,4));
JPanel data2=new JPanel(new GridLayout(3,1));
JPanel data3=new JPanel(new BorderLayout());
JPanel data4=new JPanel(new GridLayout(1,4));
JPanel data3_2=new JPanel(new GridLayout(1,7));

data.setBackground(new Color(255,255,255));

JButton ar=new JButton(new ImageIcon(getClass().getResource("/arrow.png")));
JButton arrow1=new JButton(new ImageIcon(getClass().getResource("/arrow.png")));
JButton arrow2=new JButton(new ImageIcon(getClass().getResource("/arrow.png")));
JButton arrow3=new JButton(new ImageIcon(getClass().getResource("/arrow_down.png")));
JButton arrow_point=new JButton(new
ImageIcon(getClass().getResource("/arrow_point.png")));

```

```

        JButton arrow_point1=new JButton(new
ImageIcon(getClass().getResource("/arrow_point.png")));
        JButton arrow_point2=new JButton(new
ImageIcon(getClass().getResource("/arrow_point.png")));
        JButton arrow_point3=new JButton(new ImageIcon(getClass().getResource("/sact.jpg")));

        JLabel daa=new JLabel("During Data Collection by Researcher:");
        daa.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        daa.setForeground(Color.green);

        JLabel d1=new JLabel("Water from Borehole to Storage Tanks:");
        d1.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        d1.setForeground(Color.green);

        JLabel d2=new JLabel("Power Machine is Used to Increase Water Pressure:");
        d2.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        d2.setForeground(Color.green);

        JLabel d3=new JLabel("Micro thread filter for further purification");
        d3.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        d3.setForeground(Color.green);

        JLabel d5=new JLabel("Ultravalient Water Stablizer (95%)");
        d5.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        d5.setForeground(Color.green);
        JLabel d7=new JLabel("Three-In-One Machine (Wash, Fill and Cap of Battles)");
        d7.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        d7.setForeground(Color.green);
        JLabel d8=new JLabel("Machine for Battle Labeling and Packaging");
        d8.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        d8.setForeground(Color.green);

        JLabel d9=new JLabel("Sachet Water Machine Process");
        d9.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        d9.setForeground(Color.green);
        JLabel da=new JLabel(new ImageIcon(getClass().getResource("/datacol.jpg")));
        JLabel t1=new JLabel(new ImageIcon(getClass().getResource("/stage1.jpg")));

```

```

JLabel t2=new JLabel(new ImageIcon(getClass().getResource("/Stage2.png")));
JLabel t3=new JLabel(new ImageIcon(getClass().getResource("/Stage3.png")));
JLabel t4=new JLabel(new ImageIcon(getClass().getResource("/stage4.jpg")));
JLabel t33=new JLabel(new ImageIcon(getClass().getResource("/stage33.png")));
JLabel t5=new JLabel(new ImageIcon(getClass().getResource("/packaging.jpg")));
JLabel t6=new JLabel(new ImageIcon(getClass().getResource("/Stage6.jpg")));
JLabel t7=new JLabel(new ImageIcon(getClass().getResource("/stage5.png")));
JLabel t8=new JLabel(new ImageIcon(getClass().getResource("/Sachet.jpg")));
JLabel machine=new JLabel(new ImageIcon(getClass().getResource("/m.jpg")));

JPanel p1=new JPanel(new BorderLayout());
JPanel p2=new JPanel(new BorderLayout());
JPanel p3=new JPanel(new BorderLayout());
JPanel p4=new JPanel(new BorderLayout());
JPanel p5=new JPanel(new BorderLayout());
JPanel p6=new JPanel(new BorderLayout());
JPanel p7=new JPanel(new BorderLayout());
JPanel p8=new JPanel(new BorderLayout());
JPanel mach=new JPanel(new BorderLayout());
    bar = new JProgressBar();
    JLabel result= new JLabel("");

addButt.addActionListener(new ActionListener(){// adding to database
    public void actionPerformed(ActionEvent ae){
        t.start();
    }
});
add(data3,BorderLayout.CENTER);
add(data,BorderLayout.NORTH);
progresstext.setHorizontalAlignment(JLabel.CENTER);
progresstext.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
progresstext.setForeground(Color.green);
addButt.setForeground(Color.red);

networkname.setForeground(Color.BLUE);
signal.setForeground(Color.BLUE);
datarate.setForeground(Color.BLUE);

```

```

latency.setForeground(Color.BLUE);
bandwidth.setForeground(Color.BLUE);
Time.setForeground(Color.BLUE);
noiserate.setForeground(Color.BLUE);
networkname.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
signal.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
datarate.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
latency.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
bandwidth.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
Time.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
noiserate.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));

networkname.setHorizontalAlignment(JLabel.CENTER);
signal.setHorizontalAlignment(JLabel.CENTER);
datarate.setHorizontalAlignment(JLabel.CENTER);
latency.setHorizontalAlignment(JLabel.CENTER);
bandwidth.setHorizontalAlignment(JLabel.CENTER);
Time.setHorizontalAlignment(JLabel.CENTER);
noiserate.setHorizontalAlignment(JLabel.CENTER);

p1.setBackground(new Color(255,255,255));
p2.setBackground(new Color(255,255,255));
p3.setBackground(new Color(255,255,255));
p4.setBackground(new Color(255,255,255));
p5.setBackground(new Color(255,255,255));
p6.setBackground(new Color(255,255,255));
p7.setBackground(new Color(255,255,255));
p8.setBackground(new Color(255,255,255));
p1.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
p2.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
p3.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
p4.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
p5.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
p6.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
p7.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
p8.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
data4.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));

```

```

data1.setBackground(new Color(250,25,25));
data2.setBackground(new Color(150,25,255));
data3.setBackground(new Color(255,255,255));
//data3_1.setBackground(new Color(255,255,255));
data3_2.setBackground(new Color(255,255,255));
mach.setBackground(new Color(255,255,255));

// networknametxt.setBackground(new Color(255,255,255)
networknametxt.setEditable(false);
signaltxt.setEditable(false);
dataratetxt.setEditable(false);
latencytxt.setEditable(false);
bandwidthtxt.setEditable(false);
timetxt.setEditable(false);
noiseratetxt.setEditable(false);

networknametxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
signaltxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
dataratetxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
latencytxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
bandwidthtxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
timetxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
noiseratetxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
mach.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
t1.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));
t2.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));
t3.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));
t4.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));
machine.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));
data.add(data1, BorderLayout.NORTH);
data.add(data4, BorderLayout.SOUTH);
// data1.add(data3);
data1.add(p1);
data1.add(p2);
data1.add(p3);
data1.add(p4);

```

```
data4.add(p5);
data4.add(p6);
data4.add(p7);
data4.add(p8);
p1.add(da,BorderLayout.WEST);
p1.add(ar,BorderLayout.EAST);
p1.add(daa,BorderLayout.SOUTH);

p2.add(t1,BorderLayout.WEST);
p2.add(arrow1,BorderLayout.EAST);
p2.add(d1,BorderLayout.SOUTH);

p3.add(t2,BorderLayout.WEST);
p3.add(arrow2,BorderLayout.EAST);
p3.add(d2,BorderLayout.SOUTH);
p4.add(t3,BorderLayout.WEST);
//p4.add(t33,BorderLayout.CENTER);
p4.add(arrow3,BorderLayout.EAST);
p4.add(d3,BorderLayout.SOUTH);

p5.add(t8,BorderLayout.WEST);
p5.add(arrow_point3,BorderLayout.EAST);
p5.add(d9,BorderLayout.SOUTH);
p6.add(t5,BorderLayout.WEST);
p6.add(arrow_point2,BorderLayout.EAST);
p6.add(d8,BorderLayout.SOUTH);

p7.add(t6,BorderLayout.WEST);
p7.add(arrow_point,BorderLayout.EAST);
p7.add(d7,BorderLayout.SOUTH);

p8.add(t7,BorderLayout.WEST);
p8.add(arrow_point1,BorderLayout.EAST);
p8.add(d5,BorderLayout.SOUTH);
```

```

        data3.add(mach, BorderLayout.WEST);
        mach.add(machine, BorderLayout.NORTH);
ar.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){
        data.removeAll();
        data.add(new Data_Collection());
        data.doLayout();
        data.revalidate();
        data.repaint();
    }
});
ar.setBorderPainted(false);
ar.setContentAreaFilled(false);

arrow1.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){
        data.removeAll();
        data.add(new Water_From_Borehole_Storage());
        data.doLayout();
        data.revalidate();
        data.repaint();
    }
});
arrow1.setBorderPainted(false);
arrow1.setContentAreaFilled(false);

arrow2.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){
        data.removeAll();
        data.add(new Power_Machine());
        data.doLayout();
        data.revalidate();
        data.repaint();
    }
});
arrow2.setBorderPainted(false);
arrow2.setContentAreaFilled(false);

```

```

arrow3.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){
        data.removeAll();
        data.add(new Micro_Thread_Purification());
        data.doLayout();
        data.revalidate();
        data.repaint();
    }
});
arrow3.setBorderPainted(false);
arrow3.setContentAreaFilled(false);

arrow_point.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){
        data.removeAll();
        data.add(new Ultravalent_Water_Stablizer());
        data.doLayout();
        data.revalidate();
        data.repaint();
    }
});
arrow_point.setBorderPainted(false);
arrow_point.setContentAreaFilled(false);

arrow_point1.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){
        data.removeAll();
        data.add(new Three_in_One_Machine());
        data.doLayout();
        data.revalidate();
        data.repaint();
    }
});
arrow_point1.setBorderPainted(false);
arrow_point1.setContentAreaFilled(false);

```

```

arrow_point2.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){
        data.removeAll();
        data.add(new Battling());
        data.doLayout();
        data.revalidate();
        data.repaint();
    }
});
arrow_point2.setBorderPainted(false);
arrow_point2.setContentAreaFilled(false);

arrow_point3.setBorderPainted(false);
arrow_point3.setContentAreaFilled(false);

    bar.getModel();
bar.setForeground(new Color(250,250,150));
bar.setBackground(Color.white);
bar.setBorder(BorderFactory.createLineBorder(Color.red));
// bar.setBorder(BorderFactory.createCompoundBorder());
bar.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
bar.setBorder(BorderFactory.createEtchedBorder(Color.white, Color.yellow));
bar.setStringPainted(true);
bar.setMinimum(0);
bar.setMaximum(14);
bar.setFont(new Font(Font.SANS_SERIF,Font.BOLD,40));
    setVisible(true);
    t = new Thread(){
public void run(){
    try{
        for (i= 0; i<=1.4; i++){
            bar.setValue(i);
            Thread.sleep(5000);
            repaint();
            if (i == 0){
                if(i==0){
                    bar.setForeground(new Color(4,250,10));

```

```

    }
    String lb="Battery Acid, Hydrochloric acid with (pH = 0";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);

}else if(i ==2){
    if(i==2){
        bar.setForeground(new Color(4,200,250));
    }
    String lb="Lemon Juice Vineger with (pH = 1-2)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);

}else if(i ==3){
    if(i==3){
        bar.setForeground(new Color(4,10,250));
    }
    String lb="Grape Fruit Juice, Soda, Tomato Juice with (ph = 2.5, 3-5)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);

}else if(i ==5){
    if(i==5){
        bar.setForeground(new Color(200,200,250));
    }
    String lb="Black Coffee with (pH = 5)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);

}else if(i ==6){

```

```

if(i==6){
    bar.setForeground(new Color(02,120,150));
}

String lb="Milk, Urine, Saliva with (pH = 6.3)";
searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
searchtxt.setForeground(Color.red);
} else if(i ==7){
    if(i==7){
        bar.setForeground(new Color(02,120,150));
    }

    String lb="Milk, Urine, Saliva with (pH = 6.6)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.green);
}

else if(i ==8){
    if(i==8){
        bar.setForeground(new Color(100,120,250));
    }

    String yes= JOptionPane.showInputDialog(null, " Enter 'Yes or No' to Stop or
Continue Neutralization Process? ");

    if(yes.equalsIgnoreCase("yes")){
        //JOptionPane.showConfirmDialog(null, "Neutral Water with (pH = 7.0)");
        JOptionPane.showMessageDialog(null, "Neutral Water with (pH = 6.8-7.2)");
        String lb="Neutral Water with (pH = 6.8)";
        searchtxt.setText("NAFDAC APPROVED:"+ i +"\t"+ lb);
        searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        searchtxt.setForeground(Color.red);
        currentThread().stop();
    }
}

```

```

    } else if(yes.equalsIgnoreCase("no")){
        JOptionPane.showMessageDialog(null, "Wait While Neutralization Process is
ongoing.");
    }

}

}else if(i ==9){
    if(i==9){
        bar.setForeground(new Color(10,200,250));
    }

    String lb="Blood with (pH = 7.4)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);
}

}else if(i ==10){
    if(i==10){
        bar.setForeground(new Color(100,20,50));
    }

    String lb="Sea Water with (pH = 8)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);
}

}else if(i ==11){
    if(i==11){
        bar.setForeground(new Color(150,00,150));
    }

    String lb="Baking Soda with (pH = 9.5)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);
}

}else if(i ==12){
    if(i==12){
        bar.setForeground(new Color(250,00,250));
    }

    String lb="Ammonia Solution with (pH = 10.5-11.5)";

```

```

        searchtxt.setText("LEVEL:" + i + "\t" + lb);
        searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        searchtxt.setForeground(Color.red);

    }else if(i ==13){
        if(i==13){
            bar.setForeground(new Color(237,50,125));
        }

        String lb="Bleaches, Owen Cleaner, Iye with (pH = 13.5)";
        searchtxt.setText("LEVEL:" + i + "\t" + lb);
        searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        searchtxt.setForeground(Color.red);
    }else if(i ==14){
        if(i==14){
            bar.setForeground(new Color(207,207,15));
        }

        String lb="Liquid Drain Cleaner with (pH = 14)";
        searchtxt.setText("LEVEL:" + i + "\t" + lb);
        searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        searchtxt.setForeground(Color.red);

    }

    }
} catch(InterruptedException ea){}

};

};}

public void run() {
    throw new UnsupportedOperationException("Not supported yet.");
}
}

public class Analyze extends JPanel implements Runnable{
    public JButton addButt=new JButton("CLICK TO START NEUTRALIZATION PROCESS",new
    ImageIcon(getClass().getResource("/d.jpg")));

```

```

    public JButton backButt=new JButton("Clear",new
ImageIcon(getClass().getResource("/bk.png")));
    private JLabel progresstext=new JLabel("Wait Water Neutralization process
ongoing.....");
    public JLabel s=new JLabel(new ImageIcon(getClass().getResource("/ph2.png")));

    private JLabel networkname=new JLabel("Wifi Name/Location:");
    private JLabel signal=new JLabel("Current Wifi Signal Strength:");
    private JLabel datarate=new JLabel("Network Data Rate:");
    private JLabel noiserate=new JLabel("Noise Floor Rate:");
    private JLabel latency=new JLabel("Wifi Latency Rate:");
    private JLabel bandwidth=new JLabel("Network BandWidth:");
    private JLabel Time=new JLabel("Transmission Time:");

    private JButton view_result=new JButton("Click To View Results in Chart:");
    private JTextField signalratio=new JTextField(10);
    // private JTextField networkname=new JTextField(10);
    private JTextField networktype=new JTextField(10);
    private JTextField networknametxt=new JTextField(10);
    public JTextField signaltxt=new JTextField(10);
    private JTextField dataratetxt=new JTextField(10);
    private JTextField latencytxt=new JTextField(10);
    private JTextField timetxt=new JTextField(10);
    private JTextField bandwidthtxt=new JTextField(10);
    private JTextField noiseratetxt=new JTextField(10);

    // private int size=contr.getAllStaff().size();
    public Analyze(){
        setLayout(new BorderLayout(1,5));

        networkname.grabFocus();
        addButt.setBorderPainted(false);
        addButt.setContentAreaFilled(false);

        addButt.addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent e){

```

```

    }
});

backButt.setBorderPainted(false);
backButt.setContentAreaFilled(false);
view_result.setBorderPainted(false);
view_result.setContentAreaFilled(false);

backButt.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){
        networktype.setText("");
        networkname.setText("");
    }
});

JPanel data=new JPanel(new GridLayout(5,1));
JPanel data1=new JPanel(new BorderLayout(1,1));
JPanel data2=new JPanel(new GridLayout(3,1));
JPanel data3=new JPanel(new GridLayout(2,1));
JPanel data3_1=new JPanel(new GridLayout(1,7));
JPanel data3_2=new JPanel(new GridLayout(1,7));

data.setBackground(new Color(255,255,255));

    bar = new JProgressBar();
    JLabel result= new JLabel("");

addButt.addActionListener(new ActionListener(){// adding to database
    public void actionPerformed(ActionEvent ae){
        t.start();
    }
});

add(data,BorderLayout.CENTER);

progresstext.setHorizontalAlignment(JLabel.CENTER);
progresstext.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));

```

```

progresstext.setForeground(Color.green);
addButt.setForeground(Color.red);
networkname.setForeground(Color.BLUE);
signal.setForeground(Color.BLUE);
datarate.setForeground(Color.BLUE);
latency.setForeground(Color.BLUE);
bandwidth.setForeground(Color.BLUE);
Time.setForeground(Color.BLUE);
noiserate.setForeground(Color.BLUE);

networkname.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
signal.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
datarate.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
latency.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
bandwidth.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
Time.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));
noiserate.setFont(new Font(Font.SANS_SERIF,Font.BOLD,10));

networkname.setHorizontalAlignment(JLabel.CENTER);
signal.setHorizontalAlignment(JLabel.CENTER);
datarate.setHorizontalAlignment(JLabel.CENTER);
latency.setHorizontalAlignment(JLabel.CENTER);
bandwidth.setHorizontalAlignment(JLabel.CENTER);
Time.setHorizontalAlignment(JLabel.CENTER);
noiserate.setHorizontalAlignment(JLabel.CENTER);

data2.setBackground(new Color(255,255,255));
data3_1.setBackground(new Color(255,255,255));
data3_2.setBackground(new Color(255,255,255));
data1.setBackground(new Color(255,255,255));

// networknametxt.setBackground(new Color(255,255,255));

networknametxt.setEditable(false);
signaltxt.setEditable(false);
dataratetxt.setEditable(false);
latencytxt.setEditable(false);

```

```

bandwidthtxt.setEditable(false);
timetxt.setEditable(false);
noiseratetxt.setEditable(false);

networknametxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
signaltxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
dataratetxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
latencytxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
bandwidthtxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
timetxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
noiseratetxt.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));

data.add(data2);

data2.add(addButt);
//data2.add(progresstext);
data2.add(s);
data2.add(bar);
    bar.getModel();
bar.setForeground(new Color(250,250,150));
bar.setBackground(Color.white);
bar.setBorder(BorderFactory.createLineBorder(Color.red));
// bar.setBorder(BorderFactory.createCompoundBorder());
bar.setBorder(BorderFactory.createBevelBorder(BevelBorder.RAISED));
bar.setBorder(BorderFactory.createEtchedBorder(Color.white, Color.yellow));
bar.setStringPainted(true);
bar.setMinimum(0);
bar.setMaximum(14);
bar.setFont(new Font(Font.SANS_SERIF,Font.BOLD,40));
    setVisible(true)
    t = new Thread(){
public void run(){
    try{
        for (i= 0; i<=14; i++){
            bar.setValue(i);
            Thread.sleep(5000);
            repaint()

```

```

if (i == 0){
    if(i==0){
        bar.setForeground(new Color(4,250,10));
    }
    String lb="Battery Acid, Hydrochloric acid with (pH = 0";
    searchtxt.setText("LEVEL:" + i + "\t" + lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);
    searchtxt.setText("LEVEL:" + i + "\t" + lb);

}else if(i ==2){
    if(i==2){
        bar.setForeground(new Color(4,200,250));
    }

    String lb="Lemon Juice Vineger with (pH = 1-2)";
    searchtxt.setText("LEVEL:" + i + "\t" + lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);

}else if(i ==3){
    if(i==3){
        bar.setForeground(new Color(4,10,250));
    }
    String lb="Grape Fruit Juice, Soda, Tomato Juice with (ph = 2.5, 3-5)";
    searchtxt.setText("LEVEL:" + i + "\t" + lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);

}else if(i ==5){
    if(i==5){
        bar.setForeground(new Color(200,200,250));
    }

    String lb="Black Coffee with (pH = 5)";
    searchtxt.setText("LEVEL:" + i + "\t" + lb);

```

```

        searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        searchtxt.setForeground(Color.red);
    }else if(i ==6){
        if(i==6){
            bar.setForeground(new Color(02,120,150));
        }

        String lb="Milk, Urine, Saliva with (pH = 6.3)";
        searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
        searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        searchtxt.setForeground(Color.red);
    } else if(i ==7){
        if(i==7){
            bar.setForeground(new Color(02,120,150));
        }

        String lb="Milk, Urine, Saliva with (pH = 6.6)";
        searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
        searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
        searchtxt.setForeground(Color.green);
    }

    else if(i ==8){
        if(i==8){
            bar.setForeground(new Color(100,120,250));
        }

        String yes= JOptionPane.showInputDialog(null, " Enter 'Yes or No' to Stop or
Continue Neutralization Process? ");

```

```

        if(yes.equalsIgnoreCase("yes")){
            //JOptionPane.showConfirmDialog(null, "Neutral Water with (pH = 7.0)");
            JOptionPane.showMessageDialog(null, "Neutral Water with (pH = 6.8-7.2)");
            String lb="Neutral Water with (pH = 6.8)";
            searchtxt.setText("NAFDAC APPROVED:"+ i +"\t"+ lb);
            searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
            searchtxt.setForeground(Color.red);
            currentThread().stop();
        }
    }
}

```

```

    } else if(yes.equalsIgnoreCase("no")){
        JOptionPane.showMessageDialog(null, "Wait While Neutralization Process is
ongoing.");
    }

}

}else if(i ==9){
    if(i==9){
        bar.setForeground(new Color(10,200,250));
    }

    String lb="Blood with (pH = 7.4)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);

}

}else if(i ==10){
    if(i==10){
        bar.setForeground(new Color(100,20,50));
    }

    String lb="Sea Water with (pH = 8)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);

}

}else if(i ==11){
    if(i==11){
        bar.setForeground(new Color(150,00,150));
    }

    String lb="Baking Soda with (pH = 9.5)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);

}

}else if(i ==12){
    if(i==12){
        bar.setForeground(new Color(250,00,250));
    }
}

```

```

    }

    String lb="Ammonia Solution with (pH = 10.5-11.5)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red)
}else if(i ==13){
    if(i==13){
        bar.setForeground(new Color(237,50,125));
    }

    String lb="Bleaches, Owen Cleaner, Iye with (pH = 13.5)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);

}else if(i ==14){
    if(i==14){
        bar.setForeground(new Color(207,207,15));
    }

    String lb="Liquid Drain Cleaner with (pH = 14)";
    searchtxt.setText("LEVEL:"+ i +"\t"+ lb);
    searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
    searchtxt.setForeground(Color.red);
    }

}
} catch(InterruptedException ea){}

};

};}

public void run() {
    throw new UnsupportedOperationException("Not supported yet.");
}
}

```

```

private class FrontPage extends JPanel{
    private JButton viewModuleButt=new JButton("View Admin Profile:",new
ImageIcon(getClass().getResource("/search.png")));
    private JButton searchbutt=new JButton("Check NAFDAC Approved Point.",new
ImageIcon(getClass().getResource("/view.png")));

    public FrontPage(){
        setLayout(new BorderLayout(1,5));
        setBackground(new Color(250,250,150));
        // modulesCmb.setFont(new Font(Font.MONOSPACED,Font.BOLD,15));
        viewModuleButt.setBorderPainted(false);
        viewModuleButt.setContentAreaFilled(false);
        searchbutt.setBorderPainted(false);
        searchbutt.setContentAreaFilled(false);
        viewModuleButt.addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent e){
                Object o=modulesCmb.getSelectedItem();
            }
        });
        searchbutt.addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent e){
                String dd="NAFDAC APPROVED WATER TREATMENT pH VALUE is 6.5- 7.2";
                searchtxt.setText(""+ dd);
                searchtxt.setFont(new Font(Font.SANS_SERIF,Font.BOLD,20));
                searchtxt.setForeground(Color.green);
            }
        });
        JPanel data=new JPanel(new GridBagLayout());
        data.setBackground(new Color(255,255,255));
        searchpanel=new JPanel(new BorderLayout());
        searchtxt.setBackground(new Color(250,250,150));
        searchtxt.setEditable(false);
        data.add(new JLabel("<html><b><u><span style='font-size:14px;'>NAFDAC
APPROVED POINT</span></u></b></html>"),new
GridBagConstraints(0,0,2,1,1.0,1.0,GridBagConstraints.CENTER,GridBagConstraints.NONE,new
Insets(4,4,20,4),1,1));

```

```

        data.add(new JLabel("NEUTRALIZATION STAGE: "),new
GridBagConstraints(0,1,1,1,1.0,1.0,GridBagConstraints.EAST,GridBagConstraints.NONE,new
Insets(1,1,1,1),1,1));
        data.add(searchtxt,new
GridBagConstraints(1,1,1,1,1.0,1.0,GridBagConstraints.WEST,GridBagConstraints.NONE,new
Insets(1,1,1,1),1,1));
        // data.add(bar1,new
GridBagConstraints(0,2,2,1,1.0,1.0,GridBagConstraints.WEST,GridBagConstraints.NONE,new
Insets(1,1,1,1),1,1));
        data.add(searchbutt,new
GridBagConstraints(1,2,2,1,1.0,1.0,GridBagConstraints.EAST,GridBagConstraints.NONE,new
Insets(1,1,1,1),1,1));
        // spanel1. add(bar1);
        JTabbedPane tp=new JTabbedPane();

        tp.addTab("Stages of the Water Treatment Plant:", new ApplicationController());
        tp.addTab("Start Neutralization Process:", new Analyze());
        tp.addTab("More Information:",new ApplicationInformation());

        // tp.addTab("RESULTS CHART:",new wifiopt());
        add(tp,BorderLayout.CENTER);
        info2Lbl=new JLabel("***APPLICATION CONTROL CENTRE***");
        info2Lbl.setHorizontalAlignment(JLabel.CENTER);
        info2Lbl.setFont(new Font(Font.SANS_SERIF,Font.BOLD,15));
        add(info2Lbl,BorderLayout.NORTH
JPanel main=new JPanel(new BorderLayout(1,5));
        main.add(data,BorderLayout.NORTH);
        main.add(tp,BorderLayout.CENTER);

        JScrollPane scroll=new JScrollPane(main);
        scroll.getViewport().getView().setBackground(new Color(250,250,150));

scroll.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL_SCROLLBAR_AS_NEEDED);
        scroll.setPreferredSize(new Dimension(1000,(int)5));//this panel is for control center Right
side panel

```

```
        add(scroll, BorderLayout.CENTER);
    }

}

public static void main(String[] don){
    // new Application();
}

public void run() {
    throw new UnsupportedOperationException("Not supported yet.");
}

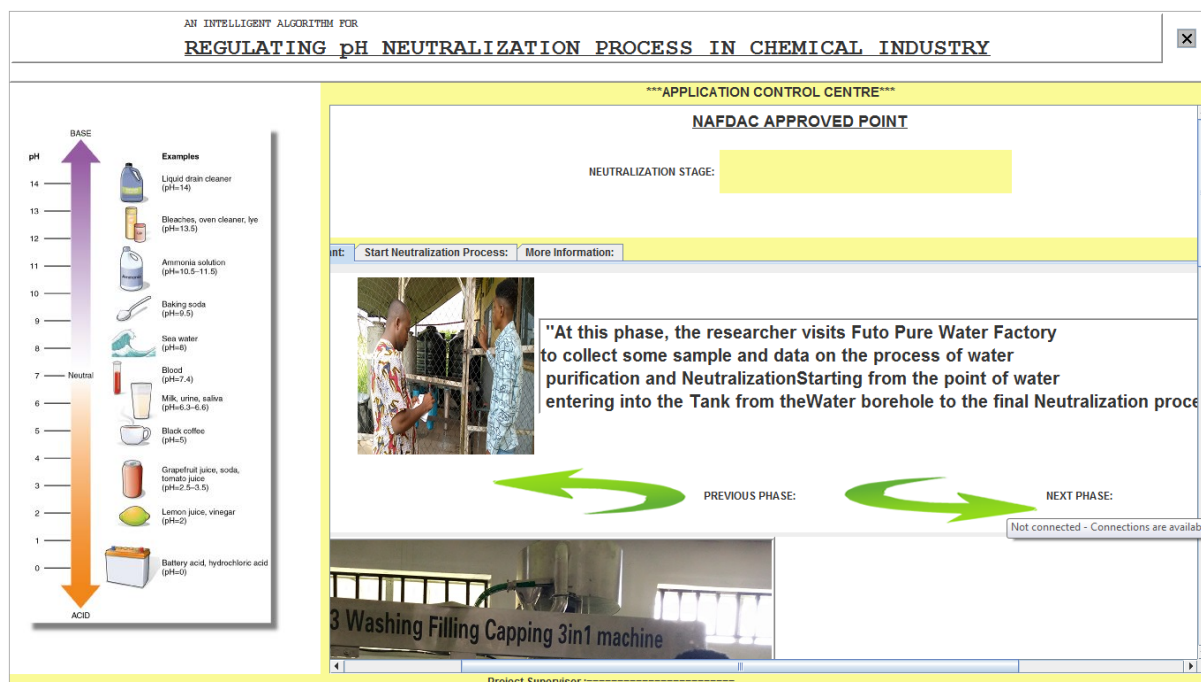
}
```

APPENDIX B

PROGRAM SAMPLE OUTPUT



Appendix B1: Login Interface for the Application



Appendix B2: Stages of Neutralization Process

AN INTELLIGENT ALGORITHM FOR
REGULATING pH NEUTRALIZATION PROCESS IN CHEMICAL INDUSTRY

APPLICATION CONTROL CENTRE

NAFDAC APPROVED POINT

NEUTRALIZATION STAGE: [Yellow Box]

Start Neutralization Process: More Information:

Project Supervisor :=====

Appendix B3: Production Process for Water Treatment Plant

AN INTELLIGENT ALGORITHM FOR
REGULATING pH NEUTRALIZATION PROCESS IN CHEMICAL INDUSTRY

APPLICATION CONTROL CENTRE

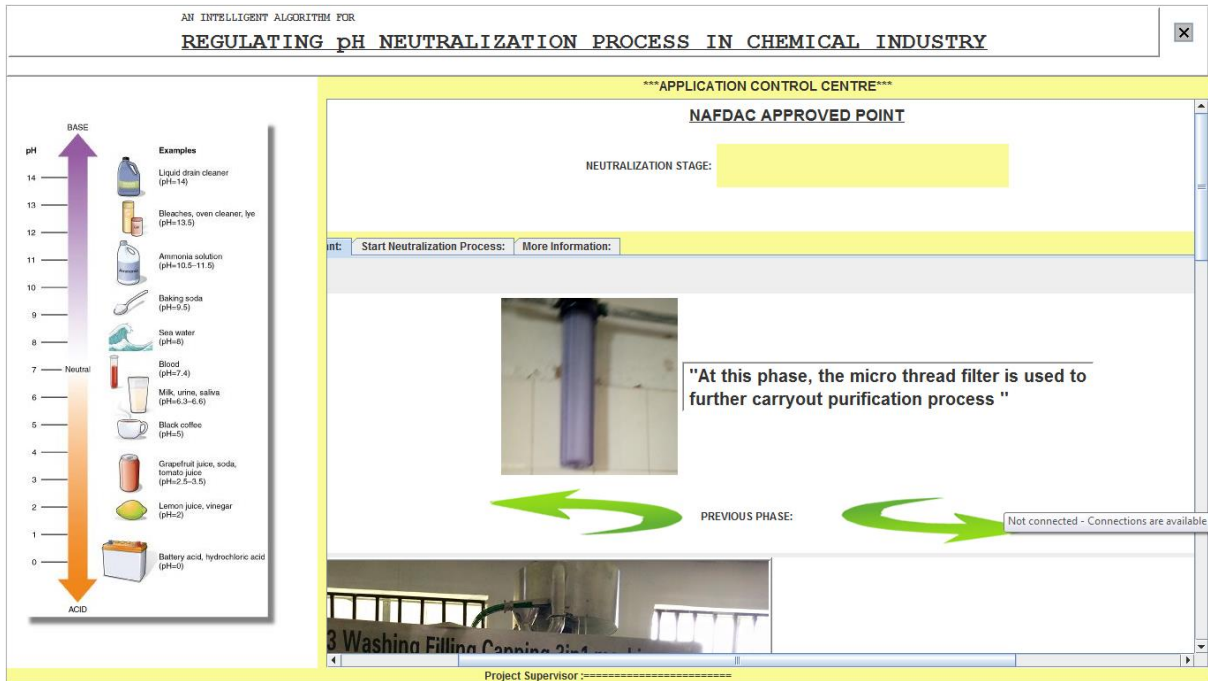
NAFDAC APPROVED POINT

NEUTRALIZATION STAGE: [Yellow Box]

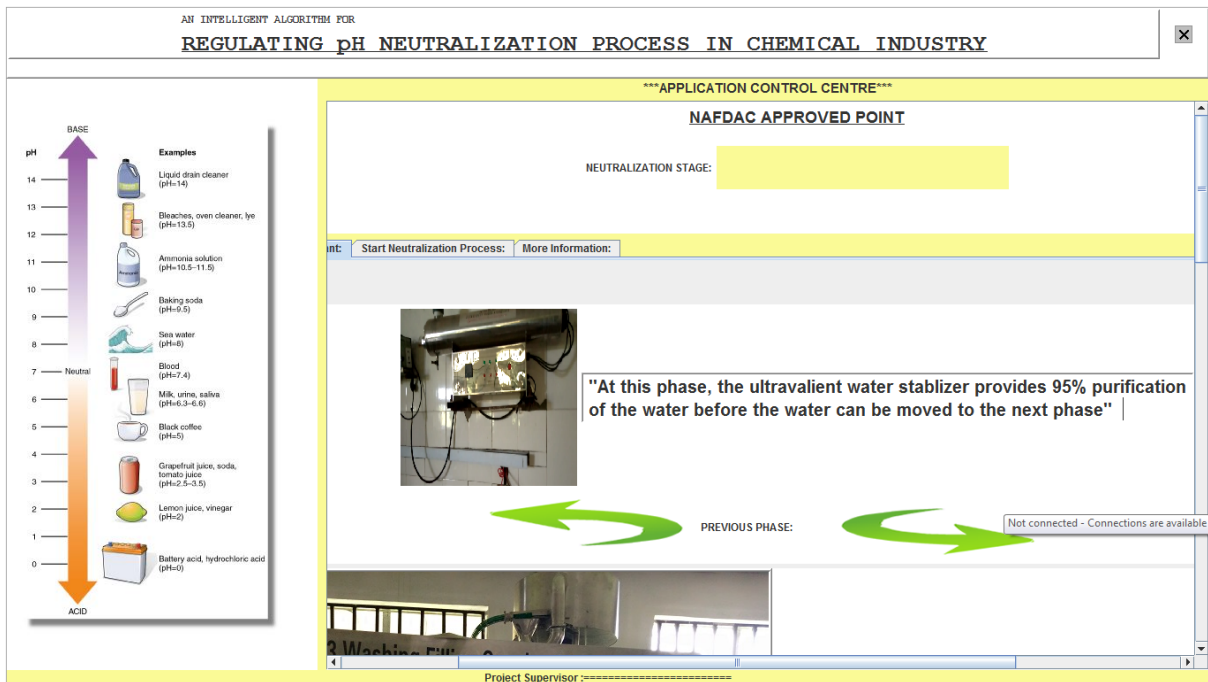
Start Neutralization Process: More Information:

Project Supervisor :=====

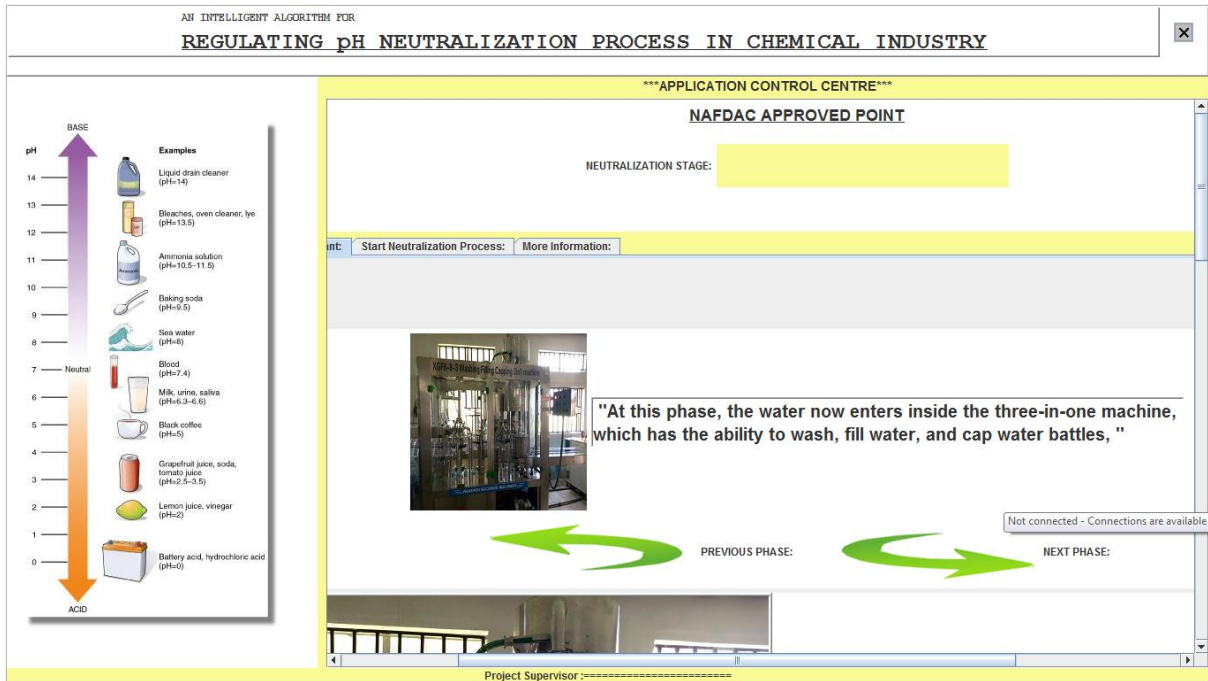
Appendix B4: Production Process Stages



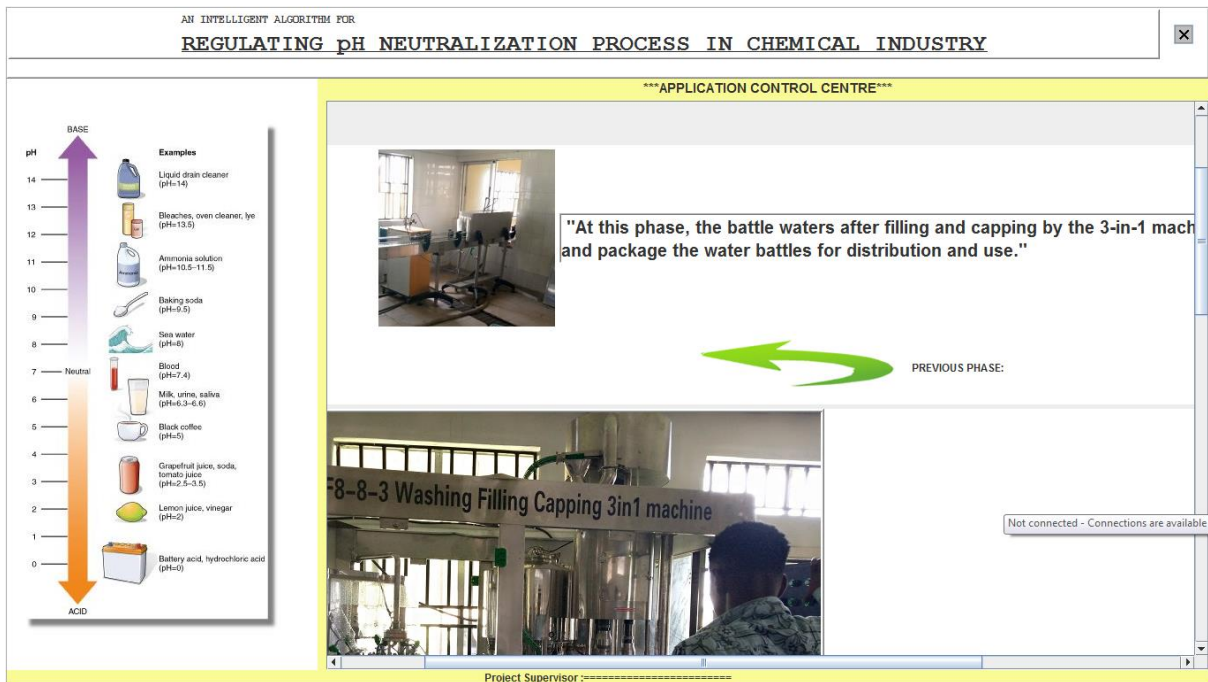
Appendix B5: Production Process Sterilization for Water Treatment Plant



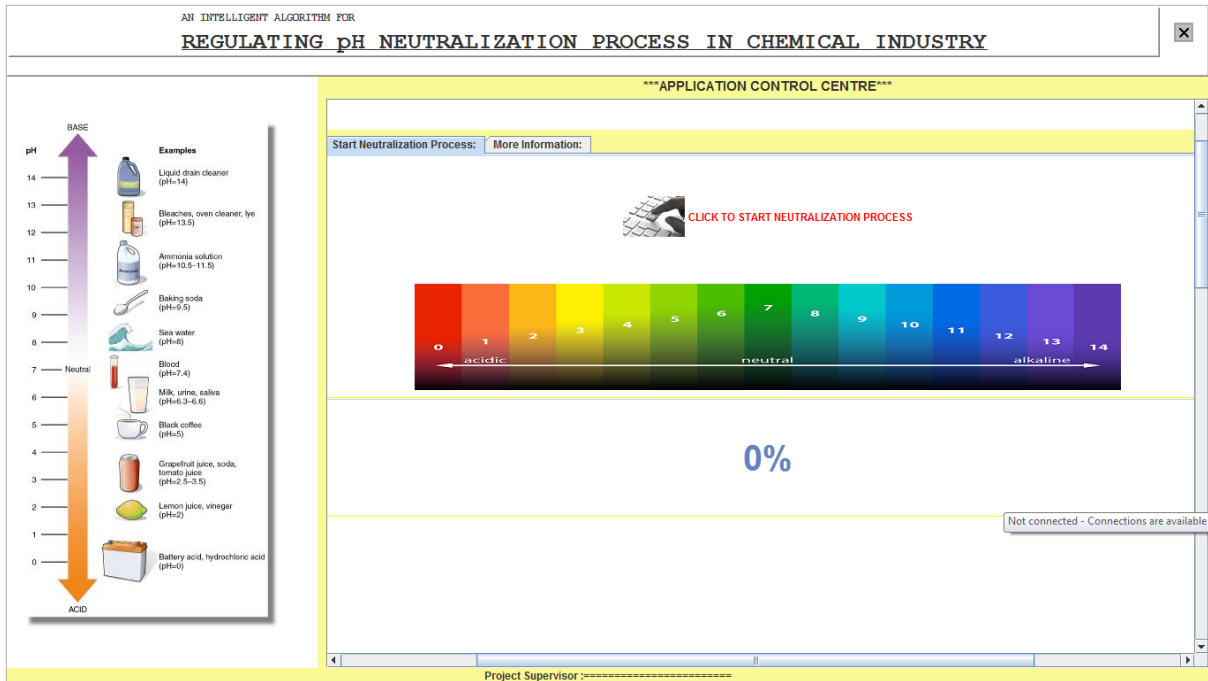
Appendix B6: Production Process Machine for Water Treatment Plant



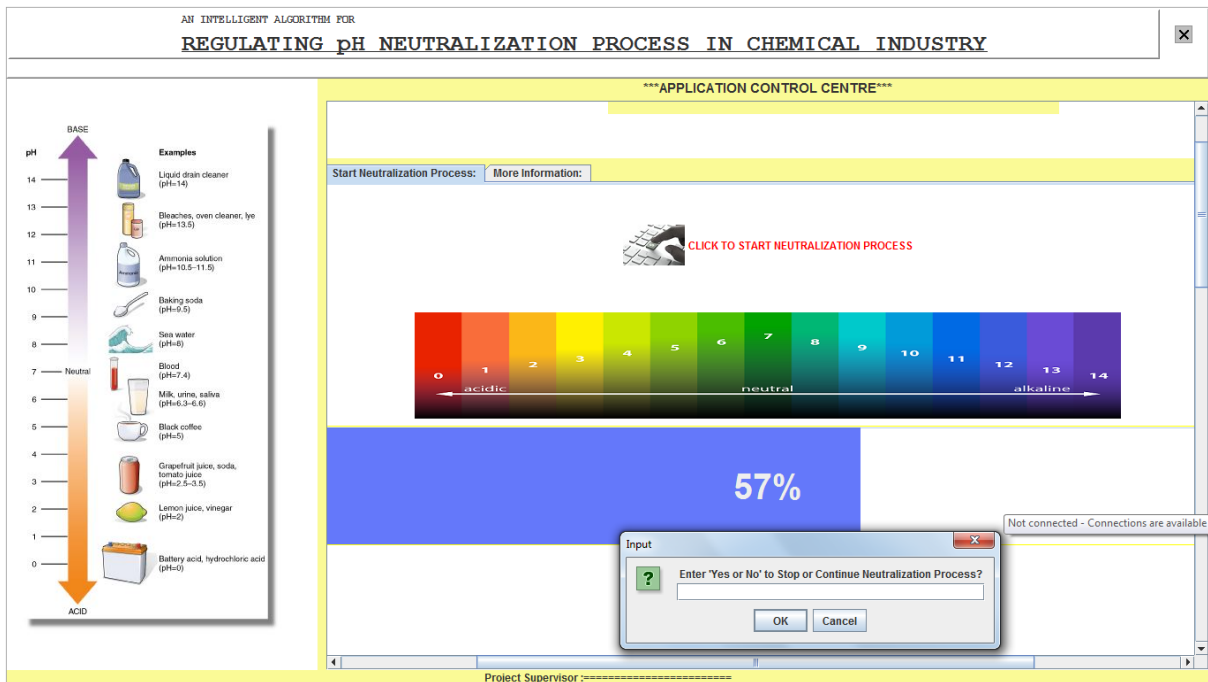
Appendix B7: Production Process Coping Machine for Water Production



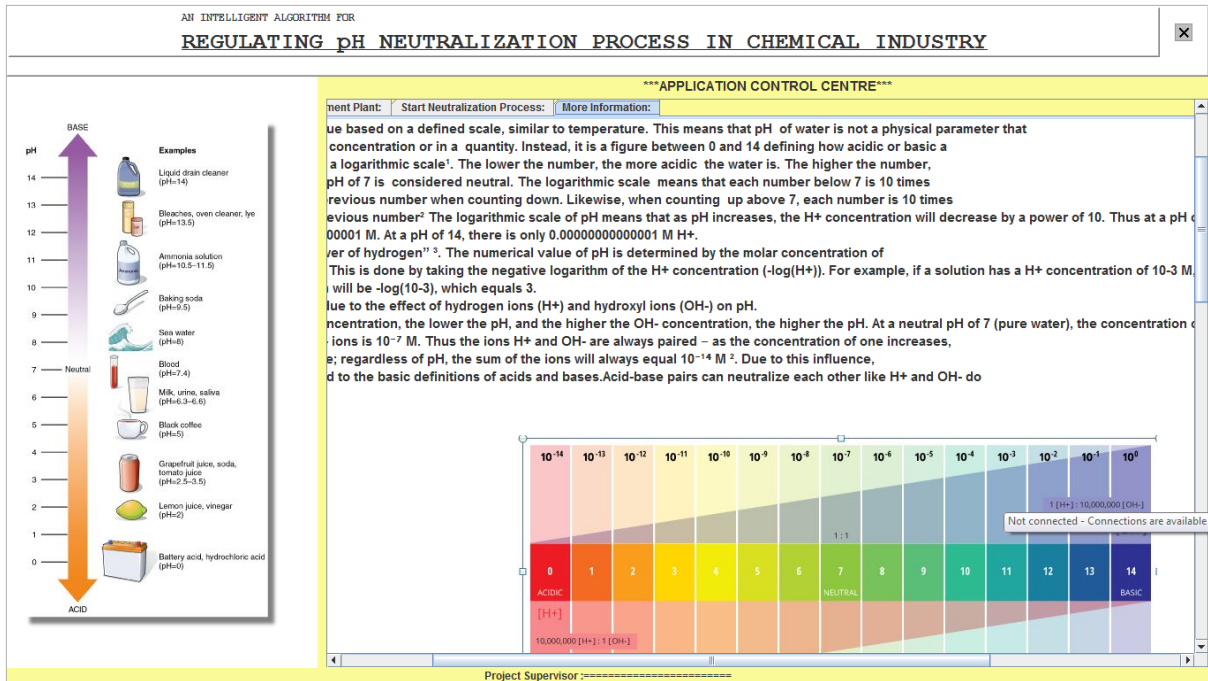
Appendix B8: Production Process Machine 3 in 1 for water Production



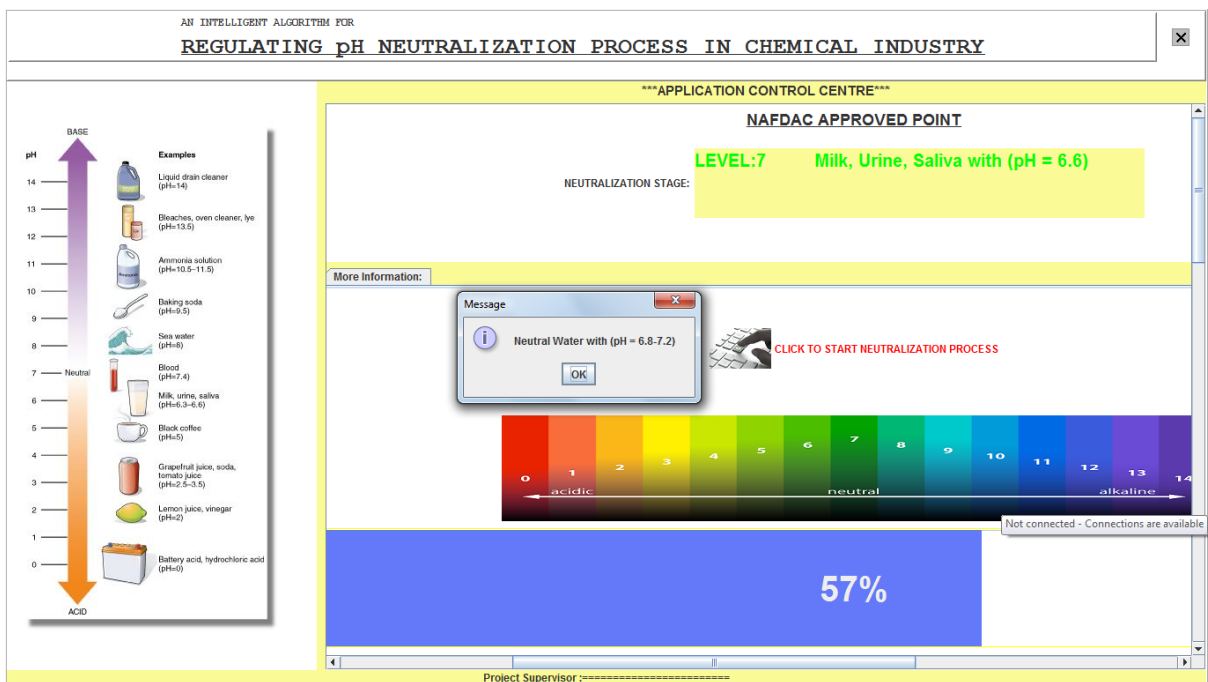
Appendix B9: Neutralization Processing Testing



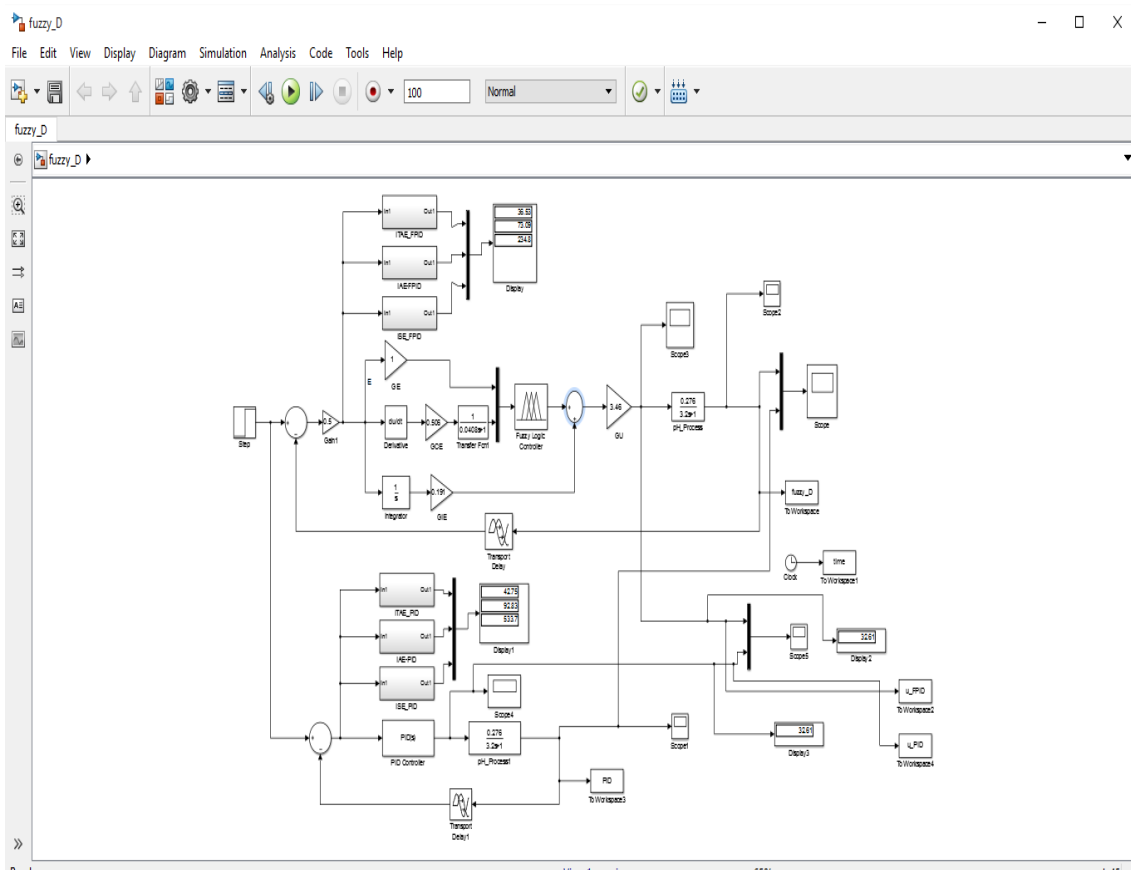
Appendix B10: Neutralization Processing acceptance level by NAFDAC



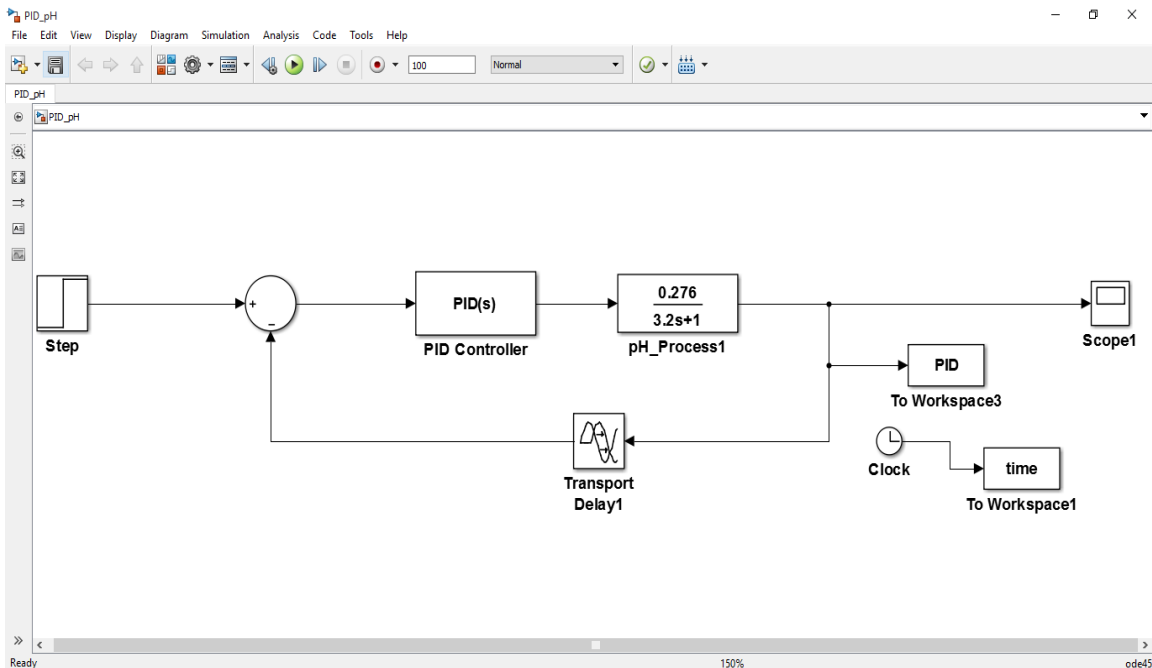
Appendix B11: Summary of Neutralization Process



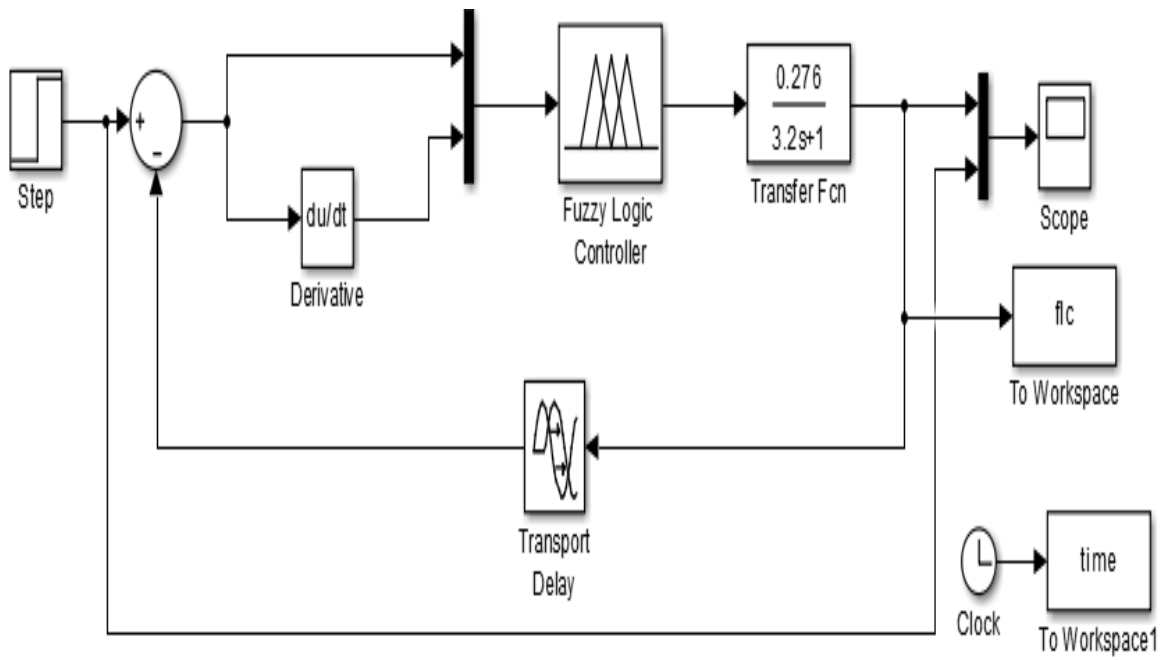
Appendix B12: Process regulation of pH Process



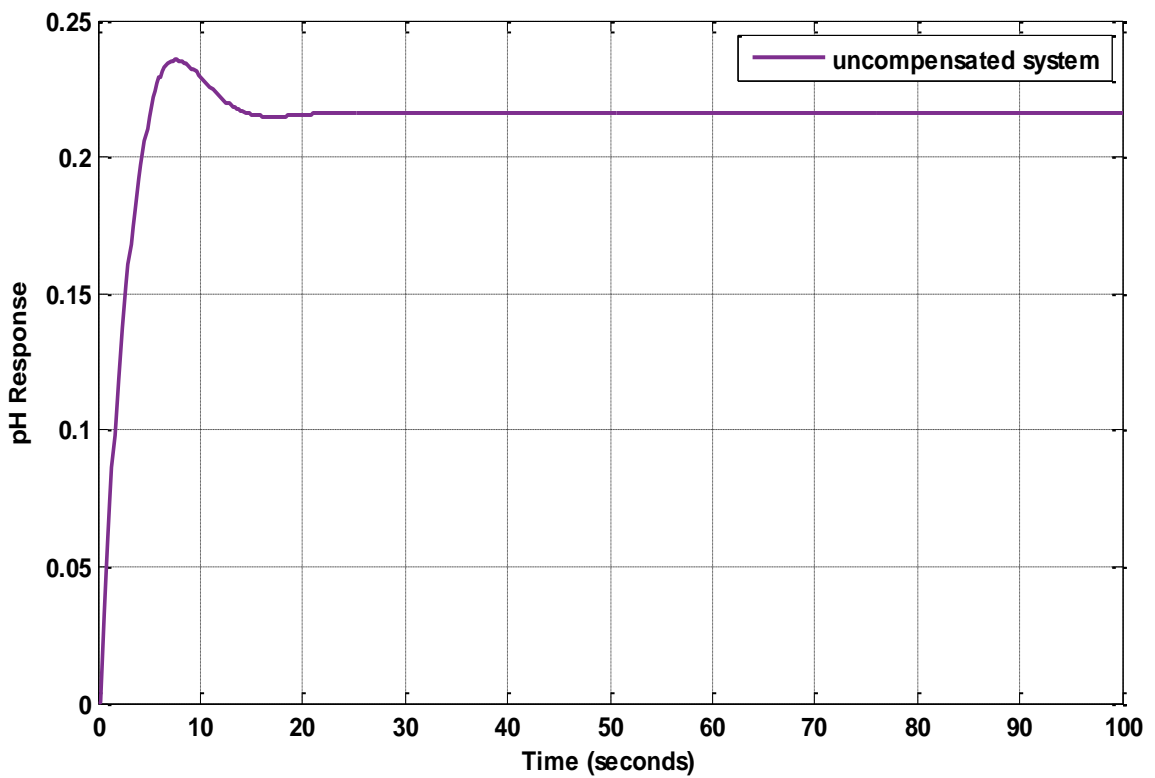
Appendix B13: Snap shot of MATLAB/Simulink GUI environment for simulation



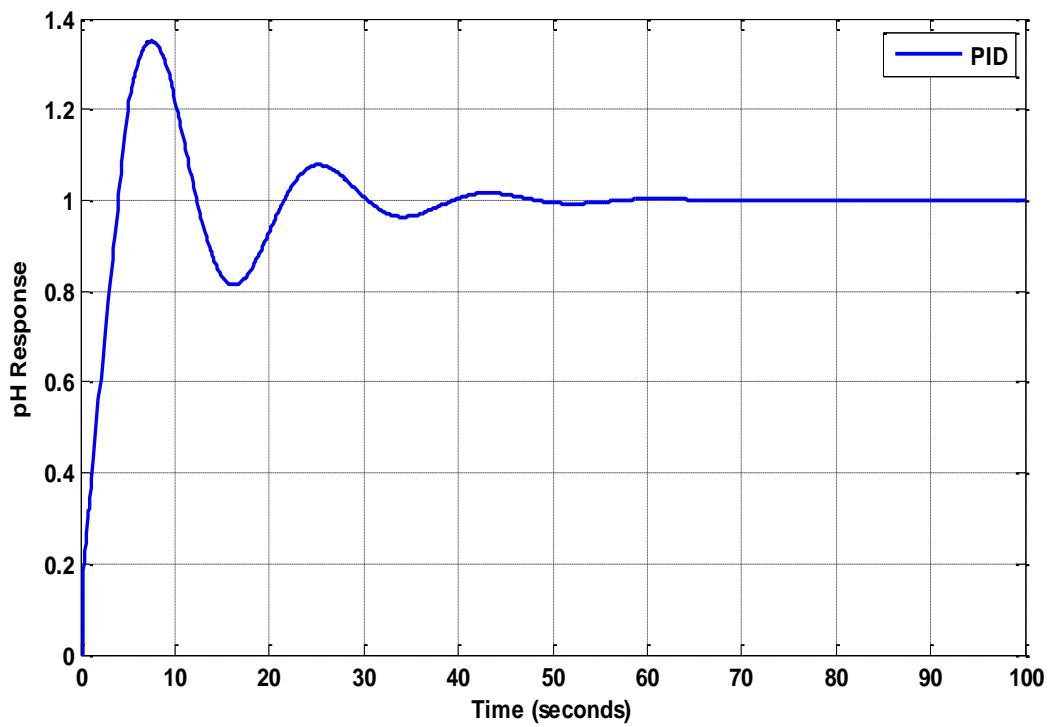
B14: Simulink model of pH process with PID compensator



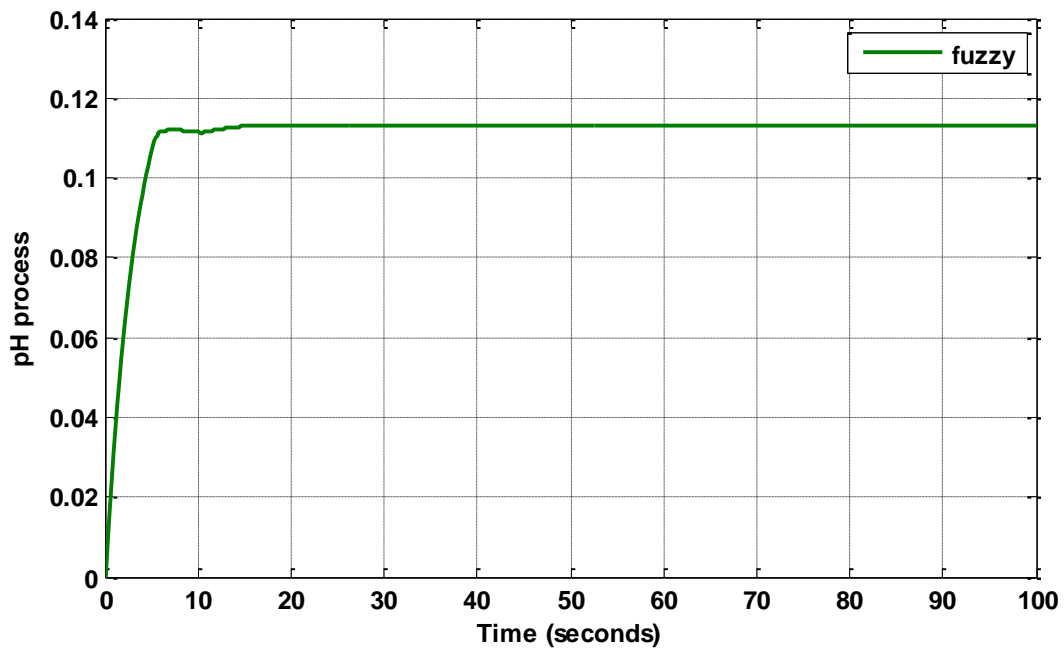
B15: Fuzzy logic compensated pH process



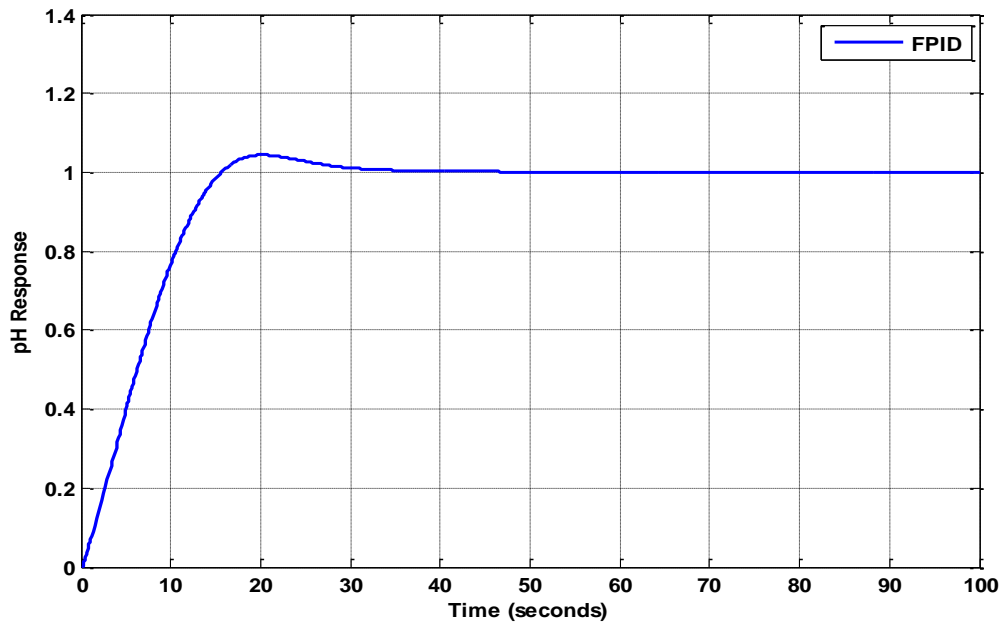
B16: Unit step Response of uncompensated pH process



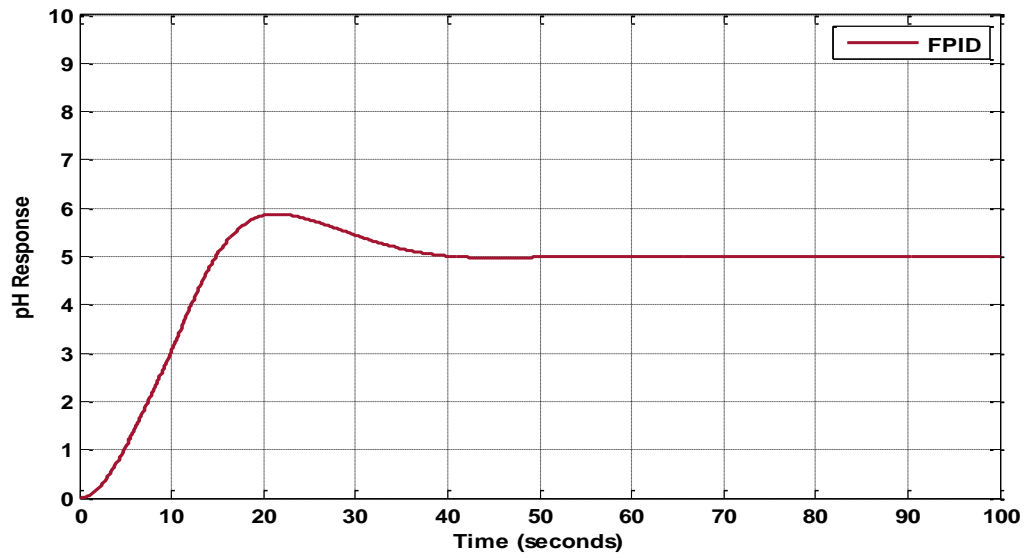
B16: Step response of PID compensated pH process



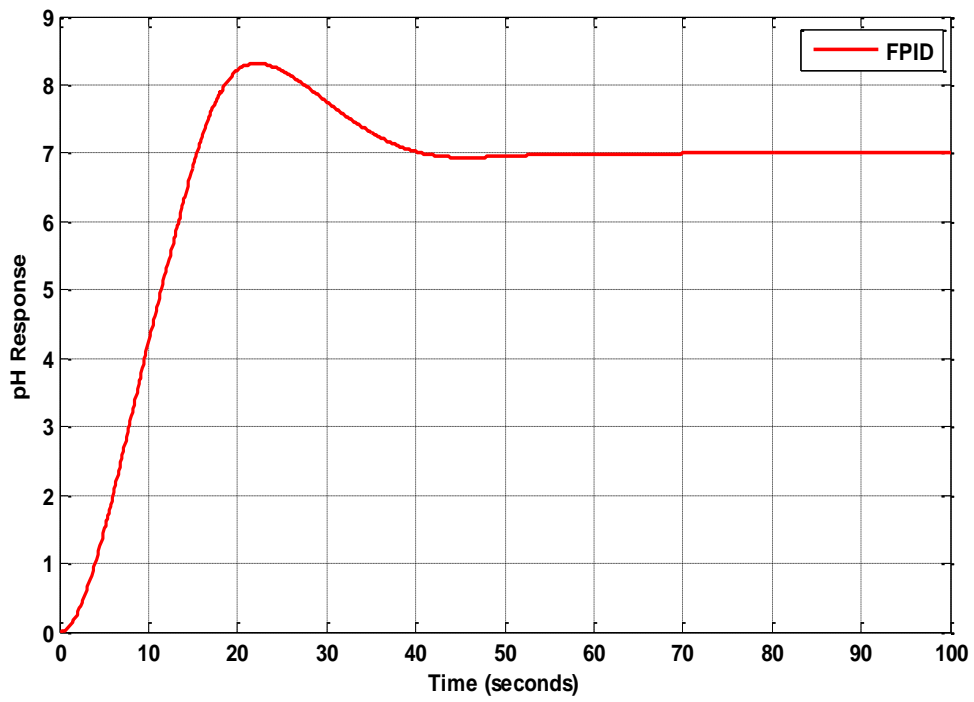
B17: Step response of Fuzzy Logic compensated pH process



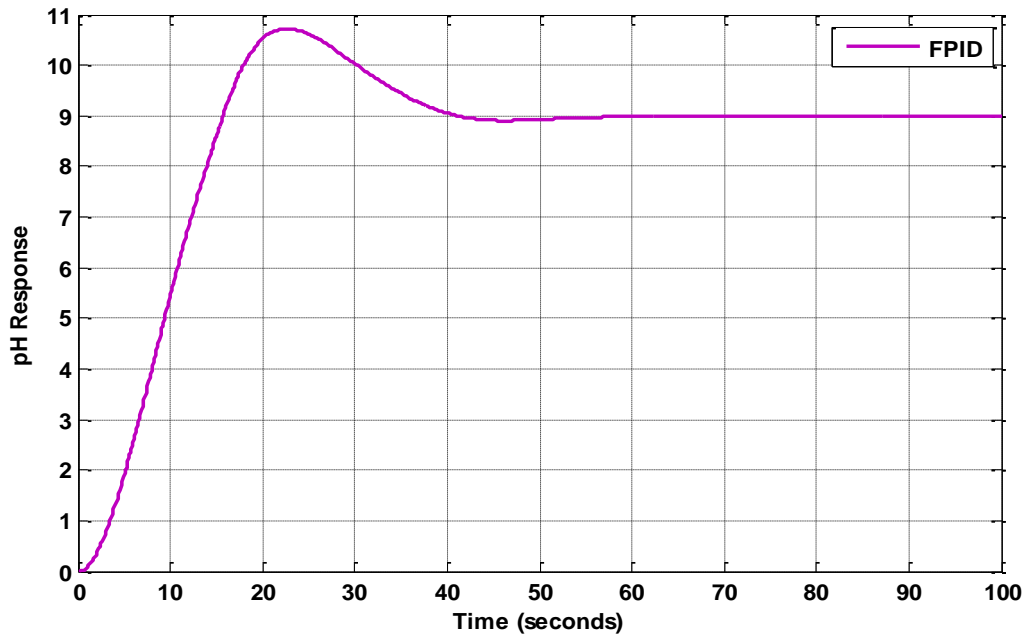
B18: Step response of Fuzzy-PID compensated pH process



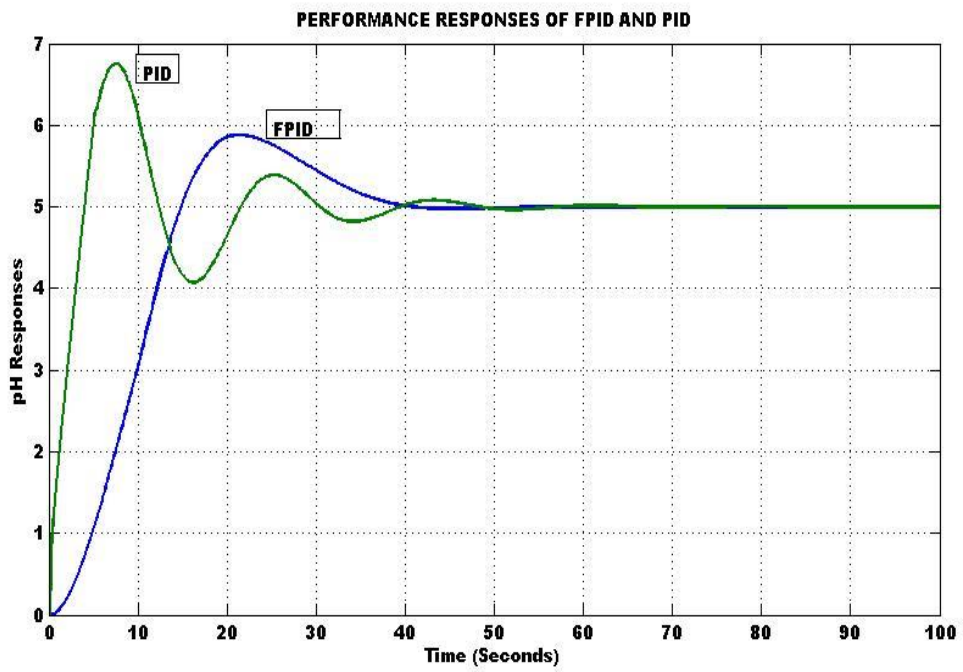
B19: Step response at pH setpoint 5 (Acid medium)



B20: Step response at pH setpoint 7 (Neutral medium)



B21: Step response at pH setpoint 9 (Base medium)



B22: Performance Responses for FPID and PID for pH value of 5(Acid Medium)