

**MODELLING AND CONTROL OF SECOND ORDER PLUS  
DEAD TIME (SOPDT) SYSTEMS**

**Thesis submitted  
by  
MAHUA PAL**

**DOCTOR OF PHILOSOPHY (ENGINEERING)**

**Department of Instrumentation & Electronics Engineering  
Faculty Council of Engineering & Technology  
Jadavpur University, (Salt lake Campus)  
Kolkata -700106  
India**

**2024**

**Jadavpur University**  
**Kolkata-700 032, INDIA**

**Index No. 90/17/E**  
**Registration No. D-7/E/05/17**

**1. Title of thesis:** **Modeling and Control of Second Order Plus Dead Time (SOPDT) Systems**

**2. Name, Designation & Institution of the Supervisors:** **Prof. Kumardeb Banerjee**  
Dept. of Instrumentation & Electronics Engineering  
Jadavpur University, Salt Lake Campus,  
Kolkata-700106, India.

**Prof. Bivas Dam**  
Dept. of Instrumentation & Electronics Engineering  
Jadavpur University, Salt Lake Campus,  
Kolkata-700106, India.

**3. List of Publications**

- [1] Mahua Pal<sup>1</sup>, D. Biswas<sup>2</sup>, A. Das<sup>3</sup>, K. Banerjee<sup>4</sup>, B.Dam<sup>4</sup>. (2021), "Rapid Modeling of a Fast-Steering Mirror Assembly from Time Response Data", Proceedings of the Fourth International Conference on Microelectronics, Computing and Communication Systems. Lecture Notes in Electrical Engineering, vol 673. Springer, Singapore.  
[https://doi.org/10.1007/978-981-15-5546-6\\_57](https://doi.org/10.1007/978-981-15-5546-6_57)
- [2] Mahua\_Pal<sup>1</sup>, Anusree Das<sup>2</sup>, Kumardeb Banerjee<sup>3</sup>, Bivas Dam<sup>3</sup> (2023), "Modelling of Fast Steering mirror assembly: A review of the various methodologies", Optik (Elsevier): Volume 287  
<https://doi.org/10.1016/j.ijleo.2023.171108>
- [3] Mahua\_Pal<sup>1</sup>, Kumardeb Banerjee<sup>2</sup>, Bivas Dam<sup>2</sup> (2023) "Black Box Modelling of Dual Axis Fast Steering Mirror Assembly", American Journal of Signal Processing, Vol .12(1): 1-10  
[doi:10.5923/j.ajsp.20231201.01](https://doi.org/10.5923/j.ajsp.20231201.01)
- [4] Mahua\_Pal<sup>1</sup>, Kumardeb Banerjee<sup>2</sup>, Bivas Dam<sup>2</sup>, "Using Constrained Convex Optimization in Parameter Estimation of Process Dynamics with Dead Time" Journal of ASME Letters in Dynamic System and Control. April 2024, 4(2): (11 pages)  
<https://doi.org/10.1115/1.4064770>
- [5] Mahua\_Pal<sup>1</sup>, Kumardeb Banerjee<sup>2</sup>, Bivas Dam<sup>2</sup>, "Efficient Parameter Estimation for Second Order Plus Dead Time Systems in Process Plant Control" Journal of Wiley Advanced Control for Applications: Engineering and Industrial Systems July 2024, 6(3):(23 pages)  
<https://doi.org/10.1002/adc2.229>

**4. List of Patents: None**

**5. List of Presentations in National/International: [1]**



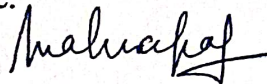
### "Statement of Originality"

I MahuaPal registered on 02/02/2017 do hereby declare that the thesis entitled "Modelling and Control of Second Order Plus Dead Time (SOPDT) Systems" contains literature survey and original research work done by undersigned candidate as part of Doctoral studies.

All information in the thesis have been obtained and presented in accordance with existing academic rules and ethical conduct. I declare that, as required by these rules and conduct, I have fully cited and referred all materials and results that are not original to this work.

I also declare that I have checked this thesis as per the "Policy on Anti Plagiarism, Jadavpur University, 2019", and the level of similarity as checked by iThenticate software is 5%.

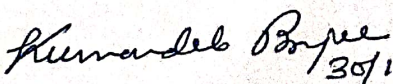
Signature of candidate:



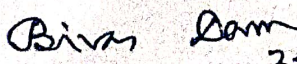
Date: 30/01/24

Certified by Supervisor(s):

(Signature with date, seal)

1.  30/1/2024.

Professor  
Dept. of Inst. & Electronics Engg.  
Jadavpur University  
Salt Lake Campus  
Kolkata-98

2.  30/1/2024

Professor  
Dept. of Inst. & Electronics Engg.  
Jadavpur University  
Salt Lake Campus  
Kolkata-98



## CERTIFICATE FROM THE SUPERVISORS

This is to certify that the thesis entitled "Modelling and Control of Second Order Plus Dead Time (SOPDT) Systems" submitted by Smt. Mahua Pal, who got her name registered on 02/01/2017 for the award of Ph.D. (Engg.) degree of Jadavpur University is absolutely based upon her own work under the supervision of Prof. Kumardeb Banerjee and Prof. Bivas Dam and neither her thesis nor any part of the thesis has been submitted for any degree/diploma or any other academic award anywhere before.

1. Kumardeb Banerjee 20/1/2024

Signature of the Supervisor and date  
with office seal

Professor  
Dept. of Inst. & Electronics Engg.  
Jadavpur University  
Salt Lake Campus  
Kolkata-98

2. Bivas Dam 27/1/2024

Signature of the Supervisor and date  
with office seal

Professor  
Dept. of Inst. & Electronics Engg.  
Jadavpur University  
Salt Lake Campus  
Kolkata-98

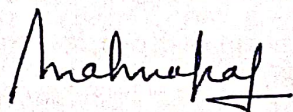


## ACKNOWLEDGEMENT

At the outset, I would acknowledge the patient efforts of my supervisors Prof. Kumardeb Banerjee and Prof. Bivas Dam in deciding upon the problem that finally has led to this thesis. Without their guidance and suggestions, it would not have been possible to come out with this research work. The invaluable feedback and encouragement of Prof. Kumardeb Banerjee greatly influenced how I conducted my experiments and interpreted my findings. I would like to thank the Faculty of Instrumentation and Electronics Engineering Department at Jadavpur University Salt Lake Campus for providing me with the resources to pursue my research work.

I'd also like to thank everyone who has been there for me emotionally and intellectually as I've worked on my research work.

Signature of candidate:



Date: 30/01/24



## **Abstract**

This thesis provides an in-depth investigation into the modelling and control of second-order systems plus dead time (SOPDT), a prevalent configuration in diverse engineering applications. Ranging from chemical processes to electrical circuits and mechanical systems, SOPDT systems pose challenges that this study addresses comprehensively. The primary objective is to develop accurate models and robust controllers ensuring stability, performance, and reliability in SOPDT systems.

The thesis begins by constructing various mathematical models for SOPDT systems, encompassing analytical, numerical, and empirical representations of underdamped, overdamped, and critically damped SOPDT systems. Evaluation and comparison of these models occur through simulation and experimental studies, considering the impact of key system parameters on behaviour. A case study on the time and frequency domain modelling of a fast steering mirror assembly is included.

Subsequently, diverse control strategies are devised for SOPDT systems, viz. PID control, fuzzy logic control, and Genetic Algorithm-based optimal control. The study assesses the effectiveness and robustness of these strategies through simulation and experimentation, analyzing the influence of controller parameters on performance.

Finally, the developed models and control strategies are applied to a temperature controller setup (FEEDBACK, Trainer 37-100). Results demonstrate the practicality and efficacy of the proposed models and control strategies in real-world applications.

In summary, this thesis offers a comprehensive exploration of the intricacies surrounding the modelling and control of SOPDT systems. Intended as a valuable resource for researchers and professionals across various engineering domains, the proposed models and control strategies have the potential to contribute significantly to the development of efficient and robust control systems for SOPDT setups, thereby enhancing their overall performance, reliability, and safety.



# Contents

## Chapter 1: Introduction

1.1 Prologue-----	13
1.2 Motivation-----	14
1.3 Objective-----	15
1.4 Contributions of the work-----	16
1.5 Organization of the Thesis-----	17

## Chapter 2: Related Works

2.1 Modelling of second order plus dead time system-----	20
2.2 Modelling of Fast Steering Mirror Assembly-----	24
2.2.1 White Box Modelling-----	26
2.2.2 Black Box Modelling-----	26
2.2.3 Grey Box Modelling-----	27
2.3 Intelligent Controllers: Fuzzy Logic and Genetic Algorithm-based PID tuning-----	27
2.4 Control of Trainer 37-100-----	30

## Chapter 3: Modelling Methodologies of Second-order Plus Dead Time Systems

3.1 Introduction-----	34
3.2 Modelling of underdamped SOPDT systems-----	36
3.2.1 The Preliminaries-----	36
3.2.2 Mathematical development related to the proposed algorithm-----	37
3.2.3 The gradient-based optimization algorithm-----	40
3.2.4 Proof of convexity of the objective function-----	41
3.2.5 The Proposed Algorithm-----	42
3.2.6 Results and analysis of the algorithm-----	45
3.2.6.1 Sensitivity of the model and global convergence-----	45
3.2.6.2 Comparison of the performance index of the proposed algorithm with two existing non-iterative methods-----	46
3.2.6.3 Parameter estimation of the SOPDT system in the presence of measurement noise and comparison with time domain and frequency domain methods-----	48
3.2.6.4 Comparison of the performance index of the proposed algorithm with the existing iterative method-----	50
3.2.6.5 Cross-validation of the proposed method on other higher-order underdamped system-----	52
3.2.6.6 Fast Convergence-----	54
3.2.6.7 Acquisition of transient time response data-----	55
3.2.7 Conclusion-----	57
3.3 Modelling of SOPDT overdamped/critically damped systems-----	56
3.3.1 Problem Formulation-----	56
3.3.2 Estimation Procedure-----	56
3.3.2.1 Estimation of damping ratio $\xi$ (dead time $t_d \neq 0$ )-----	60
3.3.2.2 Estimation of undamped natural frequency $\omega_n$ (when dead time $t_d \neq 0$ )-----	61
3.3.2.3 Measurement of dead time $t_d$ -----	63
3.3.2.4 Estimation of $\xi$ and $\omega_n$ (dead time $t_d = 0$ )-----	64
3.3.3 The Proposed Algorithm for overdamped/critically damped system-----	64
3.3.4 Validation of the algorithm-----	64
3.3.5 Effect of noise-----	70
3.3.6 Conclusion-----	74



<b>Chapter 4: Modelling of Fast Steering Mirror Assembly - A Case Study</b>	
4.1 Introduction-----	76
4.2 Problem Statement-----	76
4.3 Estimation Procedure-----	78
4.3.1 Frequency Response (FR) Method-----	78
4.3.2 Time Response (TR) Method-----	80
4.4 Experimental Set-up-----	81
4.4.1 Time Response (TR) Method-----	82
4.4.2 Frequency Response (FR) Method-----	86
4.4.3 Cross-coupling-----	87
4.5 Model comparison and validation-----	90
4.5.1 Parameter estimation-based FR method for SOPDT system with strong damping-----	90
4.5.2 Comparison of TR and FR Method with Subspace Identification Method-----	91
4.6 Conclusion -----	92
 <b>Chapter 5: Fuzzy Logic Control</b>	
5.1 Introduction-----	94
5.2 Design of Fuzzy Logic Controller for SOPDT system-----	95
5.2.1. Rule Base-----	96
5.2.2 Fuzzification-----	96
5.2.3 Inference Engine-----	97
5.2.4. Defuzzification-----	97
5.2.5 Membership functions and linguistic variables-----	97
5.3 The Fuzzy Logic Algorithm-----	98
5.3.1 Detailed Design Considerations-----	98
5.3.2 Tuning Parameters -Scaling Factors (SF), membership functions (MF), and rules-----	102
5.4 Implementation of PI-FLC in MATLAB & Simulink environment-----	104
5.5 Comparison of Fuzzy tuned PI control and Ziegler Nichols tuned PI Control for the SOPDT system under measurement noise-----	111
5.5.1 Comparison of Fuzzy controller performance under external disturbance-----	112
5.6 Summary-----	114
 <b>Chapter 6: Genetic Algorithm-based PID control</b>	
6.1 Introduction-----	117
6.2 Genetic Algorithm: - critical factors-----	117
6.2.1 Choice of fitness function -----	117
6.2.2 Encoding-----	119
6.2.3 Population-----	119
6.2.4 Reproduction-----	119
6.2.5 Crossover (Recombination) -----	120
6.2.6 Mutation-----	120
6.2.7 Replacement-----	120
6.2.8 Termination/ Solution Extraction-----	120
6.2.9 The Genetic Algorithm-----	120
6.3 Comparison of GA-tuned PID control and Ziegler Nichols-tuned PI Control for SOPDT system-----	121
6.4 Comparison of GA-tuned PID control and Ziegler Nichols-tuned PI Control for SOPDT system under measurement noise-----	124
6.5 Self-Adaptive Genetic Algorithm-----	128
6.5.1 Choice of the cost function for self-adaptive GA-----	128
6.5.2 Comparison of controller performance under external disturbance-----	129
6.6 Summary-----	132

<b>Chapter 7: Modelling and Control of Trainer 37-100-A Case Study</b>	
7.1 Introduction-----	134
7.2 Process Trainer 37-100-----	135
7.2.1 Open loop test-----	137
7.2.2 System identification and modelling of Trainer 37-100-----	138
7.3 Design of Fuzzy logic Controller for the Trainer 37-100-----	141
7.3.1 Implementation of PI-FLC in MATLAB & Simulink environment-----	141
7.4 Design of Genetic Algorithm-based PID Controller for the Trainer 37-100-----	150
7.4.1 Sensitivity and Stability Analysis-----	153
7.5 Effect of Measurement Noise-----	155
7.5.1 Comparison of Fuzzy tuned PI (PI-FLC) control and Ziegler Nichols tuned PI (ZN-PI) Control for the SOPDT system under measurement noise-----	155
7.5.2 Comparison of GA-tuned PID control and Ziegler Nichols-tuned PI Control for the Trainer Model under measurement noise-----	157
7.6 Comparison of Self tuned Fuzzy-PI Controller and Self Adaptive GA based PID Controller-----	160
7.6.1 Comparison of controller performance under external disturbance-----	162
7.7 Factors that contribute to the ability of FLC to handle noise effectively-----	164
7.8 Summary-----	164
<b>Chapter 8: Conclusion and Future Work</b>	
8.1 Summary of Contributions-----	167
8.2 Future Research-----	167
<b>References-----</b>	<b>169</b>



## List of Figures

- Fig. 3.1      Second-order LP Butterworth Filter used in the experiment
- Fig. 3.2      Universe of discourse of the parameters to be estimated
- Fig. 3.3      Raw Input-Output of the plant under test
- Fig. 3.4      Plot of objective function under step input with **a)** constant  $t_d$  **b)** with constant  $\omega_n$  **c)** with constant  $\xi$
- Fig. 3.5      Comparison of step response by different methods  $t_d = 0.0006 \text{ sec}$ ,  $\xi = 0.5$ ,  $\omega_n = 5000 \frac{\text{rad}}{\text{sec}}$
- Fig. 3.6      a) Model responses for the system with  $t_d = 0.004 \text{ sec}$ ,  $\xi = 0.6$ ,  $\omega_n = 500 \frac{\text{rad}}{\text{sec}}$  under measurement noise (Proposed method, Time domain methods of Rangiah method and Tsung)
- b) Model responses for the system with  $t_d = 0.004 \text{ sec}$ ,  $\xi = 0.6$ ,  $\omega_n = 500 \frac{\text{rad}}{\text{sec}}$  under measurement noise (Proposed method and Frequency domain method of Cheres)
- Fig. 3.7      Model responses for
- a) third order system with a complex conjugate dominant pole pair (system has  $t_d = 0.004 \text{ sec}$ ,  $\xi = 0.6$ ,  $\omega_n = 500 \text{ rad/sec}$ )
- b) fourth order system with two complex conjugat pole pairs, one dominant and other away from the origin (system has  $t_d = 0.02 \text{ sec}$ ,  $\xi = 0.2$ ,  $\omega_n = 2 \text{ rad/sec}$ )
- Fig. 4.1      State Space Model of dual axis FSM assembly
- Fig. 4.2      Experimental Set-up
- Fig. 4.3      Open loop FSM output with an input of 0-2V square wave of freq. 15Hz acquired in DSO
- Fig. 4.4      Bode plots of FSM model and actual FSM unit
- Fig. 4.5      FR Model Response and TR Actual Response for 15Hz

- Fig. 4.6      Cross-Coupling at 15 Hz
- Fig. 4.7      Cross-Coupling at 10 Hz
- Fig. 4.8      Cross-Coupling at 5 Hz
- Fig. 5.1      Block diagram of FLC
- Fig. 5.2      The basic control block diagram illustration with PI type FLC
- Fig. 5.3      Membership function with linguistic labels in FLC
- Fig. 5.4      Variation of output with inputs
- Fig. 5.5      a) Step response of a SOPDT system  
b) Rule base to control the tuning
- Fig. 5.6      Tuning SF affects all rules
- Fig. 5.7      a) Tuning peak or width of MF of if-part of variable  
b) Tuning peak or width of then-part of variable
- Fig. 5.8      Tuning a rule
- Fig. 5.9      Membership function for Error (E)
- Fig. 5.10     Membership function for change of Error ( $\Delta E$ )
- Fig. 5.11     Membership function for change of control output ( $\Delta U$ )
- Fig. 5.12     Implementation of ZN-PI and PI-FLC in Simulink
- Fig. 5.13     (a) Block diagram of self-tuned FLC  
b) Rule base for output scaling factor alpha



(c) Variation of alpha with error and change of error

- Fig. 5.14 Membership function for output scaling factor alpha
- Fig. 5.15 Implementation of ZN-PI and self-tuned PI-FLC in Simulink
- Fig. 5.16 Step response by ZN-PI and PI-FLC and self-tuned PI-FLC
- Fig. 5.17 Response of ZN-PI and Fuzzy-PI controller of the SOPDT system under measurement noise
- Fig. 5.18 Closed Loop system with external disturbance affecting the plant
- Fig.5.19 Comparison of the responses generated by the different methods( ZN-PI and Self-tuned fuzzy) under external disturbance
- Fig. 6.1 Simulation in Simulink for evaluation of cost function when  $K_p = K_i = K_d=0.5$
- Fig. 6.2 Unit step response using Ziegler Nichols and GA with different cost functions (IAE, ITAE and ISE)
- Fig. 6.3 Response of ZN-PI and GA-ITAE controller of the SOPDT system under measurement noise
- Fig. 6.4 Response of ZN-PI and GA-IAE controller of the SOPDT system under measurement noise
- Fig. 6.5 Response of ZN-PI and GA-ISE controller of the SOPDT system under measurement noise
- Fig. 6.6 a) Step input with measurement noise  
b) Response of ZN-PI and GA controllers(ITAE, IAE and ISE) of the SOPDT system under measurement noise
- Fig. 6.7 Block diagram of self-adaptive GA controller
- Fig. 6.8 Closed Loop system with external disturbance affecting the plant
- Fig. 6.9 Comparison of the responses generated by the different methods (ZN-PI and Self-adaptive GA) under external disturbance

- Fig. 7.1      Process Trainer 37-100
- Fig. 7.2      Open loop configuration of the process
- Fig. 7.3      a) Experimental set-up  
b) Acquired signal in DSO: Ch1 is the measured signal and Ch2 is the input signal
- Fig. 7.4      a) Inflection point identified from acquired output (.CSV file) read in MATLAB  
b) Comparison of the responses by FOPDT model and SOPDT overdamped model with response
- Fig. 7.5      The basic control block diagram illustration with PI type FLC
- Fig. 7.6      a) Membership function with linguistic labels in FLC  
b) Fuzzy Rules for Computation of  $\Delta U$
- Fig. 7.7      Variation of output with inputs
- Fig. 7.8      Membership function for Error(E)
- Fig. 7.9      Membership function for change of Error( $\Delta E$ )
- Fig. 7.10     Membership function for change of control output( $\Delta U$ )
- Fig. 7.11     Implementation of ZN-PI and PI-FLC in Simulink
- Fig. 7.12     a) Rule base for output scaling factor alpha  
b) Variation of alpha with error and change of error
- Fig. 7.13     Membership function for output scaling factor alpha
- Fig. 7.14     a) Implementation of ZN-PI and self-tuned PI-FLC in Simulink  
b) Implementation of PI-FLC and self-tuned PI-FLC in Simulink with display of IAE, ISE and ITAE values



- Fig. 7.15 Comparison of the responses generated by the different methods( PI-ZN and Fuzzy)
- Fig. 7.16 Simulation in Simulink for evaluation of cost function when  $K_p = K_i = K_d=0.5$
- Fig. 7.17 Response of Ziegler Nichols and GA with different cost functions (IAE, ITAE and ISE)
- Fig. 7.18 a) Nyquist Plot for GA-ITAE  
b) Nyquist Plot for GA-IAE  
c) Nyquist Plot for GA-ISE  
d) Nyquist Plot for ZN-PI
- Fig. 7.19 Closed Loop system with measurement noise added
- Fig. 7.20 Response of ZN-PI, PI-FLC and self-tuned PI-FLC under measurement noise (white Gaussian noise with zero mean and standard deviation 8% of steady state value and sampling time 0.01 sec)
- Fig. 7.21 Response of ZN-PI and GA-PID (using ITAE cost-function) under measurement noise
- Fig. 7.22 Response of ZN-PI and GA-PID (using IAE cost-function) under measurement noise
- Fig. 7.23 Response of ZN-PI and GA-PID (using ISE cost-function) under measurement noise
- Fig. 7.24 a) Input b) Response of Ziegler Nichols and GA with different cost functions (IAE, ITAE and ISE) under measurement noise
- Fig. 7.25 Comparison of the responses generated by the different methods (PI-ZN and Self-tuned Fuzzy and Self-adaptive GA)
- Fig. 7.26 Comparison of the responses generated by the different methods( PI-ZN and Self-tuned Fuzzy and Self-adaptive GA) under measurement noise
- Fig. 7.27 Closed Loop System with external disturbance  $d$  affecting the plant
- Fig. 7.28 Comparison of the responses generated by the different methods( PI-ZN and Self-tuned Fuzzy and Self-adaptive GA) under external disturbance

### List of Tables

Table 3.1	Estimated model parameters and IAE (of real time SOPDT system (LP Butterworth filter with input delay block))
Table 3.2	Comparative study
Table 3.3	Comparative study of SOPDT system (with measurement noise) under Proposed method with Rangiah [110], Tsung [109] and Cheres[24] method
Table 3.4	a) Comparative study of the Proposed method with Cox Method [35]  b) Convergence Comparison of the Proposed method with Cox Method [35] for the system with $\omega_n=10$ rad/sec, $\xi=0.3$ , $t_d=0.007$ sec.
Table 3.5	Comparative study for higher order systems) Proposed method with Rangiah [110] and Tsung [109] (system gain $K=1$ , for original system and estimated methods)
Table 3.6	Identification results for the system under test (real-time underdamped SOPDT system (LP Butterworth filter)) (N=300 sample points) by the proposed method
Table 3.7	Variation of $\xi$ with $T_R$ & $f_r$ for three different values of $\omega_n$ with $t_d = 0.2$ sec
Table 3.8	Variation of $\xi$ with $T_R$ & $f_r$ for three different values of $\omega_n$ with $t_d = 0$ sec
Table 3.9	Time for responses, estimated parameters, IAE and VAF for Example1
Table 3.10	Time for responses, estimated parameters, IAE and VAF for Example2
Table 3.11	Comparative Study using General Identification Algorithm
Table 3.12	Comparative Study using noisy process response

Table 4.1	Time Response data
Table 4.2	Estimated System Parameters
Table 4.3	Error between Actual and Model responses
Table 4.4	Cross-Coupling Coefficient
Table 4.5	Comparison list of parameter estimates
Table 4.6	VAF Evaluation
Table 5.1	Fuzzy Rules for Computation of $\Delta U$
Table 5.2	Comparison of performance indices for SOPDT system
Table 5.3	Time domain Performance Criteria
Table 5.4	Comparison of performance index for SOPDT system with measurement noise
Table 5.5	Comparison of Time Domain Performance Criteria (self-tuned fuzzy with ZN-PI) under external disturbance
Table 5.6	Comparison of performance index for SOPDT system with and without disturbance (self-tuned fuzzy with ZN-PI)
Table 6.1	Optimal tuning vlues of $K_p, K_i, K_d$ for different cost functions
Table 6.2	Time domain Performance Criteria
Table 6.3	Comparison of performance index for SOPDT system



Table 6.4	Comparison of performance index for SOPDT system with measurement noise (mean zero and standard deviation 8% of steady state value)
Table 6.5	Comparison of Time Domain Performance Criteria (self-adaptive GA with ZN-PI) under external disturbance
Table 6.6	Comparison of performance index for SOPDT system with and without disturbance (self-adaptive GA with ZN-PI)
Table 7.1	Estimation of $T_R$ and $f_r$
Table 7.2	Estimation of the parameters of the system
Table 7.3	Comparison of IAE and ISE of the estimated models
Table 7.4	Comparison of performance index for the Trainer Model
Table 7.5	Time domain Performance Criteria for FLC and ZN PI tuned system
Table 7.6	Optimal tuning values of $K_p, K_i, K_d$ for different cost functions
Table 7.7	Time domain Performance Criteria for GA-tuned PID and ZN-PI-tuned system
Table 7.8	Sensitivity and stability comparison
Table 7.9	Comparison of performance index for SOPDT system with measurement noise
Table 7.10	Comparison of performance index for SOPDT system with measurement noise and the tuning parameters

Table 7.11      Comparison of Time Domain Performance Criteria

Table 7.12      Comparison of performance index for SOPDT system with and without measurement noise (self-tuned FLC and self-adaptive GA with ZN-PI)

Table 7.13      Comparison of Time Domain Performance Criteria (self-tuned FLC and self-adaptive GA with ZN-PI) under external disturbance

# **Chapter 1**

---

## **Introduction**



## 1.1 Prologue

This thesis presents a comprehensive study on the modelling and control of second-order systems plus dead time (SOPDT). Such systems are prevalent in many engineering applications, ranging from chemical process control to electrical circuits and mechanical systems, where higher-order systems are modelled as equivalent SOPDT for the design and analysis of the controller. This study aims to develop efficient and accurate models for SOPDT systems and design robust controllers that can ensure stability, performance, and robustness in these systems.

First, various mathematical models are developed for SOPDT systems, including analytical, numerical, and empirical models of underdamped, overdamped, and critically damped SOPDT systems. The accuracy and applicability of these models are evaluated and compared through simulation and experimental studies. Furthermore, the effects of system parameters such as gain, time constants, damping coefficients and time delay on the system's behaviour are analysed in detail. As a case study modelling of fast steering mirror assembly is performed in the time domain and frequency domain.

Next, various control strategies are developed for SOPDT systems, including PID control, Smith Predictor control, fuzzy logic control and Genetic algorithm-based optimal control. The effectiveness and robustness of these control strategies are evaluated and compared through simulation and experimental studies. Moreover, the effects of controller parameters such as gain, bandwidth, and time constants on the controller's performance are analyzed in detail.

Finally, the developed models and control strategies are applied to a temperature controller (FEEDBACK Trainer 37-100). The results demonstrate the effectiveness and practicality of the proposed models and control strategies in these applications.

Overall, this thesis delivers a detailed exploration of the modelling and control intricacies of SOPDT systems. Attempts were made to make it an exhaustive study, catering to the needs of researchers and professionals across various engineering domains. The proposed models and control strategies can contribute to the development of efficient and robust control systems for SOPDT systems, which can enhance their performance, reliability, and safety.

## 1.2 Motivation

The motivation behind the study of "Modelling and Control of Second Order Plus Dead Time (SOPDT) systems" stems from the need to effectively design and optimize control strategies for real-world processes that exhibit certain dynamic characteristics. SOPDT systems are a common class of dynamic systems encountered in various engineering fields, including chemical, mechanical, and electrical engineering. Understanding and mastering the modelling and control techniques for SOPDT systems can have the following practical implications and benefits.

- ***Representative of many Real-World Processes:*** Many real-world processes, particularly in industries such as chemical engineering, biotechnology, and manufacturing, exhibit higher-order dynamics combined with time delays. The SOPDT model for similar plants provides a reasonably accurate representation of them and is simpler to analyze and plan necessary control actions.
- ***Challenging Dynamics:*** The combination of second-order dynamics and time delays makes SOPDT systems more challenging to control. Hence, mastering control strategies for SOPDT systems is crucial for ensuring stability, performance, and reliability in real-time applications.
- ***Optimization and Performance:*** Effective control of SOPDT systems is essential for optimizing performance metrics such as speed, efficiency, and accuracy in real-time operations. For instance, in chemical processes, precise control of temperature, pressure, and flow rates can significantly impact product quality and yield.
- ***Adaptability to Changes:*** Real-time systems often encounter variations in operating conditions, disturbances, and uncertainties. The ability to model and control SOPDT systems allows for adaptive control strategies that can adjust to changing conditions, ensuring consistent and reliable performance.
- ***Innovation and Advancements:*** Research and development in fields like robotics, automation, and renewable energy rely on sophisticated control strategies for complex systems. The SOPDT model serves as a foundation for developing innovative control techniques and technologies that drive advancements in various industries.
- ***Improved Process Understanding:*** Developing accurate mathematical models for SOPDT systems enables engineers to gain a deeper understanding of the underlying dynamics and behaviour of these systems. By identifying the key parameters and

*characteristics, engineers can make informed decisions about system design, performance optimization, and troubleshooting.*

In summary, the motivation behind the study of "Modelling and Control of Second Order Plus Dead Time (SOPDT) systems" lies in the pursuit of better understanding, improved control performance, and practical applications in various engineering domains. By developing accurate models and effective control strategies for SOPDT systems, engineers and researchers aim to optimize system behaviour, increase productivity, and achieve desired performance objectives in industrial processes.

### **1.3 Objective**

The main challenge in dealing with SOPDT systems lies in capturing their inherent characteristics, which include second-order dynamics and dead time. These systems often arise in processes such as chemical reactors, temperature control systems, and level control systems, where accurate modelling and control are crucial for achieving desired performance and stability.

The problem at hand involves developing a mathematical model that accurately represents the behaviour of the SOPDT system. This model should incorporate the time delay, second-order dynamics, and other relevant parameters that affect the system's response. Additionally, the model should account for uncertainties and disturbances present in real-world applications.

Once the modelling aspect is addressed, the project aims to design a control strategy to regulate the SOPDT system's output variables. The control system should be able to handle the inherent time delay and effectively compensate for the system dynamics. The control strategy should ensure stability, good disturbance rejection, and fast response time while accounting for uncertainties and maintaining desired performance under varying operating conditions.

To achieve these objectives, this project will explore various approaches such as analytical modelling techniques and advanced control algorithms. The effectiveness of the proposed modelling and control strategies will be evaluated through simulations and experimental implementation on a real-world SOPDT system.

Overall, this project aims to contribute to the advancement of modelling and control techniques for SOPDT systems, with the ultimate goal of improving the performance, stability, and robustness of industrial processes that exhibit such dynamics.



## 1.4 Contributions of the Work

The contributions of the thesis titled "Modelling and Control of Second Order Plus Dead Time (SOPDT) Systems" are as follows:

1. **Focus on SOPDT Systems:** *The thesis specifically addresses the modelling and control challenges associated with Second Order Plus Dead Time (SOPDT) systems as the majority of the process control plants, which are essentially higher-order systems, are modelled as SOPDTs. It recognizes the complexity of these systems and aims to develop effective techniques for accurate modelling and subsequent control.*
2. **Case Study:** *The thesis includes a detailed case study of a fast-steering mirror assembly along with the associated drive electronics, a system typically modelled as a SOPDT system. This case study provides a practical context for the application of the proposed modelling and control techniques. It allows for a real-world evaluation of the effectiveness of the strategies developed in the thesis.*
3. **Mathematical Modelling:** *The thesis proposes two different methodologies for the development of a mathematical model that accurately represents the behaviour of SOPDT systems – one for the underdamped systems and the other for the critically/overdamped system. It takes into account the second-order dynamics, dead time, and other relevant parameters specific to the fast-steering mirror assembly. The modelling approaches are tested for noisy systems as commonly encountered in real-world situations.*
4. **Selection of Control Strategy for SOPDT systems:** *The existing control strategies, under the purview of both conventional and intelligent control, are examined. Two control schemes, viz. fuzzy control and GA-based PID control, are selected and their variants are tested on a simulation platform for typical SOPDT systems. The performance analysis of the proposed controllers, taking into account standard performance indices, is done in the time domain. The performance of the controllers is also evaluated in the presence of measurement noise and disturbances.*
5. **Practical Case Study:** *The thesis investigates the efficacy of the proposed modelling and control strategies on a Temperature Controller Kit (FEEDBACK Trainer 37-1000), a typical SOPDT system. The proposed controllers are plugged in with the plant model and their performances are evaluated and compared with controller implementations for the same trainer kit.*

Thus the thesis aims to make a significant contribution to the field of modelling of SOPDT systems and selecting intelligent control strategies for them. It proposes two methodologies for the modelling of challenges specific to generic SOPDT systems, documents the modelling of a fast-steering mirror assembly, investigates two intelligent control strategies for generic SOPDT systems and finally investigates the efficacy of the said controllers for a typical temperature controller setup.

## 1.5 Organization of the Thesis

**Chapter 1** provides a summary of my work as well as an overview of the industrial applications of second-order plus dead time (SOPDT) systems that provide the impetus for my research. An outline is provided, emphasizing the importance of mathematical modelling and a robust control strategy. Shortcomings and challenges encountered in experiments related to control and communication systems have served as the impetus for my research. The pursuit of my research work is driven by the need to address these challenges.

**Chapter 2** provides an overview of existing research related to the issue outlined in Chapter 1, along with an exploration of the motivation behind the work. The chapter delves into a comprehensive examination of pertinent methodologies, pinpointing areas that remain open for exploration. Clear research objectives are articulated, and a critical review is undertaken on modelling approaches, including a case study on the fast steering mirror assembly. Additionally, the chapter conducts a thorough review of control strategies for Second-Order Plus Dead Time (SOPDT) systems.

**Chapter 3** of the study is divided into three sections. Initially, it offers an in-depth review of parameter estimation with Second Order Plus Dead Time (SOPDT) systems. Following this, the second part introduces a comprehensive methodology that can effectively model all systems that are either overdamped or critically damped within the SOPDT framework. The final segment of this chapter outlines a specialized modelling technique tailored for underdamped SOPDT systems. Notably, the simulation results provided in this chapter serve to underscore both the efficacy of the proposed method and its resilience in the face of potential measurement noise.

**Chapter 4** In this chapter a case study is conducted to explore the modelling of a fast-steering mirror (FSM) assembly. The case study aims to develop an accurate mathematical model for the FSM assembly, which is commonly used in optics and imaging systems to compensate for vibrations and disturbances. This model would capture the dynamic behaviour of the FSM assembly, including its second-order dynamics and dead time. Such a model would prove valuable for controller design and performance evaluation purposes.

**In Chapter 5**, the focus lies on the implementation of fuzzy logic control for a Second Order Plus Dead Time (SOPDT) system. This control methodology is meticulously examined and its performance is rigorously assessed, particularly in comparison to the conventional Ziegler-Nichols Proportional-Integral (PI) controller. By leveraging fuzzy logic, the system can handle uncertainties and imprecise inputs, offering a more robust and adaptable control approach. The comparison results shed light on the superior adaptability and robustness of the fuzzy logic controller, emphasizing its potential for enhancing control performance and stability in complex and dynamic systems.

**In Chapter 6**, the primary focus centers on the application of the Genetic Algorithm (GA) for tuning the Proportional-Integral-Derivative (PID) controller for a Second Order Plus Dead Time (SOPDT) system. This tuning methodology, driven by the principles of evolutionary computation, is meticulously investigated and its performance is thoroughly compared with the traditional Ziegler-Nichols Proportional-Integral (PI) controller. By leveraging the power of the Genetic Algorithm, the system can efficiently optimize PID parameters, enhancing control robustness and adaptability to dynamic changes. The comparative results highlight the superior performance of the Genetic Algorithm-based tuning, underscoring its effectiveness in achieving enhanced control precision and stability compared to the conventional Ziegler-Nichols PI controller.

**In Chapter 7**, a case study is conducted by implementing the overdamped modelling methodology from Chapter 2 and the control strategies discussed in Chapters 5 and 6. The real-time system used for this study is Trainer 37-100. The chapter presents results, including a comparative analysis of fuzzy control and Genetic Algorithm (GA) control against conventional Ziegler-Nichols control. The robustness of the system is tested under conditions involving measurement noise and disturbances.

**In Chapter 8**, a comprehensive conclusion is provided, summarizing the key findings and insights derived from the entire thesis. The contributions made through the thesis are highlighted, encompassing both the modelling and control aspects of Second-Order Plus Dead Time (SOPDT) systems. Additionally, the chapter outlines the future scope of the research, identifying avenues for further exploration and development in the realms of both SOPDT system modelling and control. This section serves as a wrap-up, offering a consolidated overview of the work undertaken and paving the way for potential advancements in the field.





## **Chapter 2**

---

### **Related Works**

## 2.1 Modelling of second order plus dead time system

In order to develop a control system, it is necessary to apply physical principles and mathematical modelling to describe its behavior. This involves determining key parameters in a general form. The literature contains a vast array of modelling methodologies, ranging from first-order to higher-order systems, with and without delay time. It is impossible to provide an exhaustive list of all these methods. Instead, this discussion will focus on a selective list of parameter estimation methods that represent the state-of-the-art approaches in control system analysis. Various graphical methods have been proposed in the literature for parameter estimation based on the step response of the system. For example, certain graphical methods (referred to as [1] and [2]) involve drawing tangents to the inflection point of the response curve. The accuracy of these methods heavily relies on precisely locating this sensitive inflection point.

Additionally, integral methods that estimate the area under the response curve have been reported in [3] and [4]. In [4], a least squares method is proposed to determine the parameters of a second-order system with dead time using regression equations, assuming a known steady initial condition. On the other hand, the method presented in [3] is based on the pulse response and is applicable to systems with unsteady initial states.

Another efficient technique for parameter estimation, described in [5]-[7], is the method of moments. This approach requires calculating derivatives of the step response, but it becomes susceptible to noise interference. In the work of [8], a parameter estimation method is introduced that uses four arbitrary data points from the step response curve. However, as mentioned in [6], this method lacks a theoretical foundation.

It is important to note that all these methods either rely on the sensitive inflection point, which can introduce errors, or they lack a theoretical basis for parameter estimation. Furthermore, some methods are vulnerable to noise interference, which affects the accuracy of the estimated parameters.

Dead time is a common characteristic in many industrial processes, and it significantly impacts control systems. To determine the dead time, a popular approach is to approximate it using a rational transfer function, such as Pade's approximation or Laguerre's expansion. These approximations, mentioned in [9], [10], and [11] respectively, provide a way to represent the dead time in the system.

In terms of parameter estimation methods, several existing approaches based on linear regression, as discussed in [12], have made assumptions about the time delay. This is because the delay often appears in a nonlinear manner within the regression equation. These methods involve iteratively estimating the system parameters by updating the value of a small delay through the examination of the phase contribution of the real negative zero present in the corresponding sampled system, as described in [13].

In the literature, there are methods that utilize closed-loop step response curves to approximate the dynamics of a process and evaluate the open-loop parameters of second-order plus dead-time (SOPDT) systems. In the work described in [14], the open-loop process parameters are estimated by conducting closed-loop tests with a step-change in the set point. This method employed a first-order plus dead time (FOPDT) model with a proportional controller and utilized Pade approximation for the delay term. This approach was further refined in [15] by including a functional form approximation for the delay term. Additionally, [16] made further improvements to enhance the accuracy of the method.

However, this method had a couple of drawbacks. Firstly, FOPDT models did not adequately represent many chemical processes. Secondly, the controller used in this approach was limited to a proportional-only controller, whereas industrial controllers often employ proportional-integral (PI) control. To address these limitations, [17] introduced the use of the SOPDT model instead of the FOPDT model for parameter estimation. Furthermore, [18] overcame both drawbacks by utilizing closed-loop transients of the SOPDT model with a PI controller. Despite the advantages of these methods, they do have a significant disadvantage. Determining the tuning parameters of the controller becomes difficult and laborious. This means that additional effort and expertise are required to find suitable values for the controller's tuning parameters, which can be a time-consuming and tedious process.

In a research study referenced as [19], the authors introduced a new approach to estimate the parameters of an overdamped second-order system. They achieved this by parametrizing the second-order transfer function using the dominant pole and the ratio of the fastest pole to the dominant pole. By analyzing the step response at two specific points, they derived an analytical expression to determine the system parameters.

In another study cited as [20], metaheuristic methods such as the Grey Wolf algorithm and Jaya algorithm were employed to estimate the coefficients of transfer functions for second-order systems. These methods utilized a step input with a constant amplitude, and the

convergence rate of the model response to the actual response was improved by applying the final value theorem. These metaheuristic methods were found to be more effective than graphical methods that rely on precise observations. However, it is worth noting that these methods did not address the identification of time delays, which are commonly present in industrial systems.

Additionally, the parameter estimation of a critically damped SOPDT (Second-Order Plus Dead Time) system was proposed in [21]. The researchers utilized a unique property mentioned in [22] specific to such systems. According to this property, the time taken for the system response to reach 73% of the step response is equal to 2.58 times the time constant of the system. This property was leveraged to estimate the parameters of critically damped SOPDT systems.

Several methods also exist in literature where parameter estimation is performed in the frequency domain. In the parameter estimation method [23] the open-loop input-output time-domain data of a linear system is first transformed into the frequency domain using the Fourier transform, and then this transformed data was solved recursively to obtain both system parameters and unknown delay. This recursive non-linear estimation technique was similar to the least-square method in the time domain. In [24], a method was proposed that transformed closed-loop data of a second-order plus dead-time (SOPDT) system into the frequency domain and then used least-square estimation to estimate the parameters of an SOPDT process model. Although robust to noise, the frequency domain methods of parameter estimation require data acquisition and subsequent computer analysis. It becomes a drawback if these means are not available on the plant floor.

In [25], a method is introduced that enhances the estimation of model states in a dual-rate system with time delay. This approach utilizes Kalman filtering and proposes a state augmentation technique. Furthermore, a stochastic gradient algorithm is developed to identify the system's parameters. In [26], a dual-layer optimization approach is presented for a non-linear dynamical system. Initially, a white box model is formulated to provide an approximate description of the system dynamics. Then, an optimization algorithm is employed to search for an optimal set of system parameters, minimizing the error between the optimized model and the white box model. This approach is applied to a given excitation signal.

For a SOPDT (Second-Order Plus Dead Time) system, [27] introduces a control law that iteratively predicts the process output and updates model parameters using Bayesian estimation



[28] procedure. The method estimates the current disturbance effect and predicts the future disturbance effect. The aim is to develop an effective control strategy for SOPDT systems.

The non-linear dynamics of a SOPDT system can be represented by multiple linear models, as proposed by Takagi-Sugeno [29]. This technique is employed in [30] to design a non-linear controller and in [31] to address state and parameter estimation for nonlinear time-varying systems.

In [32] a parameter estimation method is proposed for fractional order systems contaminated by white noise. The method employs a stochastic gradient algorithm to mitigate the impact of the noise and estimate the system's parameters effectively.

In [33] and [34] the focus is on investigating various convex optimization problems that can be utilized to model systems. Specifically, [33] examines a convex optimization problem that involves a time-varying cost function. The objective is to find the optimal trajectory that changes over time. On the other hand, [34] explores a convex optimization problem that incorporates a time-varying Lagrangian function. The aim here is to determine the optimal point based on the changing Lagrangian function. Both studies emphasize the utilization of convex optimization techniques to tackle these dynamic optimization problems and find optimal solutions.

Unlike previous methods [35] and [36] which have taken initial guesses as small positive real numbers that may not always converge to a global solution, the proposed approach provides accurate and reliable parameter estimation, facilitating practical applications in real systems. In a recent study [36], metaheuristic methods like the Grey Wolf and Jaya algorithms were used to estimate coefficients of second-order system transfer functions. These methods outperformed graphical techniques by accelerating convergence rates using the final value theorem. However, they didn't address the common issue of identifying time delays in industrial systems. The work of [37] proposes a new technique for identifying the parameters of a transfer function describing the dynamics of a multi-inertial system. The transfer function is of  $n$ th order, incorporating a single multiple time constant ( $T_n$ ) and a time delay ( $t_n$ ). The method leverages the system's step response ( $h_0(t)$ ) to approximate its characteristics. To facilitate control system simulations and controller tuning, [38] introduces two methods for identifying high-order repeated-pole plus dead-time (RPPDT) models. One method relies on the process reaction curve obtained from an open-loop test, while the other utilizes relay-feedback obtained from a closed-loop test.

## 2.2 Modelling of Fast Steering Mirror Assembly

Fast steering mirrors have a wide range of applications in optical systems, such as beam adjustment between the laser source and receiver, correction of tilt errors in adaptive optical systems [42] transmission, stabilization [43], and precision tracking [44] of laser beam in laser communication systems. The recent state-of-art modelling methodologies of FSM have been reviewed and presented in this paper. For this, a comprehensive taxonomy for the modelling of FSM is first developed. Then this taxonomy is applied to review various approaches for modelling FSM.

Our survey for finding a model of FSM includes (i) knowledge-based modelling (white box modelling), (ii) data-driven modelling (black box modelling), and (iii) process-based modelling (grey box modelling) as classified in [45]. White box modelling is deriving a model from the domain knowledge of the structure of the system. Black box modelling is the parameter estimation of the model from the measured data. Grey box modelling as the name suggests is the combination of the above two approaches i.e., knowledge-based, and data-driven.

In optical communication, precise pointing of laser beam is achieved by FSM which is a mirror mounted on one or more actuators capable of precise and fast movements. There have been numerous research works on the modelling [46-56] of FSM which may be single axis or dual axis. Usually, there are two actuators (piezoelectric [50] or voice coil [48]) for single-axis FSM. The work of [46] proposed a model-based design for a two-axis four-actuator system (TAFA) while [47] used structural knowledge-based modelling for single-axis two actuators (SATA) FSM. In [48] an integrated mechanical-control co-simulation (IMCS) model of TAFA was developed for achieving a specific closed-loop performance criterion. The work of [49] modelled an advanced fast steering system (AFSM) with dual axis driven by flux steering electromagnetic actuators and also proposed a small AFSM which would reduce the device size. A mathematical model was proposed in [50] for dual-axis FSM with three piezoelectric actuators to describe the dynamics, and a hysteresis model was used to represent the hysteresis. Many papers available in the literature, such as [48],[49], and [51] have a key performance criterion of high bandwidth design associated with a stable FSM system and its response speed.

In the work of [52-54] charge couple device-based control of FSM has been presented using multiloop strategy. The popular Spice program was used to model both single-axis and dual-axis FSM [55]. A white box modelling approach to model a two-axis piezoelectric FSM (PFSM) system was proposed in [56] to compensate for hysteresis. In [57]-[61] black-box

modelling approach i.e. input-output behaviour of the FSM assembly was observed and analyzed to derive a mathematical transfer function. The transfer function was then used to simulate the response of the FSM assembly to different input signals and to design control systems that regulate the mirror motion. In [62]-[66] grey box modelling approach is applied to model FSM based on known and unknown parameters of the system. The model was then used to simulate the response of the system to various inputs and disturbances and analyze the performance and stability of the control system.

A high bandwidth FSM was designed in [67] using off-the shelf piezo actuators for closed loop control. The work of [68] proposed a compound FSM driven by both limited-angle voice coil motor and push-pull FSM for high precision beam control. A control strategy combining feedforward control in Kalman filtering with auto disturbance rejection control was then used to improve trajectory and tracking accuracy.

In the presence of mechanical and environmental disturbances, it is difficult to maintain the orientation of the optical sensor in the desired direction, and this gives rise to the issue of getting not so accurate results. Stabilization of the tracking system by using inertially stabilized platforms [69] made up of an assembly of gimbal structure, gyroscope and sensor, is the only solution of this issue. To address the main issues related to FSM design, the work of [70] categorically points out performance/optical requirements that must be taken care of. A good performance criterion demands a wide angular range and high bandwidth simultaneously in the FSM system and as has been pointed out that it is difficult to achieve. The design of a double fast steering mirror [71] integrated the FSM driven by piezo actuators and the FSM driven by the voice coil actuators into a single optical path to achieve high control bandwidth together with wide angular range. To address the issue of cross-coupling between the axis of FSM which affect the tracking precision, the work of [72] designed a miniature fast steering mirror (MFSM) which had kinematic decoupling characteristic. Many modelling methodologies such [62] have used the algorithm in [73] to model FSM. The work [74] presents the design and development of a 2-DOF rotational pointing mechanism integrated with a FSM for laser beam pointing applications using compliant mechanism. A systematic way of modelling the complex hysteresis non-linearity of FSMs using a symmetric hysteresis operator and a cascaded neural network is presented in [75].

### 2.2.1 White Box Modelling

In white box modelling methodology, the output may be predicted when the input variables are subjected to the physical structure of the model and its parameters. White-box models with high accuracy have been possible nowadays due to increased trends of sophistication in computing techniques. Although not universal, this modelling methodology has been followed by most of the researchers [46-51]. There are many researchers whose works aim at CCD (charge-coupled device) based control of FSM [52-54]. In [55] Tian et al. analyzed the multiloop (position, velocity, and acceleration feedback) control structure of FSM to improve disturbance suppression performance. A comprehensive open loop model is proposed in [56] by Wang et al. for a two-axis piezoelectric FSM (PFSM) system, which can characterize various factors affecting the system performance, including the hysteresis, creep effect, mechanical dynamics, the gains of the high-voltage amplifier and strain gauge sensor and cross-coupling.

The proposed identification methodology in the above work analyzes the influence of each component on the output and their interaction on different time scales, resulting in more accurate parameter identification. The optimization problem required for parameter identification may be computationally intensive and require significant computational resources.

### 2.2.2 Black Box Modelling

Black box modelling involves the development of an FSM model without revealing any of its internal features, but using the input and the output data, either already known or available through experimental measurements. This data-driven approach may be used for modelling a full FSM system as in [57], [58] or may be used to model subsystems of an FSM system [59],[61]. The data-driven approach is also useful in online controlling and analysis of the FSM system. A black box modelling approach for modelling FSM is presented in Chapter 4.

In [58] Watkins et al. designed a laser jitter control (LJC) testbed which consisted of two FSMs, three position sensing detectors (PSD) one diode laser and several beam splitters. In [60] Zhang et al. proposed a second order model for single axis FSM taking into account cascade connection of first order electrical dynamics and first order mechanical dynamics of the voice coil motor. In [61] Xia et al. proposed a third order model for FSM with voice coil actuators, integrating the dynamics of mechanical and electrical part of the voice coil motors. The work of [62] Chen et al. presents a MIMO LTI high fidelity model of FSM using frequency domain subspace identification algorithm from frequency response data of the FSM system. An 8<sup>th</sup>-

order state space model is identified, and to compensate for the discrepancy between the experimental frequency response and model frequency response certain blocks are added to the model.

### **2.2.3 Grey Box Modelling**

This approach [62-63], [64-66] formulates a mathematical model, partially from prior knowledge of the physical system and partially, from experimental data. Thus, setting up modelling equations based on physical systems and estimating the unknown parameters in these equations using input-output data requires a high level of expertise.

## **2.3 Intelligent Controllers: Fuzzy Logic and Genetic Algorithm-based PID tuning`**

The PID controller stands as a cornerstone in process control, widely recognized as the workhorse due to its versatility and effectiveness in regulating various industrial systems. However, its performance can face challenges when dealing with systems characterized by dead time. The presence of dead time, or time delays, can lead to degradation in the PID controller's performance. Despite its prominence, the PID controller may encounter limitations in effectively handling processes with inherent dead time, prompting the need for alternative control strategies or enhancements to mitigate such challenges and maintain optimal performance. An alternative approach to PID control involves the utilization of a Smith Predictor (SP), also referred to as a dead time compensator, as introduced by Smith in 1957. The Smith Predictor employs a model of the process without incorporating time delays to predict the process output. Subsequently, this newly predicted process is governed by a conventional controller, such as a proportional-integral (PI) controller. Notably, the SP controller exhibits commendable setpoint response by eliminating internal delays from the closed-loop transfer function. However, a drawback arises in its suboptimal performance when subjected to input disturbances, particularly for processes characterized by sluggish dynamics. This limitation is particularly pronounced in cases involving original open-loop process poles which remain unchanged. Reference [141]-[143] have extensively examined and compared the performance of proportional-integral (PI), proportional-integral-derivative (PID), and Smith Predictor (SP) controllers. These investigations primarily focus on assessing controller performance under load disturbances while maintaining a fixed level of robustness. The collective findings from these studies generally indicate that PID controllers exhibit superior



performance compared to SP controllers. The [144] aligns with these conclusions, further reinforcing the notion that PID controllers outperform SP controllers in handling load disturbances under similar conditions.

The concept of fuzzy set theory was initially introduced by Zadeh as a means to manage and regulate complex systems, particularly those that were challenging to model accurately [76]. Subsequently, Mamdani pioneered the application of fuzzy control [77], leading to the widespread acceptance of fuzzy logic controllers (FLCs) as a highly promising solution for a multitude of nonlinear problems. Recent studies have emphasized that FLCs can serve as effective tools for designing controllers for diverse systems, often outperforming conventional linear PID controllers. They have demonstrated greater resilience and are less sensitive to variations in parameters [78]-[82].

Many practical systems under automatic control are nonlinear higher-order systems with significant dead time. Dead time, recognized as a challenging dynamic element, is inevitable and often changes randomly with ambient conditions or over time [83]. Effective control system design must address this dead-time challenge. For satisfactory performance, the controller output or process input needs to be a nonlinear function of certain variables. Fuzzy Logic Controllers (FLCs) attempt to capture this nonlinearity through a limited set of IF-THEN rules, but this may not always suffice for optimal performance. In such cases, relying solely on static or fixed membership functions and predefined rule sets may not be enough. To overcome this limitation, extensive research has focused on tuning FLCs, adjusting input-output scaling factors or fuzzy set definitions to match current plant characteristics, either online or offline [84], [85], [86]-[90].

Among the different variations of fuzzy logic controllers (FLCs), two commonly utilized types are the PI-FLCs and PD-FLCs. Similar to the widely employed conventional PI controllers in various systems, PI-FLCs are popular due to their ability to combine the stability of proportional (P) controllers with the offset correction feature of integral (I) controllers. These controllers are often deemed effective for managing linear first-order systems. However, their effectiveness diminishes when dealing with higher-order systems containing integrating elements or substantial dead times, and they tend to perform poorly in the context of nonlinear systems, exhibiting tendencies toward significant overshoot and excessive oscillation. In some cases, these systems may become uncontrollable altogether. On the other hand, PD-FLCs, while suitable for a specific set of systems, are cautioned against deployment in scenarios where there is the presence of measurement noise and sudden load disturbances.

An FLC operates using a fixed set of control rules that are typically derived from the expertise of domain specialists. The membership functions (MFs) associated with the input and output linguistic variables are predetermined on a shared universe of discourse. Designing effective FLCs requires careful selection of input and output scaling factors (SFs) and tuning of other controller parameters, often done through trial and error or based on training data. Among the adjustable parameters, SFs hold the utmost importance as they significantly impact the overall operation of the control system [91].

The mathematical framework of genetic algorithms [90] enables us to optimize systems, even when the system's performance is characterized by a complicated, non-linear, and non-smooth function that might have multiple local high points and low points. Unlike traditional optimization techniques, genetic algorithms are direct search algorithms, which means they can navigate through the function without the need to calculate its gradient or higher derivatives. This characteristic is precious when calculating these derivatives is intricate or unfeasible. Consequently, genetic algorithms allow for the efficient identification of optimal solutions in complex and challenging situations. Enhancing the productivity of process plants involves integrating cutting-edge information technologies into control automation systems. A key challenge lies in identifying the specific parameters that facilitate the smooth operation of the automatic control system (ACS) for various technological processes, in alignment with the desired quality benchmarks. This issue necessitates a comprehensive solution during the initial phase of system design [92]-[94], as well as regular adjustments to the ACS parameters to accommodate fluctuating operational conditions within the facility, which directly impact the control precision of the technological parameters.

The PID (Proportional–Integral–Derivative) controller is a commonly employed controller in industrial applications, particularly in the realms of process control and robotics. Auto-tuning of PID controllers is crucial in industries for optimizing performance and adapting to variations in operating conditions. This process involves utilizing the unit step response of the plant and modifying the proportional ( $K_p$ ), integral ( $K_i$ ), and derivative ( $K_d$ ) gains. The primary objective of this auto-tuning is to achieve stability in the plant [95]-[97]. Certain well-established auto-tuning algorithms, as outlined in reference [98], utilize methods such as the relay and Ziegler-Nichols techniques. The Ziegler-Nichols method involves inducing oscillations in the system to extract parameters, specifically the critical gain ( $K_u$ ), and oscillation period ( $T_u$ ), for computing PID gains. However, this oscillatory procedure can stress the plant. To mitigate this, the relay method serves as an alternative. This method involves

limiting the controller's output to induce controlled oscillation in the plant, offering a way to extract parameters without causing undue stress to the system.

An alternative method involves employing artificial intelligence, as highlighted in reference [99]. Fuzzy Logic applications for PID tuning rely on a predefined expert system that utilizes rule tables for each controller parameter, leading to satisfactory outcomes. Additional studies utilizing fuzzy logic [100]-[103] and innovative algorithms like flower pollination [104] have demonstrated success in obtaining optimal PID parameters. The artificial neural network (ANN) has a drawback, it requires a substantial amount of plant data. In contrast, the genetic algorithm (GA) relies solely on the unit-step response. The GA demonstrates superior results compared to the Ziegler-Nichols method in terms of performance [105] [106]. Hence, this study suggests employing the GA to fine-tune the values of the proportional( $K_p$ ), integral( $K_i$ ), and derivative( $K_d$ ) gains. The innovative aspect of this work lies in the selection of the fitness function, specifically tailored to minimize the worst-case value among significant performance metrics such as Integral of Absolute Error (IAE), Integral of Squared Error (ISE), and Integral of Time-weighted Absolute Error (ITAE). This particular choice of fitness function has been demonstrated as a crucial element in attaining improved control performance.

## **2.4 Control of Trainer 37-100**

Achieving the ability to follow varying target values without encountering any steady-state deviation, even when faced with consistent external influences, stands as a highly sought-after feature within a feedback control system. This objective is effectively addressed in traditional control techniques for linear systems through the integration of control actions, often seen in widely used algorithms such as proportional-integral (PI) or proportional-integral-derivative (PID) controllers [112]. Temperature control has been investigated in [112]-[115] due to its paramount importance and critical impact on processes, materials, and comfort in various fields.

The 37-100 Trainer serves as a process simulator device, encompassing a centrifugal blower responsible for propelling air from the surroundings. This air is directed across a heating grid and travels through a tube before being released into the atmosphere. The fundamental objective of the control mechanism is to maintain the air within the tube at a predetermined temperature. Because of its distinctive traits, the Trainer 37-100 has been employed in various research studies. For instance, in [116], the 37-100 is harnessed for predicting system output through machine learning techniques such as Support Vector Machine and Regularization

Networks. In [117], a study is conducted to compare the performance of a generalized predictive control strategy for regulating the temperature of the 37-100 Trainer (previously called PT326 process trainer) with that of a PID control system incorporating auto-tuning techniques. In [118], the Trainer 37-100 (PT326)-based system is controlled using an Adaptive Backstepping Sliding Mode Control (ABSMC) law for an industrial heat exchanger. The work of [119] proposes a data-driven control strategy for steady-state error-free set point tracking. It computes control sequences using past signals from a process historian database. Reference [120] explores multilayer feed-forward neural networks in Simulink/MATLAB, specifically the Neural Network Toolbox NNDT, to control the Trainer 37-100 (PT326) Process Trainer Laboratory equipment.

PID controllers are widely used due to their simplicity and versatility in industries. While more advanced methods of PID tuning [121]-[126] exist, they lack PID's straightforwardness. However, a PID controller tuned for a specific condition might not work well beyond that range. Fuzzy logic techniques are well-documented in the literature and have been widely employed to fine-tune PID (Proportional-Integral-Derivative) controllers. These techniques [127]-[129] leverage the principles of fuzzy logic to adjust the PID controller's parameters in response to varying system conditions. In some cases, non-fuzzy methods are employed to fine-tune fuzzy controllers, as mentioned in [130] and [131]. Conversely, there are instances where fuzzy inference mechanisms are utilized to fine-tune non-fuzzy controllers, as noted in references [132] and [133]. Furthermore, there exist numerous fuzzy controllers, such as those in references [134], [135], and [136] that are tuned using fuzzy inference mechanisms. However, there's a lack of systematic design methods for fuzzy controllers. Typically, a trial-and-error approach is used to set the membership functions and input output-scaling factors, which is time-consuming and often doesn't yield optimal results

Genetic algorithms (GAs) which were originally developed by Holland [137] in the early 1970s often outperform fuzzy logic for PID tuning due to their ability to systematically explore a broader solution space, and without relying on predefined linguistic rules. GAs have become a valuable tool for tackling challenging search, optimization, and machine-learning problems that were previously hard to solve automatically [138]. They offer a fast and dependable approach to addressing complex problems. The work in [139] introduced enhanced GAs for tuning controllers, particularly in systems with nonlinearities. Reference [140] applied a population-based distribution GA to optimize PID controllers and observed improved search capabilities through competition among distribution populations, ultimately reducing the

search area. These GA-based PID controllers deliver strong performance and find practical applications. However, standard GAs do have drawbacks, including premature convergence and slower convergence speeds. The approach proposed here seeks to enhance the PID controller's performance by employing a genetic algorithm that can adjust its parameters in a more flexible and responsive manner, ultimately leading to better control system performance for the temperature control of Trainer 37-100.





## **Chapter 3**

---

# **Modelling Methodologies of Second-order Plus Dead Time Systems**

### 3.1 Introduction

In the realm of control systems engineering, the accurate identification and estimation of system parameters play a pivotal role in designing effective control strategies. One class of systems that frequently appears in industrial processes is the Second Order Plus Dead Time (SOPDT) system. These systems find applications in various engineering domains, such as chemical processes, thermal systems, and mechanical systems.

The accurate determination of SOPDT system parameters is crucial for achieving optimal control performance, system stability, and responsiveness to changes in the environment. Precise identification of these parameters is a challenging task due to the inherent complexities and nonlinearities associated with real-world processes. This study focuses on the development and application of methods for the parameter estimation of SOPDT systems. The objective is to enhance our understanding of the underlying dynamics of such systems and improve the robustness of control strategies. Researchers and engineers alike seek reliable techniques to extract accurate parameter values, enabling the implementation of advanced control algorithms that can adapt to dynamic changes in the system.

Throughout this Chapter, various methodologies employed in SOPDT system parameter estimation is explored. This includes experimental techniques, system identification algorithms, and computational approaches. By addressing the challenges posed by SOPDT system characteristics, this research aims to contribute valuable insights to the broader field of control systems engineering, fostering advancements in industrial automation, process optimization, and overall system performance. The Chapter is structured into two distinct sections. The initial part focuses on the parameter estimation of underdamped systems, elucidating the methodologies and considerations specific to systems exhibiting underdamped characteristics. In contrast, the latter section delves into the parameter estimation of overdamped and critically damped systems, addressing the unique challenges and intricacies associated with these particular types of dynamic systems. This division allows for a comprehensive exploration of parameter estimation techniques tailored to the distinct behaviour exhibited by underdamped and overdamped/critically damped systems, thereby offering a more nuanced understanding of the subject matter.

The proposed convex optimization method of parameter estimation of the underdamped SOPDT system (given in Section 3.2) outperforms existing techniques for SOPDT system

parameter estimation, offering better accuracy, faster convergence, and lower computational complexity. Incorporating system constraints and reliable initial guesses, it ensures more realistic outcomes.

Additionally, (in Section 3.3) a robust and intuitive method for parameter estimation of second-order systems, both with and without dead time, utilizing the process step response, applicable to both overdamped and critically damped system is proposed. This method avoids direct use of the point of inflection and provides a mathematical basis for estimating all the parameters involved including the dead time, without knowledge of the controller and without computational complexity. The Chapter is organized as follows:

The document's Section 3.2 presents a comprehensive explanation of the modelling technique intended for underdamped Second Order Plus Dead Time (SOPDT) systems. This method involves a detailed process or set of procedures for accurately representing and analyzing systems that exhibit underdamped characteristics, along with a specific dead time. To make the paper as self-contained as possible Section 3.2.1 presents the essential propositions related to the Hessian of a function, on which convex optimization principles are grounded. These include notions of convex analysis and the Karush Kuhn Tucker (KKT) conditions. The remaining part of the paper is organized as follows: Section 3.2.2 introduces the mathematical developments related to the proposed algorithm. Section 3.2.3 describes the algorithm using Newton's Method and SQP. Section 3.2.4 gives the proof to verify that cost function chosen is convex.

Section 3.2.5 states the algorithm stepwise. Section 3.2.6 provides the results, analysis of the algorithm through illustrative examples, and its comparison with selected existing methods [109-110]. Section 3.2.7 includes the concluding remarks.

Furthermore, in Section 3.3 of the document, a detailed description of the modelling method for an overdamped SOPDT system is provided. This section elucidates a distinct set of procedures, equations, or strategies that are specifically tailored to the comprehensive modelling and analysis of systems demonstrating characteristics of overdamped behaviour within the context of a Second Order Plus Dead Time system. The Section 3.3.1 gives the problem formulation and the choice of the model for parameter estimation of the SOPDT system. Section 3.3.2 introduces the theoretical and mathematical basis for the estimation procedure. Section 3.3.3 presents the least-square based iterative algorithm for overdamped and critically damped second-order systems with and without dead time. Section 3.3.4 presents validation of the algorithm by comparison with two published methods [6] and [8]. Section 3.3.5

documents the robustness of the proposed method against measurement noise. Finally, the concluding remarks are presented in Section 3.3.6.

## 3.2 Modelling of underdamped SOPDT systems

The main contribution of the method is to propose the formulation of a parameter estimation method of SOPDT systems from its step response, exploring a close tie between the classic gradient-based Newton's method and Sequential Quadratic Programming (SQP) using constrained convex optimization principles. The starting values of the parameters are determined analytically, including the dead time. Starting with the initial guess of the parameters, a Quadratic Programming (QP) sub-problem is solved and an estimate of the Hessian and gradient of the QP is updated iteratively. One of the main advantages of gradient-based optimization algorithms is that they can handle complex, high-dimensional objective functions that have many local optima. These algorithms can also be used for unconstrained and constrained optimization problems. Further, in this work, the objective function is chosen to be convex, which guarantees the convergence of local minima and global minima. The initial guess serves as the starting point for the optimization algorithm, and depending on its proximity to the true optimal solution, it can significantly affect the convergence and final result of the optimization process. If the initial guess is far from the optimal solution, the algorithm may require more iterations to converge, or worse, it may converge to a suboptimal solution. The optimization method used for parameter estimation of Cox [35] and Rui [36] have taken initial guesses as small positive real numbers which may not always converge to a global solution. The proposed optimization methods in this work have determined the initial guess from the analytical method and hence the estimation converges to the true solution with greater accuracy and minimum iterations.

### 3.2.1 The Preliminaries

In this section, we briefly state the propositions related to convex functions and Lagrange multipliers as they are of crucial importance for the parameter-based optimization technique applied here. A function  $f: \mathbb{R}^n \rightarrow \mathbb{R}$  is convex if and only if its curve lies below any chord joining two of its points. Here the domain of function  $f$  (objective cost function) is  $\mathbb{R}^3$ , as  $f$  is a function of three parameters to be estimated, so  $f$  will be described as  $f: \mathbb{R}^3 \rightarrow \mathbb{R}$  in the rest of the article. The proofs of all the Propositions stated below are omitted, as they are basic proofs of multivariable calculus.



*Proposition 1:* A twice differentiable function:  $R^3 \rightarrow R$  is convex if and only if the Hessian  $\nabla^2 f$  is positive semi-definite  $\forall x \in R^3$ . More generally a function  $f$  will be convex if  $\exists$  two values  $a, b \in R^3$  in the feasible region such that

$$f(\lambda a + (1 - \lambda)b) \leq \lambda f(a) + (1 - \lambda)f(b) \text{ where } 0 < \lambda < 1$$

*Proposition 2:* If a function  $f: R^3 \rightarrow R$  is convex and differentiable, then  $x \in R^3$  is a global minimum of  $f$  if and only if  $\nabla f(x) = 0$ .

*Proposition 3:* If functions  $f, g: R^3 \rightarrow R$  are convex then  $f + g$  is also convex.

*Proposition 4:* If a function  $f: R^3 \rightarrow R$  is convex subject to the constraints  $g_i(x) \leq 0$ ,

$i = 1, 2 \dots m$  then the KKT equations [40] stated below in (i-iii) are both necessary and sufficient for a global optimal solution point.

(i)  $\nabla f(x) + \sum_i \mu_i \nabla g_i(x) = 0$  (ii)  $\mu_i g_i(x) = 0$  (iii)  $\mu_i \leq 0$  (for minima) where  $\mu_i$  are the Lagrange multipliers.

### 3.2.2 Mathematical development related to the proposed algorithm

The problem formulation for this work involves utilizing constrained convex optimization principles to estimate parameters accurately for process dynamics with dead time (a real-time second-order LP Butterworth filter with an input delay block) as shown in Figure 3.1.



Figure 3.1 Second-order LP Butterworth Filter used in

the experiment

This section describes the mathematical steps to derive the estimation equations. The governing equation of the underdamped SOPDT model is taken as:

$$G(s) = Ke^{-t_d s} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right) \quad (1)$$

(where  $t_d$  is the dead time,  $\omega_n$  is the undamped natural frequency,  $\xi$  is the damping factor and  $K$  is the gain of the system to be estimated). The analytic unit step response of this system is given as

$$y(t - t_d) = u(t - t_d) \left( 1 - \frac{e^{-\xi\omega_n(t-t_d)}}{\sqrt{1-\xi^2}} \left( \sqrt{1-\xi^2} \cos\omega_d(t-t_d) + \xi \sin\omega_d(t-t_d) \right) \right) \quad (2)$$

where  $\omega_d = \omega_n \sqrt{1-\xi^2}$  is the damped frequency of the system, and  $u(t)$  is the unit step function.

Let the actual step response be denoted by  $y(t)$  and the model step response by  $\hat{y}(t)$ . We define the objective function for the algorithm as  $f_{T_i}(\hat{\theta})$  which is the integral absolute error of the system given as (3)

$$f_{T_i}(\hat{\theta}) = \sum_{T_1}^{T_N} |y(t_i) - \hat{y}(t_i)| \quad (3)$$

In SQP class of algorithms, the objective function  $f_{T_i}(\hat{\theta})$  is assumed to be quadratic, and hence its Hessian will be constant where  $\hat{\theta}$  denotes the unknown parameters ( $\xi$ ,  $\omega_n$ , and  $t_d$ ) to be estimated, and  $N$  is the collected data lengths, for sampled time instants  $T_i$  ( $i = 1, 2 \dots N$ ). The problem objective is to determine the unknown parameters in (1) from the input-output observations  $\{u(t_i), y(t_i)\}$  of the step response satisfying (4)

$$\hat{\theta} = \arg \min_{\hat{\theta}} (f(\hat{\theta})) \quad (4)$$

such that

$$\text{lower bound (lb)} < (\hat{\theta}) < \text{upper bound (ub)} \quad (5)$$

$$\hat{\theta} = [\xi, \omega_n, t_d]^T \quad (6)$$

Combining Equation 4-6 in a single Lagrangian Function [35]

$$L(\hat{\theta}, \lambda, \mu) = f_{T_i}(\hat{\theta}) + \sum_i \lambda_i h_{T_i}(\hat{\theta}) + \sum_i \mu_i g_{T_i}(\hat{\theta}) \quad (7)$$

were the Lagrangian multipliers  $\lambda_i$  are the coefficients of the equality constraints represented by the equation  $h_{T_i}(\hat{\theta}) = 0$  and Lagrangian multipliers  $\mu_i$  are coefficients of the inequality constraints represented by equation  $g_{T_i}(\hat{\theta}) \leq 0$ .

In our case  $\lambda_i = 0$  as there are no equality constraints. Thus from (7)

$$L(\hat{\theta}, \lambda, \mu) = f_{T_i}(\hat{\theta}) + \sum_i \mu_i g_{T_i}(\hat{\theta}) \quad (8)$$

Using Proposition 4, optimality is reached when the KKT conditions [36] are fulfilled, which gives (9)

$$\nabla L = 0, g_{T_i}(\hat{\theta}) \leq 0, \mu_i g_{T_i}(\hat{\theta}) = 0, \mu_i \leq 0 \quad (\text{for minima}) \quad (9)$$

We define gradient  $\nabla f_{T_i}(\hat{\theta})$  and Hessian  $\nabla^2 f_{T_i}(\hat{\theta})$  of the objective function  $f: \mathbb{R} \rightarrow \mathbb{R}^3$  as

$$\nabla f_{T_i}(\hat{\theta}) = \begin{bmatrix} \frac{\partial f}{\partial \xi} \\ \frac{\partial f}{\partial \omega_n} \\ \frac{\partial f}{\partial t_d} \end{bmatrix} \quad (10) \quad \nabla^2 f_{T_i}(\hat{\theta}) = \begin{bmatrix} \frac{\partial^2 f}{\partial \xi^2} & \frac{\partial^2 f}{\partial \xi \partial \omega_n} & \frac{\partial^2 f}{\partial \xi \partial t_d} \\ \frac{\partial^2 f}{\partial \omega_n \partial \xi} & \frac{\partial^2 f}{\partial \omega_n^2} & \frac{\partial^2 f}{\partial \omega_n \partial t_d} \\ \frac{\partial^2 f}{\partial t_d \partial \xi} & \frac{\partial^2 f}{\partial t_d \partial \omega_n} & \frac{\partial^2 f}{\partial t_d^2} \end{bmatrix} \quad (11)$$

Here Newton's Method is used to find the parameters of the model which minimizes the objective function  $f_{T_i}$ . Applying the second-order Taylor series to the objective function about the current point  $\hat{\theta}_k$  we get (12)

$$f_{T_i}(\hat{\theta}_{k-1} + s_k) = f_{T_i}(\hat{\theta}_{k-1}) + s_k (\nabla(f_{T_i}(\hat{\theta}_{k-1})))^T + \frac{1}{2} s_k^T \nabla^2(f_{T_i}(\hat{\theta}_{k-1})) s_k \quad (12)$$

where the step size  $s_k$  is given as in (13)

$$s_k = \hat{\theta}_k - \hat{\theta}_{k-1} \quad (13)$$

Differentiating (12) with respect to  $s_k$  and setting its L.H.S to zero we find the step size  $s_k$  which minimizes the quadratic model as (14)

$$s_k \nabla^2(f_{T_i}(\hat{\theta}_{k-1})) = \nabla(f_{T_i}(\hat{\theta}_{k-1})) \quad (14)$$

Hence

$$\hat{\theta}_k = \hat{\theta}_{k-1} - \nabla^2(f_{T_i}(\hat{\theta}_{k-1}))^{-1} \nabla(f_{T_i}(\hat{\theta}_{k-1})) \quad (15)$$

Strong convexity of  $(f_{T_i}(\hat{\theta}_{k-1}))$  is the sufficient condition of for the invertibility of the Hessain.

Strong convexity of  $f_{T_i}(\hat{\theta}_{k-1})$  implies  $\exists$  a constant  $c$  such that  $(f_{T_i}(\hat{\theta}_{k-1}) - c|\hat{\theta}_{k-1}|^2)$  is convex. This can be straightforwardly proved using (3) and Proposition 3 of Section 2. Hence, Hessian of  $f_{T_i}$  will always be invertible.

When KKT conditions are satisfied, critical points of the objective function and Lagrangian function become the same as given in (16).

$$\begin{bmatrix} \hat{\theta}_k \\ \lambda_k \\ \mu_k \end{bmatrix} = \begin{bmatrix} \hat{\theta}_{k-1} \\ \lambda_{k-1} \\ \mu_{k-1} \end{bmatrix} - (\nabla^2 L_{k-1})^{-1} \nabla L_{k-1} \quad (16)$$

$$\text{where } \nabla L_k(\hat{\theta}, \mu) = \begin{bmatrix} \frac{dL_k}{d\hat{\theta}} \\ \frac{dL_k}{d\mu_1} \\ \vdots \\ \frac{dL_k}{d\mu_m} \end{bmatrix} = \begin{bmatrix} \nabla f_{T_i}(\hat{\theta}) + \sum_I \mu_I \nabla g_{T_i}(\hat{\theta}) \\ g_{T_1}(\hat{\theta}) \\ \vdots \\ g_{T_m}(\hat{\theta}) \end{bmatrix} \quad (17)$$

$$\text{and } \nabla^2 L_k(\hat{\theta}, \mu) = \begin{bmatrix} \nabla^2 f_{T_i}(\hat{\theta}) + \sum_I \mu_I \nabla^2 g_{T_i}(\hat{\theta}) & \nabla g_{T_1}(\hat{\theta}) & \cdots & \nabla g_{T_m}(\hat{\theta}) \\ \nabla g_{T_1}(\hat{\theta}) & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \nabla g_{T_m}(\hat{\theta}) & 0 & 0 & 0 \end{bmatrix} \quad (18)$$

where  $i = 1, 2 \dots m$ .

Thus, updating the Hessian of the Lagrangian  $\nabla^2 L$  from (18) and the gradient of the Lagrangian  $\nabla L$  from (17) in every iteration, optimal values of the parameters are found from (16)

### 3.2.3 The gradient based optimization algorithm

The primary objective of studying this iterative optimization algorithm is to enhance parameter estimation accuracy using finite measurement input-output data. The study involves a SOPDT system (1) driven by a unipolar square wave signal with 1 Volt amplitude. Step input and response data of the plant are acquired in .CSV format from a DSO and analyzed in MATLAB to obtain initial parameter values.

:  $\hat{\theta}_0 = [\xi_0, \omega_{n_0} \text{ and } t_{d_0}]$ . The solution space for the parameters is as follows:  $0 < \xi < 1, \omega_n > 0, 0 < t_d < 1$ , the universe of discourse and feasible solution space as depicted in Figure 3.2.

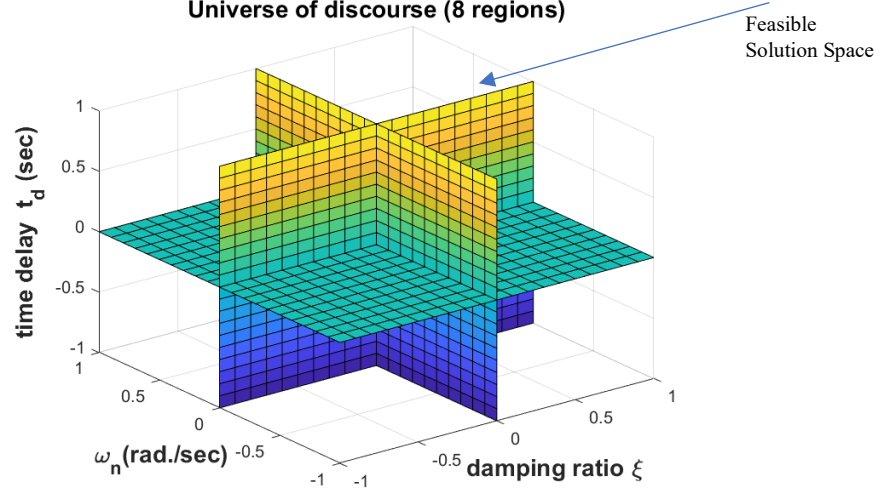


Figure 3.2 Universe of discourse of the parameters to be estimated

### 3.2.4 Proof of convexity of the objective function

If the objective function (3) is convex, then in the minimization problem (4), global minima are guaranteed [37]. In (3)  $y(t_i)$  is the output data collected in the DSO. Hence

$$\hat{y}(t_i, \hat{\theta}) = \hat{x}^T \hat{\theta} \quad (19)$$

where  $\hat{x}^T$  are the inputs and  $\hat{\theta}$  are the model parameters  $\hat{\theta} = [\xi, \omega_n, t_d]$ . Using Proposition 3, and (3) it is enough to verify here that  $g(\hat{\theta})$  is convex where

$$g(\hat{\theta}) = |y(t_i) - \hat{x}^T \hat{\theta}| \quad (20)$$

Using Proposition 1, we need to show

$$g(\lambda \hat{\theta}_1 + (1 - \lambda) \hat{\theta}_2) \leq \lambda g(\hat{\theta}_1) + (1 - \lambda) g(\hat{\theta}_2) \quad (21)$$

where  $0 < \lambda < 1$  for any  $\hat{\theta}_1, \hat{\theta}_2 \in \mathbb{R}^3$  of the feasible region shown in Figure 2

Using (20) and (21) we get

$$|y(t_i) - \hat{x}^T (\lambda \hat{\theta}_1 + (1 - \lambda) \hat{\theta}_2)| \leq \lambda |y(t_i) - \hat{x}^T \hat{\theta}_1| + (1 - \lambda) |y(t_i) - \hat{x}^T \hat{\theta}_2| \quad (22)$$

As  $0 < \lambda < 1$ , so  $\lambda$  and  $(1 - \lambda)$  are positive and they can be put inside the absolute values in (22) such that

$$|y(t_i) - \hat{x}^T (\lambda \hat{\theta}_1 + (1 - \lambda) \hat{\theta}_2)| \leq |\lambda y(t_i) - \lambda \hat{x}^T \hat{\theta}_1| + |(1 - \lambda) y(t_i) - (1 - \lambda) \hat{x}^T \hat{\theta}_2| \quad (23)$$

$$\text{Let } a = \lambda y(t_i) - \lambda \hat{x}^T \hat{\theta}_1 \quad \text{and} \quad b = (1 - \lambda) y(t_i) - (1 - \lambda) \hat{x}^T \hat{\theta}_2 \quad (24)$$

$$\text{Thus } a + b = y(t_i) - \hat{x}^T (\lambda \hat{\theta}_1 + (1 - \lambda) \hat{\theta}_2) \quad (25)$$

So, (23) can be written as  $|a + b| \leq |a| + |b|$  which is always true.

Hence the objective function is convex.

Without any loss of generality, we may assume the objective function to be differentiable everywhere except when  $y(t_i) = \hat{y}(t_i, \hat{\theta})$  which is almost impossible.

For a convex and differentiable objective function, Proposition 2 ensures that its local minima are also global minima.

### 3.2.5 The Proposed Algorithm

The proposed algorithm is a versatile optimization technique that can be adapted to a wide range of systems and data types. Its flexibility lies in its ability to handle different cost functions and optimization objectives, making it a fundamental tool in the realm of machine learning and optimization. Step by step breakdown of the algorithm is given below:

**Step 1:** The .CSV file, comprised of 300 data points for both input and output signal, is taken at a sampling frequency of 200KHz for a real time underdamped SOPDT system (LP Butterworth filter) ( which has  $\xi = 0.5, \omega_n = 5000 \text{ rad/sec}, t_d = 0.6 \times 10^{-3}$ ).

**Step 2:** Read the .CSV file in MATLAB. Collect the input-output observations  $(u(t_i), y(t_i))$  where  $i = 1, 2, \dots, 300$  and plot the time response (output) of the system together with the input signal as shown in Figure 3.3.

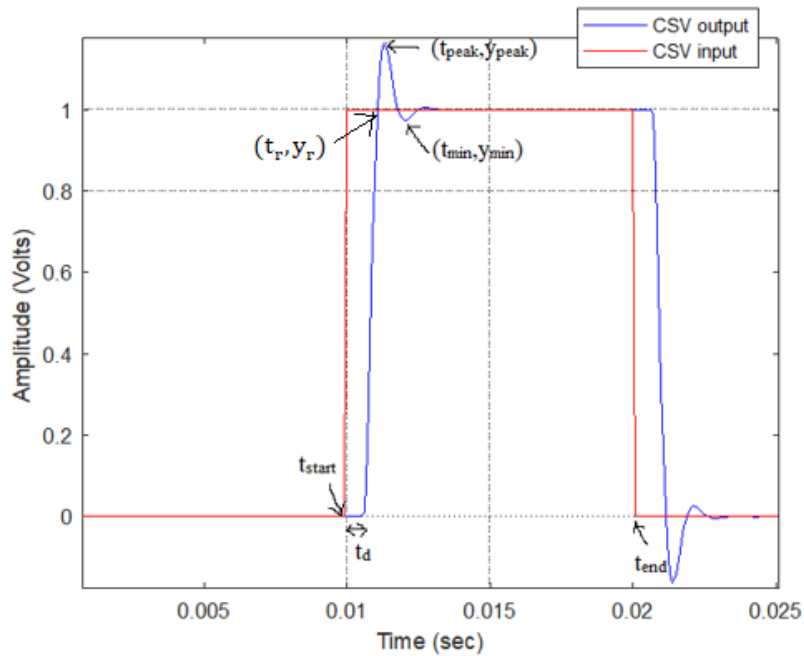


Figure 3.3. Raw Input-Output of the plant under test

**Step 3:** Identify the start time  $t_{start}$  that is the time when the input signal first time, reaches unity from zero, and the end time  $t_{end}$  when the input signal first time, reaches zero from unity.

**Step 4:** Identify the index corresponding to the peak amplitude of the output and hence find  $y_{peak}$  and the corresponding time  $t_{peak}$ . Also, identify the time  $t_r$  when the output reaches steady state  $y_r$  for the first time.

**Step 5:** Identify the index corresponding to the minimum amplitude of the output and hence find  $y_{min}$  and the corresponding time  $t_{min}$  between the start time  $t_{start}$  and end time  $t_{end}$ .

**Step 6:** Evaluate the damped frequency  $\omega_d = \frac{\frac{\pi}{2}}{t_{peak}-t_r}$  where  $t_r$  is the time when the output reaches the steady state value for the first time. (as  $\omega_d = \frac{2\pi}{4(t_{peak}-t_r)}$  where the time period between consecutive crest is  $4(t_{peak} - t_r)$  from Figure 3).

**Step 7:** Evaluate constant  $L = \frac{1}{\pi} \ln \left( \frac{1}{y_{peak}-1} \right)^2$  and damping coefficient  $\xi = \sqrt{\frac{L}{1+L}}$ .

It is known damping ratio  $\xi = \frac{(\ln M)^2}{\pi^2 + (\ln M)^2}$  where the peak overshoot  $M = y_{peak} - 1 = e^{\frac{-\xi\pi}{\sqrt{1-\xi^2}}}$ .

Rearranging we get a constant  $L = \frac{(\xi)^2}{1-(\xi)^2} = \frac{1}{\pi} \ln \left( \frac{1}{y_{peak}-1} \right)^2$  and thus  $\xi = \sqrt{\frac{L}{1+L}}$ .

**Step 8:** Evaluate natural frequency  $\omega_n = \frac{\omega_d}{\sqrt{1-\xi^2}}$  and the peak time  $t_p$  which corresponds to a system without time delay  $t_d$  where  $t_p = \frac{\pi}{\omega_d}$ .

**Step 9:** Evaluate the time delay  $t_d = (t_{peak} - t_p) - t_{start}$ .

**Step 10:** Get the starting values of the parameters  $\hat{\theta}_0$  from Steps 7-9. We get  $[\xi_0, \omega_{n0} \text{ and } t_{d0}] = [0.5000, 5003.58, 5.99 \times 10^{-4}]$ .

**Note:** The starting values of the parameters are determined using established analytical approach as indicated in Steps 3-10.

**Step 11:** Create lower-bound (lb) and upper-bound (ub) of the obtained parameters  $\xi$ ,  $\omega_n$ , and  $t_d$  as  $lb = \hat{\theta}_0(m) - 1.5\% * \hat{\theta}_0(m)$  and  $ub = \hat{\theta}_0(m) + 1.5\% * \hat{\theta}_0(m)$

where  $m=1,2,3$  such that  $\hat{\theta}_0(1) = \xi$ ,  $\hat{\theta}_0(2) = \omega_n$  and  $\hat{\theta}_0(3) = t_d$ .

Here  $lb = [0.492, 4925.0, 5.91 \times 10^{-4}]$  and  $ub = [0.507, 5078.64, 6.09 \times 10^{-4}]$

**Step 12:** Start with the initial values of the parameters as  $\hat{\theta}_{k-1} = \hat{\theta}_0$ .



**Step13:** With this  $\hat{\theta}_{k-1}$  create the SOPDT model and perform Steps 3-9 to evaluate new values of  $\xi$ ,  $\omega_n$  and  $t_d$  and create a new SOPDT model with these values, and find the estimated step response  $\hat{y}_{k-1}(t)$ .

**Step 14:** Compute the objective function  $f_{T_i}(\hat{\theta}_{k-1})$  from Equation (3) and hence the Lagrangian function from Equation (8).

**Step15:** Compute the gradient of the Lagrangian  $\nabla L_k(\hat{\theta}, \mu)$  from Equation (17) and Hessian of the Lagrangian  $\nabla^2 L_k(\hat{\theta}, \mu)$  from Equation (18), such that at each iteration, the gradient (or derivative) of the cost function with respect to the model parameters is computed.

**Step 16:** Update the value of  $\hat{\theta}$  and  $\mu$  and from Equation (16)

**Note:** The parameters are updated by subtracting a small fraction  $((\nabla^2 L_{k-1}(\hat{\theta}, \mu))^{-1})$  of the gradient from the current parameter values as given in (16). This fraction is the learning rate and is a crucial hyperparameter that needs to be carefully chosen. A small learning rate may lead to slow convergence, while a large one can lead to overshooting and divergence.

**Step 17:** Repeat Steps (13-16) for  $\hat{\theta}$  such that  $lb \leq \hat{\theta} \leq ub$ , but skip the step when upper-bound of  $\xi \geq 1$ .

**Note:** The stopping/convergence criterion for an optimization algorithm is typically based on the objective of finding the minimum of a cost function within a specified range, which is defined by lower and upper bounds.

**Step18:** Identify the optimized parameter values  $\xi$ ,  $\omega_n$  and  $t_d$  which correspond to the minimum value of  $f_{T_i}$  between steps 13 to 17.

**Step19:** Find the system gain  $K$  as the ratio of the amplitude of the output response at steady state to the amplitude of input. Clearly from Figure 3,  $K=1$ .

**Step 20:** Create the SOPDT model with the optimized parameters and compare its unit step response with the  $y(t_i)$  of Step 2.

The estimated optimized parameters and the corresponding IAE are tabulated in Table 3.1.

**TABLE 3.1: Estimated model parameters and IAE (of real time SOPDT system (LP Butterworth filter with input delay block))**

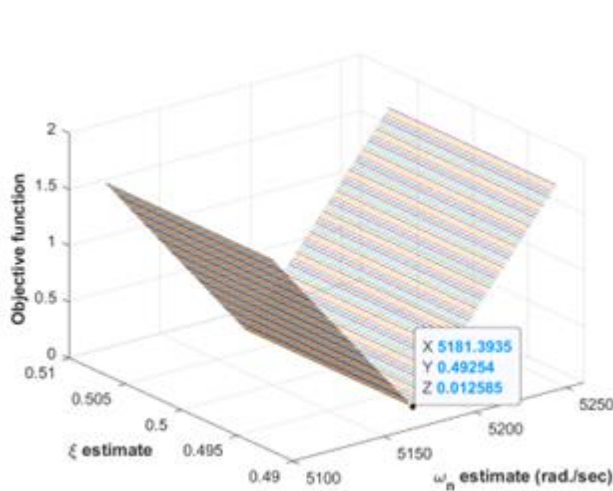
Assumed sampling period =  $10^{-3}$  sec

	$\omega_n \left( \frac{\text{rad}}{\text{sec}} \right)$	$\xi$	$t_d \text{ (sec)}$	IAE
Model 1 (original system)	$5.0 \times 10^3$	0.5	$0.6 \times 10^{-3}$	0.0011
Model 2 (estimated model)	$5.0006 \times 10^3$	0.500	$0.60 \times 10^{-3}$	

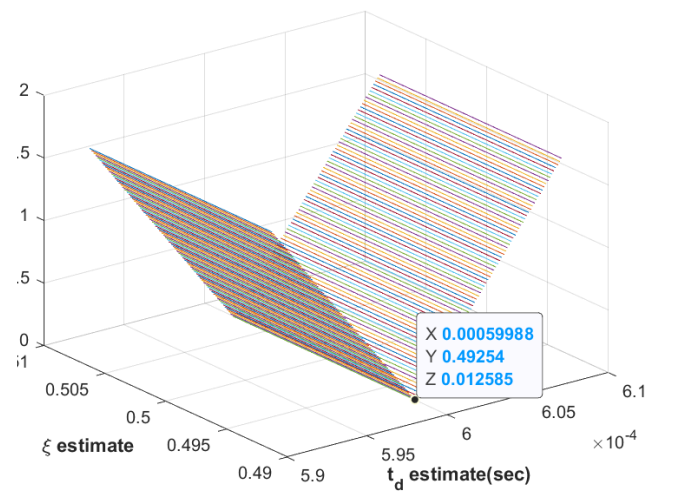
### 3.2.6 Results and analysis of the algorithm

#### 3.2.6.1 Sensitivity of the model and global convergence

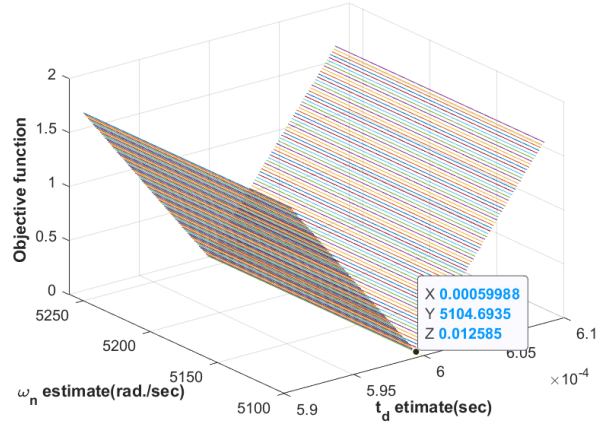
The proposed algorithm requires an initial value to start the iteration process, and an arbitrary choice may lead to converging to a local minimum. To assess model sensitivity to computational parameters, sensitivity analysis is performed which varies some parameters while keeping others fixed. Numerical simulations in MATLAB (Figure 3.4a, 3.4b, 3.4c) under step tests show that changing any two parameters within a specific range results in the same minimum objective cost function, affirming insensitivity to computational parameter changes. The convexity of the objective function in all three model parameters (verified by Figure 4a, 4b, 4c) ensures global minima, guaranteeing global convergence.



3.4a.



3.4b.



3.4c.

Figure 3.4 Plot of objective function under step input with **a)** constant  $t_d$  **b)** with constant  $\omega_n$  **c)** with constant  $\xi$

### 3.2.6.2 Comparison of the performance index of the proposed algorithm with two existing non-iterative methods

The proposed method's efficacy is shown by comparing it with methods [109] and [110] using IAE (Integral of Absolute Error) as the fitting criterion. Table 3.2 displays the comparative study of estimated parameters and IAE values. The step response for the SOPDT system under test ( $\omega_n=5000$  rad/sec,  $\xi=0.5$ ,  $t_d=0.0006$  sec) is shown in Figure 3.5 for the three methods.

**TABLE 3.2** Comparative study

Original System	Proposed Method	RangiahMethod [109]	Tsung Method [110]
$\omega_n=5000$ rad/sec. $\xi=0.5$	IAE=1.689 $\omega_n=5000.06$ rad/sec. $\xi=0.5000$ $t_d=0.0006$ sec	<b>3-point method</b> IAE =12.32 $\omega_n=6334$ rad/sec. $\xi=0.623$	<b>5-point method</b> IAE =0.4392 $\omega_n=4977$ rad/sec. $\xi=0.500$

$t_d=0.0006$ sec **	(Initial values IAE 0.0011 $\omega_n=50003.58$ rad/sec. $\xi=0.500$ $t_d=0.0006$ sec)	$t_d=0.00097$ sec  2-point method IAE =10.86 $\omega_n=5661$ rad/sec. $\xi=0.623$ $t_d=0.00076$ sec	$t_d=0.0005927$ sec  3-point method IAE =0.4624 $\omega_n=4976$ rad/sec. $\xi=0.500$ $t_d=0.0005953$ sec
$\omega_n=1$ rad/sec. $\xi=0.6$ $t_d=0.007$ sec	IAE= 0.0501 $\omega_n=0.999$ rad/sec. $\xi=0.599$ $t_d=0.0069$ sec (Initial values IAE=0.144 $\omega_n=1.0005$ rad/sec. $\xi=0.600$ $t_d=0.0068$ sec)	3-point method IAE=6.65 $\omega_n=1.18$ rad/sec. $\xi=0.696$ $t_d=0.0056$ sec  2-point method IAE=6.31 $\omega_n=1.145$ rad/sec. $\xi=0.696$ $t_d=0.0058$ sec	5-point method IAE=0.229 $\omega_n=0.996$ rad/sec. $\xi=0.600$ $t_d=0.00064$ sec  3-point method IAE=0.234 $\omega_n=0.967$ rad/sec. $\xi=0.600$ $t_d=0.00035$ sec

\*\* a real-time SOPDT system (Second order LP Butterworth filter with input delay block)

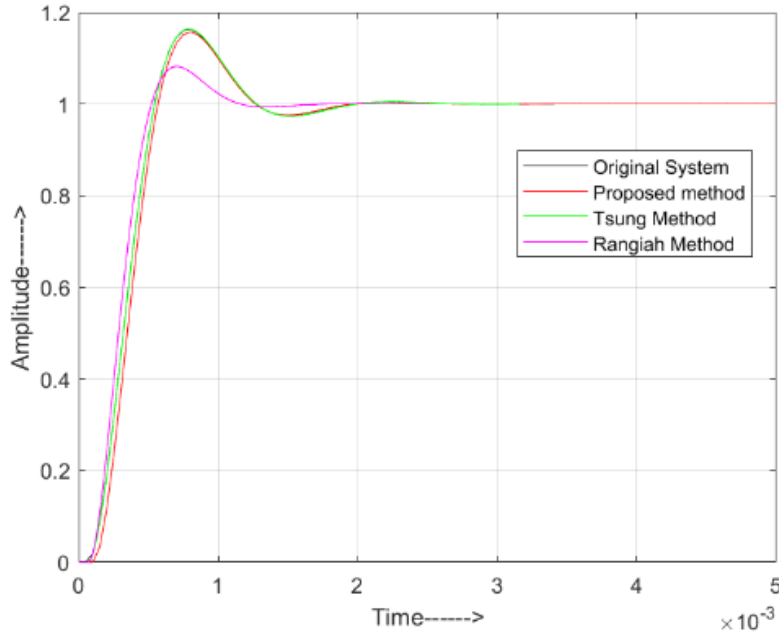


Figure 3.5 Comparison of step response by different methods

$$t_d = 0.0006 \text{ sec}, \xi = 0.5, \omega_n = 5000 \frac{\text{rad}}{\text{sec}}$$

### 3.2.6.3 Parameter estimation of the SOPDT system in the presence of measurement noise and comparison with time domain methods [109] [110] and frequency domain method [24].

The algorithm might get affected by measurement noise, which can cause variations in the gradient direction, slowing down convergence. To counter this L1 (Lasso) regularization technique is employed. The L1 regularization enhances the model's resilience to noise by incorporating penalty terms in the cost function based on the sum of the absolute values of the model's parameters. These penalties push the model to have smaller parameter values, helping it avoid overfitting to noisy data. The study assesses estimation algorithm robustness in the presence of measurement noise using White Gaussian noise (mean=0, and 1% standard deviation). The proposed method performs better in estimating process parameters with noise, outperforming three other methods [109] [110] [24] as shown in (Table 3.3). Figures 3.6a and 3.6b demonstrate an IAE comparison for a specific system configuration of time domain methods and frequency domain methods respectively.

**TABLE 3.3 Comparative study of SOPDT system (with measurement noise) under Proposed method with Rangiah [110], Tsung [109] and Cheres[24] method**

Original System	Proposed Method	Rangiah Method [110] (time domain method)	Tsung Method [109] (time domain method)	Cheres Method [24] (frequency domain method)
$\omega_n=5000$ rad/sec. $\xi=0.5$ $t_d=0.003$ sec	IAE=8.92 $\omega_n=4931$ rad/sec. $\xi=0.487$ $t_d=0.00299$ sec (Initial values IAE=16.55 $\omega_n=4752$ rad/sec. $\xi=0.483$ $t_d=0.00295$ sec)	<b>3-point method</b> IAE =17.69 $\omega_n=6185$ rad/sec. $\xi=0.583$ $t_d=0.0030$ sec <b>2-point method</b> IAE =16.19 $\omega_n=5465$ rad/sec. $\xi=0.5$ $t_d=0.0030$ sec	<b>5-point method</b> IAE =9.58 $\omega_n=4839$ rad/sec. $\xi=0.471$ $t_d=0.0029$ sec <b>3-point method</b> IAE=9.57 $\omega_n=4864$ rad/sec. $\xi=0.471$ $t_d=0.0029$ sec	IAE=8.98 $\omega_n=6023$ rad/sec. $\xi=0.587$ $t_d=0.00299$ sec
$\omega_n=500$ rad/sec. $\xi=0.6$ $t_d=0.004$ sec	IAE=8.68 $\omega_n=497.5$ rad/sec. $\xi=0.583$ $t_d=0.00395$ sec  (Initial values IAE=11.62 $\omega_n=505.1$ rad/sec. $\xi=0.574$ $t_d=0.00390$ sec)	<b>3-point method</b> IAE =12.32 $\omega_n=572.96$ rad/sec. $\xi=0.65$ $t_d=0.00416$ sec <b>2-point method</b> IAE =12.22 $\omega_n=568.55$ /sec. $\xi=0.65$ $t_d=0.00414$ sec	<b>5-point method</b> IAE =8.728 $\omega_n=478.77$ rad/sec. $\xi=0.574$ $t_d=0.0039$ sec <b>3-point method</b> IAE =8.69 $\omega_n=480.01$ rad/sec. $\xi=0.574$ $t_d=0.00391$ sec	IAE=10.58 $\omega_n=597.$ rad/sec. $\xi=0.625$ $t_d=0.00495$ sec

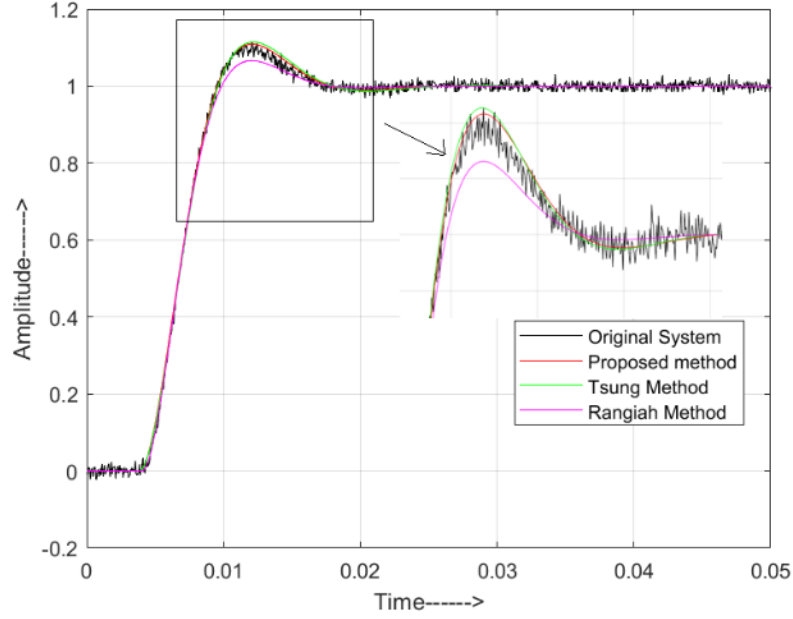


Figure 3.6a Model responses for the system with  $t_d = 0.004 \text{ sec}$ ,  $\xi = 0.6$ ,  $\omega_n = 500 \frac{\text{rad}}{\text{sec}}$  under measurement noise (Proposed method, Time domain methods of Rangiah method and Tsung)

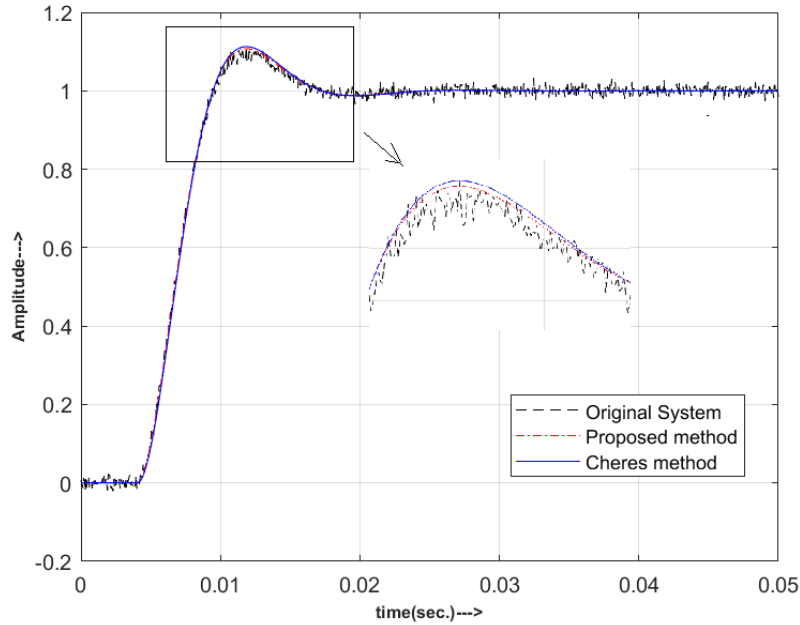


Figure 3.6b Model responses for the system with  $t_d = 0.004 \text{ sec}$ ,  $\xi = 0.6$ ,  $\omega_n = 500 \frac{\text{rad}}{\text{sec}}$  under measurement noise (Proposed method and Frequency domain method of Cheres)

#### 3.2.6.4 Comparison of the performance index of the proposed algorithm with the existing iterative method [35]

Table 3.4 compares the proposed method with Cox's least square-based iterative optimization method [35]. The Cox method had limitations with unsatisfactory outcomes when using fixed initial values of  $[0.5, 0.5, 0.5]$ . To improve performance, the Cox method is made to adopt the



same initial value selection as the proposed method. Table 3.4a and Table 3.4b illustrates the IAE and convergence comparison respectively between the proposed method and Cox's method, indicating the superior and faster convergence of the proposed approach.

**TABLE 3.4a Comparative study of the Proposed method with Cox Method [35]**

Original System	Proposed Method	Cox Method [35]
$\omega_n=10$ rad/sec. $\xi=0.3$ $t_d=0.007$ sec	IAE=1.260 $\omega_n=10.044$ rad/sec. $\xi=0.299$ $t_d=0.0073$ sec	IAE=1.529 $\omega_n=10.072$ rad/sec. $\xi=0.298$ $t_d=0.0073$ sec
$\omega_n=250$ rad/sec. $\xi=0.7$ $t_d=0.0006$ sec	IAE=0.00012 $\omega_n=249.99$ rad/sec. $\xi=0.699$ $t_d=0.000599$ sec	IAE=0.0031 $\omega_n=249.97$ rad/sec. $\xi=0.697$ $t_d=0.000594$ sec

**TABLE 3.4b Convergence Comparison of the Proposed method with Cox Method [35]**  
for the system with  $\omega_n=10$  rad/sec,  $\xi=0.3$ ,  $t_d=0.007$  sec.

Iterative step k	$\omega_n$ estimated (rad/sec)		$\xi$ estimated		$t_d \times 10^{-3}$ sec Estimated		IAE	
	Proposed	Cox	Proposed	Cox	Proposed	Cox	Proposed	Cox
0 (start value)	10.256	10.256	0.302	0.302	7.04	7.04	1.625	1.625
5	10.232	10.254	0.301	0.303	7.04	7.05	1.588	1.611
10	10.102	10.202	0.299	0.301	7.04	7.04	1.489	1.599
15	10.081	10.181	0.299	0.299	7.04	7.04	1.388	1.556
20	10.044	10.072	0.299	0.298	7.03	7.03	1.260	1.529

### 3.2.6.5 Cross-validation of the proposed method on other higher-order underdamped systems

The proposed methodology is applied to a third order dead time system with complex conjugate dominant pole pair (poles are  $-2 \times 10^4$  and  $-10^4 \pm j0.04$ ), and to a fourth order system, with two complex conjugate pole pairs: one dominant and the other away from origin (poles are  $(-2.5 \pm j4.33)$  and  $(-0.4 \pm j1.95)$ ) as shown in Table 6 to find out whether the proposed method can model such a system by second order dynamics. The simulation results is shown in Figure 8a and 8b. Estimated parameters and measured IAE given in Table 3.5 clearly indicate that the proposed method gives much reliable parameter estimates compared to Rangiah [110] and Tsung [109] method.

**TABLE 3.5 Comparative study for higher order systems) Proposed method with Rangiah [110] and Tsung [109] (system gain K=1, for original system and estimated methods)**

Original System		Rangiah Method [110]	Tsung Method [109]
<p><b>A third order system, where dominant pole is complex conjugate pole pair</b></p> $G(s) = \frac{e^{-t_d s}}{(Ts + 1)} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right)$ <p><math>\omega_n = 500 \text{ rad/sec.}</math>  <math>\xi = 0.6</math>  <math>t_d = 0.004 \text{ sec}</math>  <math>T = 0.00005</math></p>	<p>IAE=0.0077  <math>\omega_n = 499.95</math>  rad/sec.  <math>\xi = 0.600</math>  <math>t_d = 0.00404 \text{ sec}</math>  (Initial values  IAE=0.056  <math>\omega_n = 500.2</math>  rad/sec.  <math>\xi = 0.600</math>  <math>t_d = 0.00405 \text{ sec}</math>  )</p>	<p><b>Three pt</b>  IAE = 6.65  <math>\omega_n = 589.9 \text{ rad/sec.}</math>  <math>\xi = 0.69</math>  <math>t_d = 0.0043 \text{ sec}</math>  <b>Two pt</b>  IAE = 6.31  <math>\omega_n = 572.4 \text{ rad/sec.}</math>  <math>\xi = 0.69</math>  <math>t_d = 0.0042 \text{ sec}</math></p>	<p><b>Five pt method</b>  IAE = 0.227  <math>\omega_n = 498.25 \text{ rad/sec.}</math>  <math>\xi = 0.600</math>  <math>t_d = 0.0040 \text{ sec}</math>    <b>Three pt method</b>  IAE = 0.234  <math>\omega_n = 498.209 \text{ rad/sec}</math>  <math>\xi = 0.600</math>  <math>t_d = 0.0040 \text{ sec}</math></p>

<p><b>A fourth order system, with two complex conjugate pole pairs: one dominant and the other away from origin</b></p> $G(s) = Ke^{-t_d s} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right) \left( \frac{\omega_{n2}^2}{s^2 + 2\xi_2\omega_{n2} s + \omega_{n2}^2} \right)$ <p><math>\omega_n=2\text{rad/sec.}</math>  <math>\xi=0.2</math>  <math>t_d=0.02 \text{ sec}</math>  <math>\omega_{n2}=5\text{rad/sec.}</math>  <math>\xi_2=0.5</math></p>	<p>IAE=14.31  <math>\omega_n=1.99</math>  rad/sec.  <math>\xi=0.21</math>  <math>t_d=0.021 \text{ sec}</math>  (Initial values  IAE=16.34  <math>\omega_n=1.997</math>  rad/sec.  <math>\xi=0.22</math>  <math>t_d=0.023 \text{ sec})</math></p>	<p><b>Three pt</b>  IAE =68.97  <math>\omega_n=2.15 \text{ rad/sec.}</math>  <math>\xi=0.3100</math>  <math>t_d=0.04 \text{ sec}</math></p> <p><b>Two pt</b>  IAE =95.95  <math>\omega_n=3.65\text{rad/sec}</math>  <math>\xi=0.3600</math>  <math>t_d=0.03 \text{ sec}</math></p>	<p><b>Five pt method</b>  IAE =81.71  <math>\omega_n=2.30 \text{ rad/sec.}</math>  <math>\xi=0.17</math>  <math>t_d=0.030 \text{ sec}</math></p> <p><b>Three pt method</b>  IAE= 82.26  <math>\omega_n=2.29 \text{ rad/sec.}</math>  <math>\xi=0.17</math>  <math>t_d=0.028 \text{ sec}</math></p>
---	--	--	--

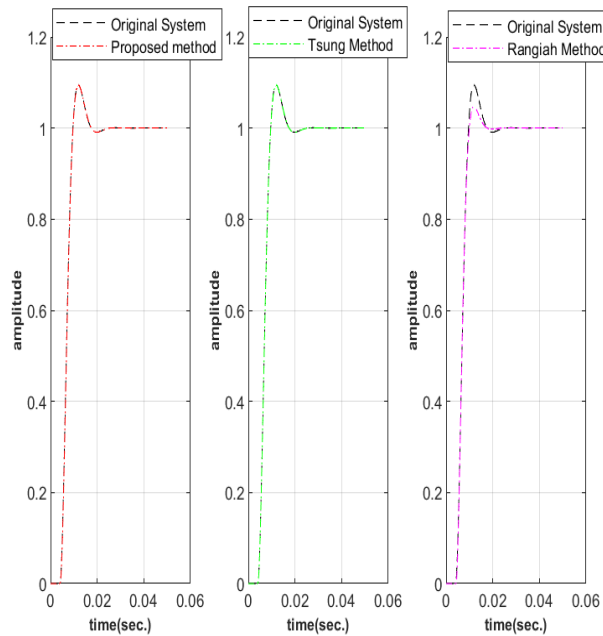


Fig. 3.7(a)

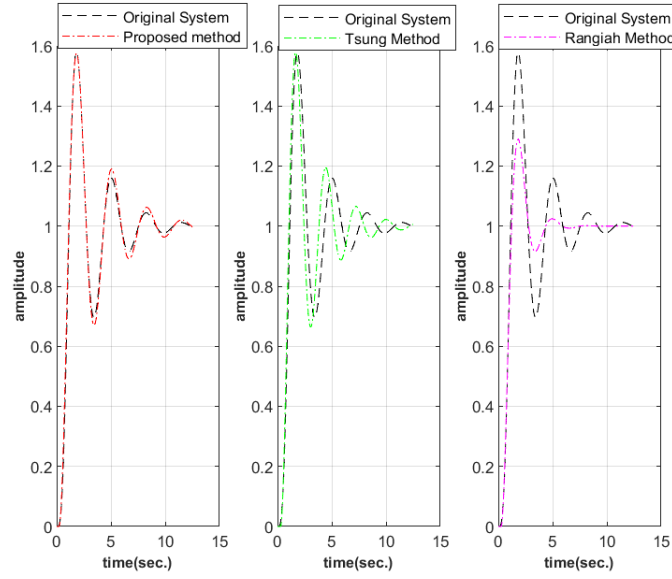


Fig. 3.7(b)

Figure 3.7 Model responses for

a) third order system with a complex conjugate dominant pole pair (system has  $t_d = 0.004 \text{ sec}$ ,  $\xi = 0.6$ ,  $\omega_n = 500 \text{ rad/sec}$ )

b) fourth order system with two complex conjugat pole pairs, one dominant and other away from the origin (system has  $t_d = 0.02 \text{ sec}$ ,  $\xi = 0.2$ ,  $\omega_n = 2 \text{ rad/sec}$ )

The simulation results suggest that the proposed methodology successfully models complex third and fourth order systems with second-order dynamics, and the estimated parameters are notably more reliable compared to alternative methods, [109] and [110]. The simulation results and comparison tables provide evidence supporting the effectiveness of the proposed method in capturing the dynamics of these challenging system configurations.

### 3.2.6.6. Fast Convergence

The proposed algorithm when implemented on *the real-time underdamped SOPDT system (Second order LP Butterworth filter with an input delay block)* (which has  $\xi = 0.5$ ,  $\omega_n = 5000 \text{ rad/sec}$ ,  $t_d = 0.6 \times 10^{-3}$ ) ensures fast convergence, as evidenced by Table 3.6. The computational load is notably reduced because the number of sampled data for identification (N) is much larger than the number of iterative steps (k).

**TABLE 3.6 Identification results for the system under test ( *real-time underdamped SOPDT system (LP Butterworth filter)* ) (N=300 sample points) by the proposed method**

Iterative step k	$\omega_n$ estimated (rad/sec)	$\xi$ estimated	$t_d \times 10^{-4} \text{sec}$ estimated	IAE
0 (start value) *	5003.58	0.501	6.00	1.689
5	5003.10	0.501	6.00	1.388
9	5002.91	0.500	6.00	1.289
15	5001.03	0.499	6.00	0.235
19	5000.08	0.500	6.00	0.0337
25	5000.06	0.500	6.00	0.0011

\*Initial estimates are evaluated in Step10 of the algorithm (see section 4.2)

### 3.2.6.7 Acquisition of transient time response data

The algorithm's initial parameter viz.  $\xi$ ,  $\omega_n$ , and  $t_d$ , estimate is based on data points up to the first peak of the step response. For electrical systems using DSO or DAC, the auto-set sampling rate may not precisely determine the first peak and crest of the transient response.

A higher sampling frequency (200 KHz) from a 70 MHz, 2 GS/s, 2-channel DSO (GDS-2072A) from M/S GWINSTEK Inc is used for data acquisition in this study.

### 3.2.7 Conclusion

A gradient-based iterative algorithm is developed for parameter estimation of SOPDT systems from step response. This work ensures an improvement in the parameter estimation of dynamic systems with time delays as system-specific constraints are incorporated in the estimation process; it may thus be extended to the estimation of similar processes in other application areas. By incorporating constraints and leveraging convex optimization, this work enhances the accuracy and reliability of the parameter estimation process, thereby contributing to more effective and robust process of control and optimization. Illustrative examples clearly show that the proposed method gives much more reliable and accurate results compared to traditional time domain methods like [109] [110], frequency domain method like [24] and established iterative method like [35]. The proposed method is a versatile, efficient, and has robust approach for handling diverse objective functions, including non-linear and noisy ones, while ensuring convergence from local to global optima, making it suitable for large datasets. The proposed algorithm may find wide practical applicability in fields like machine learning,

neural networks, natural language processing, and image processing. However, challenges arise in its implementation, such as selecting appropriate learning rates, the risk of converging to local minima, high computational complexity for large datasets, and the risks of overfitting or underfitting. Our future research endeavours to discover methods to mitigate these challenges and enhance its practical utility.

### 3.3 Modelling of SOPDT overdamped/critically damped systems

Designing a controller for a process plant typically modelled as a first or second-order system with dead time involves an efficient and accurate estimation of its parameters. Since many process plants are characterized as second-order plus dead time (SOPDT) overdamped or critically damped systems, this study presents a straightforward parameter estimation method using transient response data from a step input at three specific time instants. The proposed least-square-based iterative parameter estimation algorithm estimates the parameters of the canonical transfer function of the plant from the time response data at the three specific time instants. The method is effective and robust even with measurement noise, making it valuable for control, not only in process plants but also in other systems having similar time responses.

#### 3.3.1 Problem Formulation

Chemical plants are usually high-order overdamped systems and are characterized by large time constants and time delays. However, the dynamics of these high-order systems may be approximated by overdamped SOPDT systems (whose steady-state gain is normalized to unity), of the following form:

$$G(s) = \frac{e^{-t_d s}}{(1+T_1 s)(1+T_2 s)} \quad (26)$$

If the open-loop transfer function exhibit oscillatory behaviour (1a) takes the following form:

$$G(s) = \frac{e^{-t_d s}}{T^2 s^2 + 2\xi T s + 1} \quad (27)$$

Mathematically the aforesaid two forms are equivalent, but (27) is used for an underdamped case to avoid complex values of  $T$  and (26) is used for an overdamped case whose unit step response without dead time is given as:

$$y(t) = 1 + \frac{1}{T_2 - T_1} \left( T_1 e^{-\frac{t}{T_1}} - T_2 e^{-\frac{t}{T_2}} \right) \quad (28)$$

During critical damping  $T_1 \rightarrow T_2$ , and thus (28) may lead to computational difficulty. Also, for mild oscillations in the underdamped case when  $0.85 \leq \xi \leq 1$ , the use of (27) is preferred. Moreover, (26) is highly sensitive to variations in time constants and may lead to difficult convergence when using least-square estimation. Thus, we avoid its usage in the algorithm and we choose (27) as the governing equation of our SOPDT model.

We rewrite (27) in the following form:

$$G(s) = e^{-t_d s} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right) \quad (29)$$

(where  $t_d$  is the dead time,  $\omega_n$  is the undamped natural frequency and  $\xi$  is the damping factor of the system).

In this paper, we propose an algorithm for parameter estimation of a generic second-order plant with /without dead time with  $\xi \geq 1$  from its transient response to a step input. The transfer function of a plant whose parameters are to be estimated is assumed to take the canonical form of (29).

### 3.3.2 Estimation Procedure

The proposed estimation algorithm provides the values  $t_d$ ,  $\omega_n$  and  $\xi$  in a sequential manner by using three points of the transient response. For overdamped and critically damped systems with dead time, a three-point measurement of the transient response, viz.,  $t_5$ ,  $t_{10}$  and  $t_{90}$  is acquired for estimation of  $\omega_n$ ,  $\xi$  and where  $t_{90}$  is the time when step response attains 90% of its final value,  $t_{10}$  is the time when step response attains 10% of its final value and  $t_5$  is the time when step response attains 5% of its final value. For overdamped and critically damped systems, the measurable transient response performance index (PI) is the rise time  $t_r$ , which by definition is the time taken for the response to rise from 10% ( $t_{10}$ ) to 90% ( $t_{90}$ ) of the steady-state value. However, with  $\omega_n$  unknown, it is impossible to correlate  $\xi$  with  $t_r$ ; the correlation parameters change with a change in  $\omega_n$ . After a thorough investigation, it is found that for a given  $\xi$ , the ratio  $T_R = \frac{t_{90}-t_5}{t_{10}-t_5}$ , a PI defined for this purpose, remains the same for different values of  $\omega_n$ . The inclusion of  $t_5$  in calculating  $T_R$  ensures the elimination of the dead time. Moreover, for a fixed value of  $\omega_n$ ,  $T_R$  increases monotonically as  $\xi$  is increased as illustrated in Table 3.7. This points to the fact that a correlation between  $\xi$  and  $T_R$  may be formulated that is independent of  $\omega_n$ .



Further, the investigations revealed that for a fixed value of  $\xi$ , a linear relationship exists between  $\omega_n$  and  $f_r = \frac{1}{t_{90}-t_{10}}$ , where  $f_r$  is the reciprocal of the rise time  $t_r$ . Thus, for a given value of  $\xi$ , a correlation may also be formulated between  $\omega_n$  and  $f_r$ .

As is obvious (refer to Table 3.8), for identical systems with zero dead time, the correlations between  $\xi$  and  $T_R$ , and  $\omega_n$  and  $f_r$  are the same.

**Table 3.7 Variation of  $\xi$  with  $T_R$  &  $f_r$  for three different values of  $\omega_n$  with  $t_d = 0.2\text{sec}$**

$\omega_n = 1 \frac{\text{rad}}{\text{sec}}$			$\omega_n = 5 \frac{\text{rad}}{\text{sec}}$			$\omega_n = 10 \frac{\text{rad}}{\text{sec}}$		
$\xi$	$T_R$ $= \frac{t_{90}-t_5}{t_{10}-t_5}$	$f_r$ $= \frac{1}{t_{90} - t_{10}}$ $\left(\frac{1}{\text{sec}}\right)$	$\xi$	$T_R$ $= \frac{t_{90}-t_5}{t_{10}-t_5}$	$f_r$ $= \frac{1}{t_{90} - t_{10}}$ $\left(\frac{1}{\text{sec}}\right)$	$\xi$	$T_R$ $= \frac{t_{90}-t_5}{t_{10}-t_5}$	$f_r$ $= \frac{1}{t_{90} - t_{10}}$ $\left(\frac{1}{\text{sec}}\right)$
1.0	20.026	0.2978	1.0	20.026	1.489	1.0	20.026	2.978
1.2	24.335	0.2287	1.2	24.335	1.144	1.2	24.335	2.287
1.4	27.978	0.1862	1.4	27.978	0.931	1.4	27.978	1.862
1.6	30.951	0.1577	1.6	30.951	0.789	1.6	30.951	1.577
1.8	33.372	0.1371	1.8	33.372	0.686	1.8	33.372	1.371
2.0	35.317	0.1215	2.0	35.317	0.607	2.0	35.317	1.215

**Table 3.8 Variation of  $\xi$  with  $T_R$  &  $f_r$  for three different values of  $\omega_n$  with  $t_d = 0$  sec**

$\omega_n = \frac{1rad}{sec}$			$\omega_n = \frac{5rad}{sec}$			$\omega_n = \frac{10rad}{sec}$		
$\xi$	$T_R$ $= \frac{t_{90}-t_5}{t_{10}-t_5}$	$f_r$ $= \frac{1}{t_{90}-t_{10}}$ $\left(\frac{1}{sec}\right)$	$\xi$	$T_R$ $= \frac{t_{90}-t_5}{t_{10}-t_5}$	$f_r$ $= \frac{1}{t_{90}-t_{10}}$ $\left(\frac{1}{sec}\right)$	$\xi$	$T_R$ $= \frac{t_{90}-t_5}{t_{10}-t_5}$	$f_r$ $= \frac{1}{t_{90}-t_{10}}$ $\left(\frac{1}{sec}\right)$
1.0	20.026	0.2978	1.0	20.026	1.489	1.0	20.026	2.978
1.2	24.335	0.2287	1.2	24.335	1.144	1.2	24.335	2.287
1.4	27.978	0.1862	1.4	27.978	0.931	1.4	27.978	1.862
1.6	30.951	0.1577	1.6	30.951	0.789	1.6	30.951	1.577
1.8	33.372	0.1371	1.8	33.372	0.686	1.8	33.372	1.371
2.0	35.317	0.1215	2.0	35.317	0.607	2.0	35.317	1.215

For an overdamped and a critically damped system with or without time delay, a mechanism for parameter estimation is developed in this work, from three-time instants  $t_5$ ,  $t_{10}$  and  $t_{90}$  of transient step response, as shown in Figure 3.8.

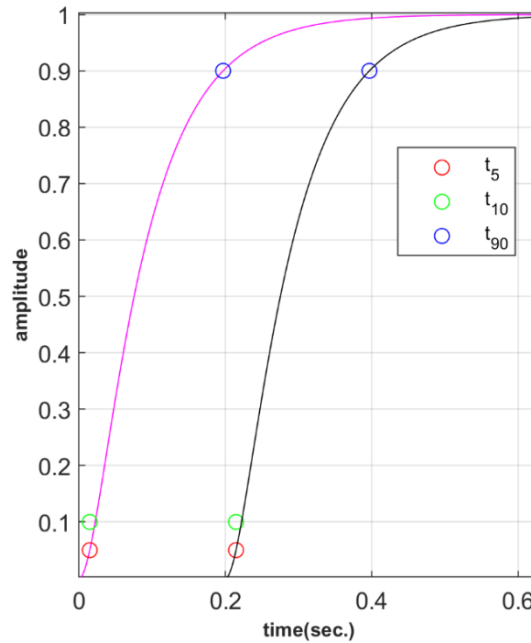


Figure 3.8: System response at three-time instants (Color magenta  $t_d = 0$  sec,  $\xi = 1.2$ ,  $\omega_n = 12 \frac{rad}{sec}$   
Color black  $t_d = 0.2$  sec,  $\xi = 1.2$ ,  $\omega_n = 12 \frac{rad}{sec}$ )

### 3.3.2.1 Estimation of damping ratio $\xi$ (dead time $t_d \neq 0$ )

For a fixed value of  $\omega_n$  and  $t_d \neq 0$  in Equation (29) we obtain the step response data by varying the damping ratio in the range  $1 \leq \xi \leq 2$  whence we compute  $T_R = \frac{t_{90}-t_5}{t_{10}-t_5}$ . It is experimentally determined that  $T_R$  has a quadratic relationship with  $\xi$  (refer to Figure 3.9) and it remains independent of  $\omega_n$  (refer to Table 3.7).

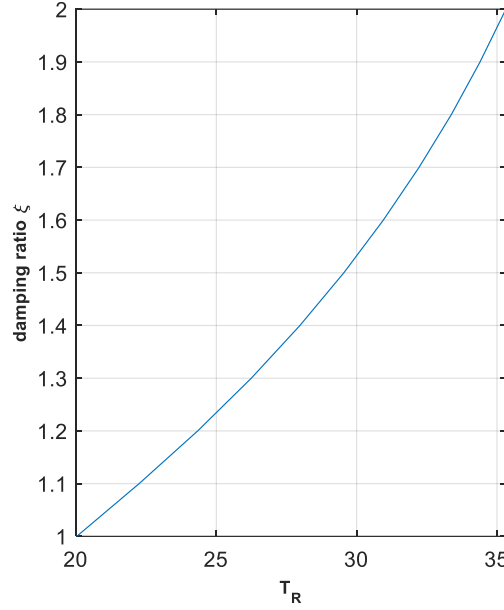


Figure 3.9: Quadratic relation between damping ratio  $\xi$  and  $T_R$  for all  $\omega_n$

Without any loss of generality, we may write a quadratic relationship between the estimated value of  $\xi$ , viz.  $\xi_{\text{est}}$  and  $T_R$  as:

$$\xi_{\text{est}(T_R)} = \alpha_1 T_R^2 + \alpha_2 T_R + \alpha_3 \quad (30)$$

where  $\alpha_1, \alpha_2$  and  $\alpha_3$  are coefficients of the polynomial  $\xi(T_R)$ , whose values are determined in the range  $1 \leq \xi \leq 2$  by minimizing the sum of the squares of the deviations of  $\xi$  from the one obtained from (30) (least square fit). The values obtained are:

$$\alpha_1 = 2.0489 \times 10^{-3}, \alpha_2 = -4.9765 \times 10^{-2}, \alpha_3 = 1.18660 \quad (31)$$

Hence (30) becomes

$$\xi_{\text{est}} = (2.0489 \times 10^{-3}) T_R^2 - (4.9765 \times 10^{-2}) T_R + 1.18660 \quad (32)$$

Figure 3.10 illustrates a comparison plot of  $\xi$  and  $\xi_{\text{est}}$ .

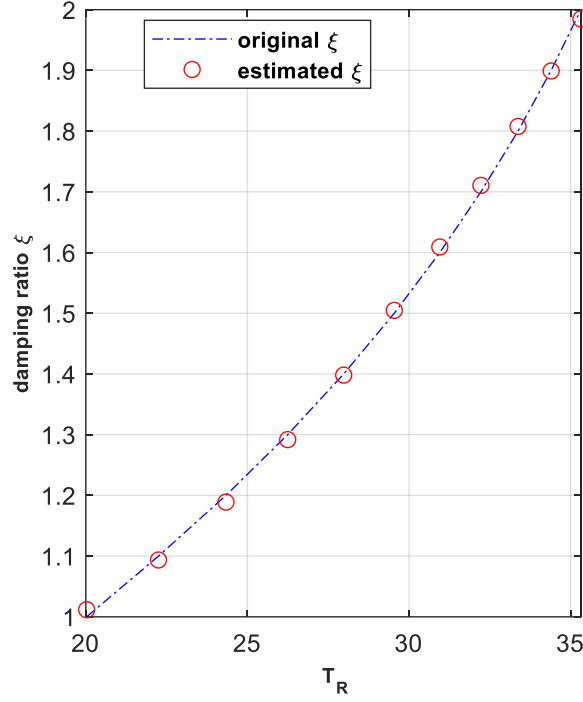


Figure 3.10: Comparison plot of damping ratio  $\xi$  and  $\xi_{\text{est}}$  vs  $T_R$  for all  $\omega_n$

### 3.3.2.2 Estimation of undamped natural frequency $\omega_n$ (dead time $t_d \neq 0$ )

For a given value of  $\omega_n$  and  $\xi$  in (29) with  $t_d \neq 0$ , we obtain the step response from which we calculate  $f_r = \frac{1}{t_{90}-t_{10}}$ . It is observed that for a given damping ratio  $\xi$  if the undamped natural frequency  $\omega_n$  is varied from 1 Hz to 1000 Hz, and the corresponding values of  $t_{90}$  and  $t_{10}$  are calculated, then  $\omega_n$  bears a linear relationship with  $f_r$ . Figure 3.11 illustrates the aforesaid series of curves for different values of  $\xi \in [1, 2]$ .

Without any loss of generality, for the six different values of  $\xi \in \{1, 1.2, 1.4, 1.6, 1.8, 2\}$ , we may write the corresponding linear relationships between  $\omega_n$  and  $f_r$  as:

$$\omega_{n_{\text{est}}}(f_r) = m_i f_r + c_i \quad (33)$$

for  $i=1, 2, 3, \dots, 6$

where  $m_i$  and  $c_i$  are coefficients of  $\omega_{n_{\text{est}}}(f_r)$ , for the corresponding value of  $\xi$ . The six sets of values of  $m_i$  and  $c_i$  are determined by minimizing the sum of the squares of the deviations in  $\omega_n$  from the corresponding estimates as obtained from (33) (least square fit). The values of  $m_i$  and  $c_i$  thus determined for the six different values of  $\xi$  are expressed in the following vector form:

$$m = [3.357 \quad 4.371 \quad 5.370 \quad 6.340 \quad 7.291 \quad 8.229] \quad (34)$$

$$c = 10^{-11}[-0.0131 \quad 0.1204 \quad -0.1524 \quad 0.1938 \quad 0.3896 \quad 0.2691] \quad (35)$$

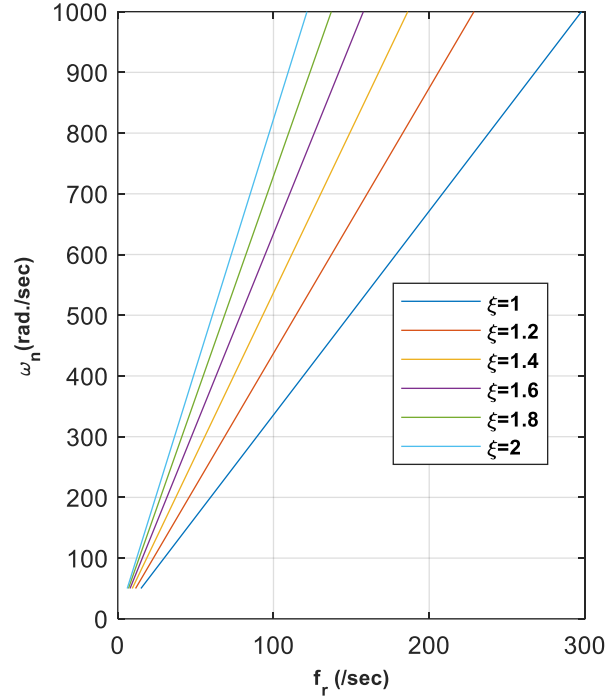


Figure 3.11: Linear relationship between  $\omega_n$  and  $f_r$  for a given  $\xi$

The Pearson correlation coefficient (denoted as  $r_{xy}$ ) measures how two variables ( $x$  &  $y$ ) are related in a linear way, with values between -1 (perfect negative relationship) and 1 (perfect positive relationship). It's calculated using the covariance of the variables i.e.  $cov(xy)$  and divides it by the product of their standard deviation i.e.  $\sigma_x \cdot \sigma_y$  as given below:

$$r_{xy} = \frac{cov(xy)}{\sigma_x \sigma_y} \quad (36)$$

The Pearson correlation coefficient  $r_{\xi m}$  between the elements of vectors  $\mathbf{m}$  and the corresponding values of  $\xi$  is found to be 0.998. This high value of 0.998 indicates a very strong linear relationship between these two variables. In other words, as one variable ( $\xi$ ) changes, the other ( $m$ ) tends to change in a very predictable and closely related manner. Also, the Pearson correlation coefficient  $r_{\xi c}$  between the variables  $\xi$  and corresponding vector  $\mathbf{c}$  is found to be 0.322.

So, without any loss of generality, we may write (37) and (38).

$$m_{est}(\xi) = A_1 \xi + A_2 \quad (37)$$

$$c_{est}(\xi) = B_1 \xi + B_2 \quad (38)$$

where  $A_1$  and  $A_2$  are coefficients of the polynomial  $m(\xi)$ ,  $B_1$  and  $B_2$  are coefficients of the polynomial  $c(\xi)$ , whose values are determined by minimizing the sum of the squares of the deviations of the actual values of  $m_i$  and  $c_i$  from their estimated values  $m_{est}$  &  $c_{est}$  as obtained from (37) and (38) (least square fit) respectively. The coefficients obtained are:

$$A_1 = 4.869, \quad A_2 = -1.477, \quad B_1 = 0.3665 \times 10^{-11}, \quad B_2 = -0.4152 \times 10^{-11} \quad (39)$$

Substituting these values in Equation (37) and (38) to obtain  $m_{est}$  and  $c_{est}$ , (33) modifies to (40).

$$\omega_{n_{est}} = (4.869 \xi - 1.477)f_r + 10^{-11} (0.3665 \xi - 0.4152) \quad (40)$$

Values of  $B_1$  and  $B_2$  are so negligible that if ignored do not affect the calculation of  $\omega_{n_{est}}$ . Hence  $\omega_{n_{est}}$  maybe written as:

$$\omega_{n_{est}} = (4.869 \xi - 1.477)f_r \quad (41)$$

Figure 3.12 illustrates a comparison plot of  $\omega_n$  and  $\omega_{n_{est}}$ .

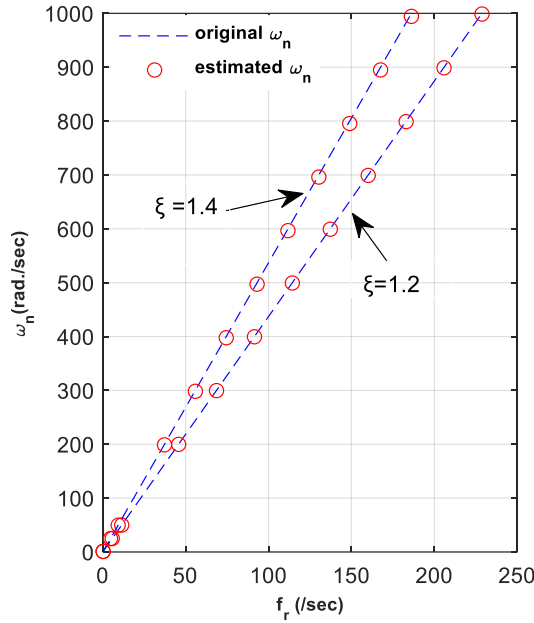


Figure 3.12: Comparison plot between  $\omega_n$  and  $\omega_{n_{est}}$  vs  $f_r$  for a given  $\xi$

### 3.3.2.3 Estimation of dead time $t_d$

From the three points of the step response transient, viz.  $t_5$ ,  $t_{10}$  and  $t_{90}$  of the given SOPDT system,  $\xi_{est}$  and  $\omega_{n_{est}}$  are determined from (32) and (41) respectively. With these values of  $\xi_{est}$  and  $\omega_{n_{est}}$  we simulate the step response of the system with zero dead time and again find  $t_{90}$  from the simulated time response. The difference of the  $t_{90}$  of the SOPDT system and that of the simulated system will give a measure of the dead time  $t_d$ .

### 3.3.2.4 Estimation of $\xi$ and $\omega_n$ (dead time $t_d = 0$ )

$\xi_{est}$  and  $\omega_{n_{est}}$  for overdamped and critically damped systems with  $t_d = 0$  are obtained by acquiring the three points of their transient response to a step input, calculating  $T_R$  &  $f_r$  and then using the same equations (32) and (41). Since a common algorithm runs for both the systems, viz. with and without dead time, a proper threshold sets  $t_d = 0$  for systems without dead time.

### 3.3.3 The Proposed Algorithm for overdamped/critically damped system

Thus, the general parameter estimation algorithm generates the transfer function parameters by using three instants of the step response curve, viz.  $t_5$ ,  $t_{10}$  and  $t_{90}$  is as listed below:

**Step 1:** From the step response curve of any overdamped SOPDT obtain  $t_5$ ,  $t_{10}$  and  $t_{90}$

**Step 2:** Calculate the ratio  $T_R$  and use (32) to obtain an estimate of damping ratio  $\xi_{est}$ .

**Step 3:** Calculate  $f_r = \frac{1}{t_{90}-t_{10}}$ .

**Step 4:** Substitute the value of  $\xi_{est}$  obtained from Step2 and the value of  $f_r$  obtained from Step3 in (14) to obtain  $\omega_{n_{est}}$ .

**Step 5:** With the value of  $\xi_{est}$  obtained in Step 2 and  $\omega_{n_{est}}$  obtained in Step 4 simulate the system and obtain the step response and evaluate  $t_{90}$ , of the simulated system.

**Step 6:** The difference of  $t_{90}$ , in Step1 and Step 5 gives the dead time  $t_d$ . Apply thresholding to eliminate negligible non-zero measures of  $t_d$  for systems with zero dead time.

### 3.3.4 Validation of the algorithm

To evaluate the performance of the proposed algorithm two performance indices are utilized viz. the variance-accounted-for (VAF) and the integral of absolute error (IAE).

The VAF value signifies the consistency between the model and actual system as stated by Song et al., 2017. To check the accuracy of the model the VAF value of the recovered model is calculated as:

$$VAF = \left[ 1 - \frac{\text{var}(y_{actual} - y_{model})}{\text{var}(y_{actual})} \right] \times 100\% \quad (42)$$

where,

$y_{actual}$  = output from the actual system for a unit step input

$y_{model}$  = output from the model for the same input

Var = variance

So, a VAF of 100% ensures an exact match of the estimated system with the actual system.

The IAE measures the closeness of the step response of the original plant to that of the estimated plant and here is calculated as:

$$IAE = \sum_1^n |y_{actual}(t) - y_{model}(t)| \cdot \Delta t \quad (43)$$

where,



$n$  = number of sample points

$\Delta t$  = sampling interval

Thus, an exact match between the estimated system and the actual system ensures an IAE value of zero.

**Example 1:** Let the actual second order system be characterized by  $t_d = 0$ ,  $\xi = 1.4$ ,  $\omega_n = 4 \text{ rad/sec}$ . Applying here General Identification Algorithm, we get values as shown in Table 3.9.

**Table 3.9 Time for responses, estimated parameters, IAE and VAF for Example1**

$t_{90}$ (sec)	$t_{10}$ (se c)	$t_5$ (sec)	$\xi$ estimated	$\omega_n$ estimated (rad/sec)	$t_d$ estimated (sec)	IAE	VAF
1.485 5	0.142 9	0.0932	1.398	3.970	0	1.064	99.998 %

Step response of actual transfer function and model fit is shown in Figure 3.13.

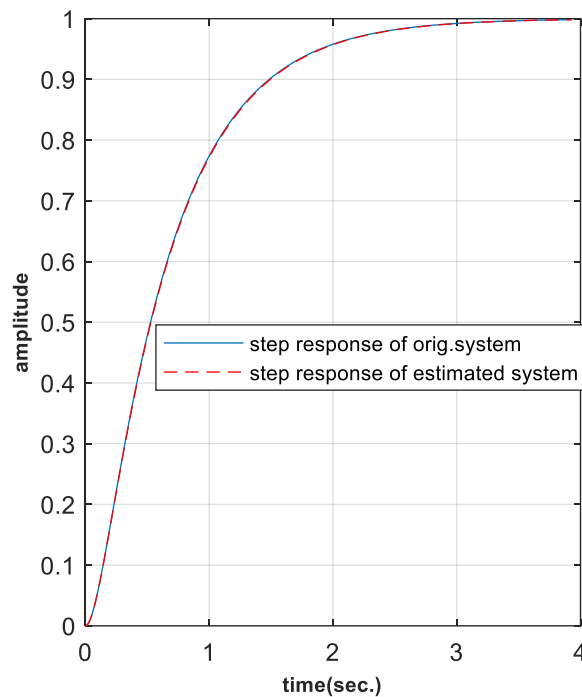


Figure 3.13: Comparison between step response of the actual system and the estimated model of Example 1

**Example 2** Let the actual second order system be characterized by  $t_d = 0.05$ ,  $\xi = 1.3$ ,  $\omega_n = 10 \text{ rad/sec}$ .

Applying here General Identification Algorithm we get values as shown in Table 3.10. Step response of actual transfer function and model fit is shown in Figure 3.14.

**Table 3.10 Time for responses, estimated parameters, IAE and VAF for Example2**

$t_{90}$ (sec)	$t_{10}$ (sec)	$t_5$ (sec)	$\xi$ estimated	$\omega_n$ estimated (rad/sec)	$t_d$ estimated (sec)	IAE	VAF
0.593	0.106	0.086	1.292	9.874	0.0472	0.7532	99.999%

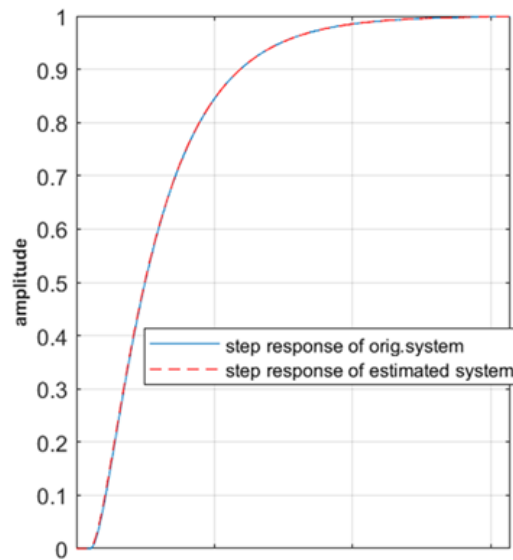


Figure 3.14: Comparison between step response of the actual system and the estimated model of Example 2

Estimation of parameters of arbitrary critically damped and overdamped systems with and without dead times are carried out with the proposed estimation algorithm and the corresponding IAE and VAF values are calculated and compared with the same for two existing methods in literature (Rangaiah and Krishnaswami, 1994[6] and Tsung and Huang, 1993[8]). The results are tabulated in Table 3.11.

**Table 3.11 Comparative Study using General Identification Algorithm**

S.No.	System Specifications	Estimated Parameters Proposed Method	Estimated Parameters Rangaiah Method,1994 [6]	Estimated Parameters Tsung Method, 1993[8]
<b>1</b>	$\xi$ <b>1</b>	0.986	1.004	0.9886
	$\omega_n$ (rad./sec) <b>6</b>	5.9140	5.999	5.8771
	$t_d$ (sec.) <b>0</b>	0	0	0
	IAE	0.9608	0.0504	1.444
	VAF %	99.991	99.999	99.995
<b>2</b>	$\xi$ <b>1.1</b>	1.093	1.101	1.072
	$\omega_n$ (rad./sec) <b>20</b>	19.925	20.017	19.354
	$t_d$ (sec.) <b>0</b>	0	0	0
	IAE	0.9648	0.0732	3.476
	VAF %	99.994	99.999	99.997
<b>3</b>	$\xi$ <b>1.2</b>	1.188	1.200	1.179
	$\omega_n$ (rad./sec) <b>5</b>	4.932	5.003	4.863
	$t_d$ (sec.) <b>0</b>	0	0	0
	IAE	0.964	0.084	2.659
	VAF %	99.994	99.999	99.989
<b>4</b>	$\xi$ <b>1.3</b>	1.291	1.299	1.298
	$\omega_n$ (rad./sec) <b>12</b>	11.848	12.001	11.915
	$t_d$ (sec.) <b>0</b>	0	0	0
	IAE	1.0601	0.0667	0.1715
	VAF %	99.997	99.999	99.999
<b>5</b>	$\xi$ <b>1.4</b>	1.391	1.399	1.419
	$\omega_n$ (rad./sec) <b>8</b>	7.914	7.986	7.497
	$t_d$ (sec.) <b>0.200</b>	0.1999	0.2000	0.2001
	IAE	1.211	0.2543	8.148
	VAF %	99.997	99.999	99.927
<b>6</b>	$\xi$ <b>1.5</b>	1.504	1.499	1.531
	$\omega_n$ (rad./sec) <b>11</b>	10.948	10.985	10.798
	$t_d$ (sec.) <b>0.08</b>	0.079	0.079	0.080

	IAE	1.3685	0.1892	8.561
	VAF %	99.996	99.998	99.959
7	$\xi$ 1.6	1.609	1.599	1.631
	$\omega_n$ (rad./sec) 15	15.041	14.993	15.332
	$t_d$ (sec.) 0.06	0.058	0.059	0.061
	IAE	1.171	0.0636	5.878
	VAF %	99.996	99.998	99.983
8	$\xi$ 1.7	1.7105	1.7003	1.717
	$\omega_n$ (rad./sec) 10	10.492	9.962	11.228
	$t_d$ (sec.) 0.08	0.0682	0.0699	0.0701
	IAE	0.7020	0.1295	7.264
	VAF %	99.999	99.999	99.953
9	$\xi$ 1.8	1.807	1.8005	1.7953
	$\omega_n$ (rad./sec) 18	18.079	17.997	17.9609
	$t_d$ (sec.) 0.3	0.2999	0.2999	0.3001
	IAE	0.1409	0.0445	1.0679
	VAF %	99.999	99.999	99.999
10	$\xi$ 1.9	1.898	1.9004	1.8726
	$\omega_n$ (rad./sec) 6	6.028	6.000	6.579
	$t_d$ (sec.) 0.05	0.0504	0.0499	0.0499
	IAE	0.538	0.0443	4.7008
	VAF %	99.999	99.999	99.993
11	$\xi$ 2.0	1.984	2.000	1.9596
	$\omega_n$ (rad./sec) 1	0.9949	1.0001	0.9791
	$t_d$ (sec.) 0.2	0.228	0.199	0.1939
	IAE	1.274	0.036	5.336
	VAF %	99.956	99.999	99.915

*\*\*Systems in S.No.1,2,3,4 have estimated  $t_d$  after thresholding.*

### 3.3.5 Effect of noise

Measurement noise usually corrupts the system output data and hence performance of the estimation algorithms needs evaluation in presence of measurement noise to check its robustness. The noise signal considered here is assumed White Gaussian with zero mean and a

specified standard deviation of 1% of steady-state value. Representative results for the proposed estimation method and the other two methods cited in literature [6] and [8] shown in Table 3.12.

**Table 3.12 Comparative Study using noisy process response**

S.No.	System Specifications	Estimated Parameters Proposed Method	Estimated Parameters Rangaiah Method,1994[6]	Estimated Parameters Tsung Method ,1993[8]
<b>1</b>	$\xi$ <b>1</b>	1.028	0.909	0.939
	$\omega_n$ (rad./sec) 10	10.557	8.932	9.385
	** $t_d$ (sec.) 0	0	0	0
	IAE	9.258	15.794	12.529
	VAF %	99.809	99.501	99.683
<b>2</b>	$\xi$ <b>1.1</b>	1.168	0.989	1.0057
	$\omega_n$ (rad./sec) 15	16.197	18.694	17.554
	** $t_d$ (sec.) 0	0	0	0
	IAE	11.123	18.125	16.307
	VAF %	99.722	99.421	99.541
<b>3</b>	$\xi$ <b>1.2</b>	1.224	1.0933	1.092
	$\omega_n$ (rad./sec) 5	5.432	4.553	4.4562
	** $t_d$ (sec.) 0	0	0	0
	IAE	13.334	17.2648	17.0804
	VAF %	99.681	99.552	99.532
<b>4</b>	$\xi$ <b>1.3</b>	1.304	1.1658	1.1844
	$\omega_n$ (rad./sec) 12	12.257	10.7261	11.238
	** $t_d$ (sec.) 0	0	0	0
	IAE	10.492	19.962	17.305
	VAF %	99.623	99.5022	99.552
<b>5</b>	$\xi$ <b>1.4</b>	1.562	1.317	1.341
	$\omega_n$ (rad./sec) 4	4.028	3.874	3.9009
	$t_d$ (sec.) 0.200	0.200	0.2000	0.1997
	IAE	13.783	28.374	23.312
	VAF %	99.845	99.361	99.576
	$\xi$ <b>1.5</b>	1.455	1.339	1.409
	$\omega_n$ (rad./sec) 3.5	3.4626	3.1558	3.3741

<b>6</b>	$t_d$ (sec.)	0.08	0.0781	0.0777	0.079
	IAE		10.737	43.374	21.498
	VAF %		99.990	98.917	99.757
<b>7</b>	$\xi$	<b>1.6</b>	1.539	1.229	1.391
	$\omega_n$ (rad./sec)	15	13.545	11.128	13.512
	$t_d$ (sec.)	0.03	0.0215	0.0215	0.0558
	IAE		10.279	33.643	36.373
	VAF %		99.857	99.273	99.214
<b>8</b>	$\xi$	<b>1.7</b>	1.724	1.298	1.616
	$\omega_n$ (rad./sec)	10	9.4927	7.8784	8.721
	$t_d$ (sec.)	0.08	0.0684	0.068	0.0710
	IAE		14.001	142.90	47.376
	VAF %		99.868	85.037	97.870
<b>9</b>	$\xi$	1.8	1.709	1.352	1.6489
	$\omega_n$ (rad./sec)	8	7.515	10.418	11.6123
	$t_d$ (sec.)	0.3	0.297	0.2889	0.3002
	IAE		10.548	145.84	25.148
	VAF %		99.89	87.903	99.702
<b>10</b>	$\xi$	<b>1.9</b>	1.911	1.217	1.518
	$\omega_n$ (rad./sec)	2.8	3.0972	1.777	2.2900
	$t_d$ (sec.)	0.05	0.0506	0.039	0.0481
	IAE		18.900	127.19	64.856
	VAF %		99.779	91.963	98.313
<b>11</b>	$\xi$	<b>2.0</b>	1.849	1.470	1.745
	$\omega_n$ (rad./sec)	1	0.9808	0.729	0.883
	$t_d$ (sec.)	0.2	0.1596	0.018	0.101
	IAE		11.324	75.316	37.827
	VAF %		99.842	97.356	99.411

*\*\*Systems in S.No.1,2,3,4 have estimated  $t_d$  after thresholding.*

It is evident from this table that the process parameters estimated under noisy conditions by the proposed method are better than those estimated by the other two methods [6] and [8], thereby its robustness. A comparison between the original noisy response and the recovered

model responses is shown in Figure 3.15 for the SOPDT system with  $t_d = 0 \text{ sec}$ ,  $\xi = 1.2$ ,  $\omega_n = 5 \text{ rad/sec}$  and in Figure 3.16 for SOPDT system with  $t_d = 0.03 \text{ sec}$ ,  $\xi = 1.6$ ,  $\omega_n = 15 \text{ rad/sec}$ .

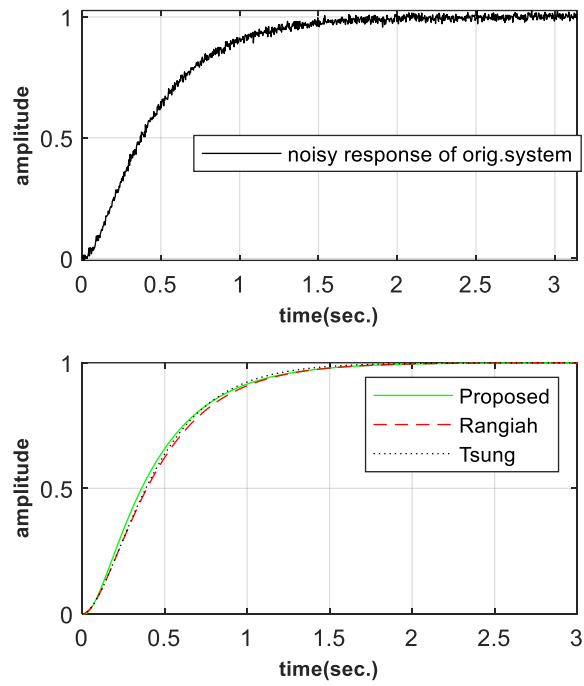


Figure 3.15: Comparison under the noisy process response for  $t_d = 0 \text{ sec}$ ,  $\xi = 1.2$ ,  $\omega_n = 5 \text{ rad/sec}$

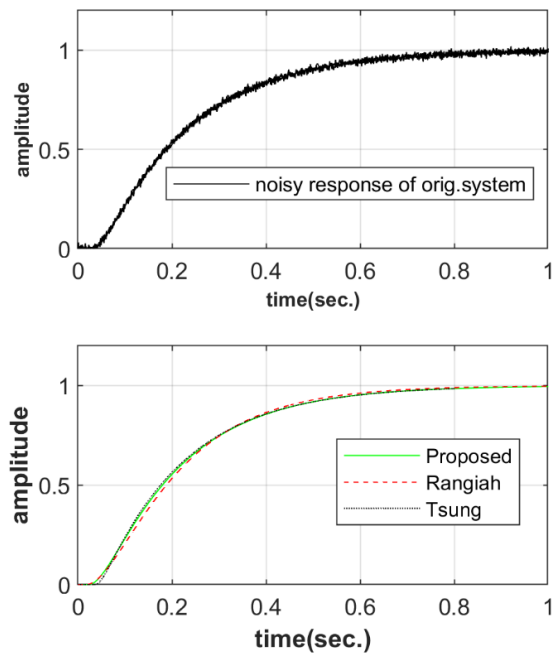


Figure 3.16: Comparison under the noisy process response for  $t_d = 0.03 \text{ sec}$ ,  $\xi = 1.6$ ,  $\omega_n = 15 \text{ rad/sec}$

### 3.3.6 Conclusion

This work presents a simple and elegant method for estimating, with reasonable accuracy, the parameters of a critically damped or overdamped second-order system with and without dead time. The quantitative indices for measuring the accuracy of the closeness of response of the estimated SOPDT system with the actual system, viz. VAF and IAE, illustrate comparable results with the existing estimation schemes [6] and [8]. The computational complexity is significantly reduced compared to other estimation methods as reported in the literature. The illustrative examples also clearly indicate that the proposed method is more robust against measurement noise compared to the other two methods ([6] and [8]).





## **Chapter 4**

---

# **Modelling of Fast Steering Mirror Assembly -A Case Study**

## 4.1 Introduction

This work introduces a systematic black-box approach to model a dual-axis tip-tilt fast steering mirror assembly, featuring integrated piezo actuators and pick-off sensors. The method employs a two-fold paradigm utilizing time response (TR) or frequency response (FR) data, contingent on the system's damping characteristics. It estimates the cross-coupling coefficient between the two axes by transforming the time response signal into the frequency domain using the FFT of the output on one axis due to a sinusoidal input on the other axis. The TR method employs step response data, collected via a digital storage oscilloscope, to formulate a mathematical model. Meanwhile, the FR method employs sinusoidal excitation to derive the frequency response, combined with a least square estimator. Operating in open-loop mode, the algorithm obviates the need for controller insight and works uniformly across all damping cases. Validation depends on data type: TR modelling is verified in the frequency domain and FR in the time domain. Model accuracy is assessed using variance accounted for (VAF) and compared against conventional data-driven methods.

The Chapter is organized as follows:

Section 4.2 gives the problem formulation and the choice of the model for parameter estimation of the dual-axis FSM system. Section 4.3 introduces the theoretical and mathematical basis for the estimation procedure of the TR and FR methods. Section 4.4 presents the experimental setup for the parameter estimation of the dual-axis FSM system. Section 4.5 presents the validation of the algorithm by comparison with three published methods [109],[110], and [111]. Finally, the concluding remarks are presented in Section 4.6.

## 4.2 Problem Statement

For a dual-axis (x and y) FSM assembly, an actuator input voltage from the digital drive-in y-axis will rotate the FSM about the x-axis, and thus sensor output is obtained in the x-direction. To account for the time lag between the actuator input and the sensor output, a dead time  $t_d$  is added to the model of the FSM assembly which is considered a second-order system in line with [54] and [55]. In the absence of cross-coupling between the axis, both axis assumes an identical second-order plus dead time (SOPDT) model characterized by (1)

$$G(s) = Ke^{-t_d s} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right) \quad (1)$$

K is the system gain,  $t_d$  is the dead time,  $\omega_n$  is the undamped natural frequency and  $\xi$  is the damping coefficient. The first problem addressed in this paper deals with the estimation of these four parameters that characterize the model in (1) using the TR algorithm or FR algorithm. The second problem deals with the determination of the cross-coupling factor of the dual-axis FSM system by capturing the time domain input-output signals of the x-channel and y-channel respectively and converting them to the frequency domain using the FFT algorithm.

The FSM system is a dual axis system and is capable of rotation about the x-axis and y-axis. The transfer function  $T_{x1}$  for a single axis relates the input voltage  $V_y$  (given to the y- channel of the d-drive) to the response angle  $\theta_x$  by which the FSM is rotated (in the horizontal x-direction)

For the dual input dual output FSM system used here, the transfer function T of the system can then be written as:

$$T = \begin{bmatrix} T_{x1} & T_{x2} \\ T_{y1} & T_{y2} \end{bmatrix} = \begin{bmatrix} \frac{\theta_x}{V_y} & \frac{\theta_x}{V_x} \\ \frac{\theta_y}{V_y} & \frac{\theta_y}{V_x} \end{bmatrix} \quad (2)$$

To capture the effect of cross-coupling between the two axis the state space model of the FSM is formulated as given in (3) and (4). The block diagram of the state space model is given in Fig.4.1.

$$\begin{bmatrix} \dot{\theta}_x \\ \ddot{\theta}_x \\ \dot{\theta}_y \\ \ddot{\theta}_y \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\omega_x^2 & -2\xi_x\omega_x & A_x\omega_x^2 & 2A_x\xi_x\omega_x \\ 0 & 0 & 0 & 1 \\ A_y\omega_y^2 & 2A_y\xi_y\omega_y & -\omega_y^2 & -2\xi_y\omega_y \end{bmatrix} \begin{bmatrix} \theta_x \\ \dot{\theta}_x \\ \theta_y \\ \dot{\theta}_y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ K\omega_x^2 & 0 \\ 0 & 0 \\ 0 & K\omega_y^2 \end{bmatrix} \begin{bmatrix} V_y \\ V_x \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} \theta_x \\ \theta_y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \theta_x \\ \dot{\theta}_x \\ \theta_y \\ \dot{\theta}_y \end{bmatrix} \quad (4)$$

where  $A_x$  and  $A_y$ , are cross-coupling coefficients about the x-axis and y-axis respectively, and  $\theta_x$  and  $\theta_y$  are rotation of FSM about the x-axis and y-axis respectively,  $\omega_x$  and  $\omega_y$  are undamped natural frequency about x-axis and y-axis respectively and  $\xi_x$  and  $\xi_y$  are damping coefficient about x-axis and y-axis respectively.

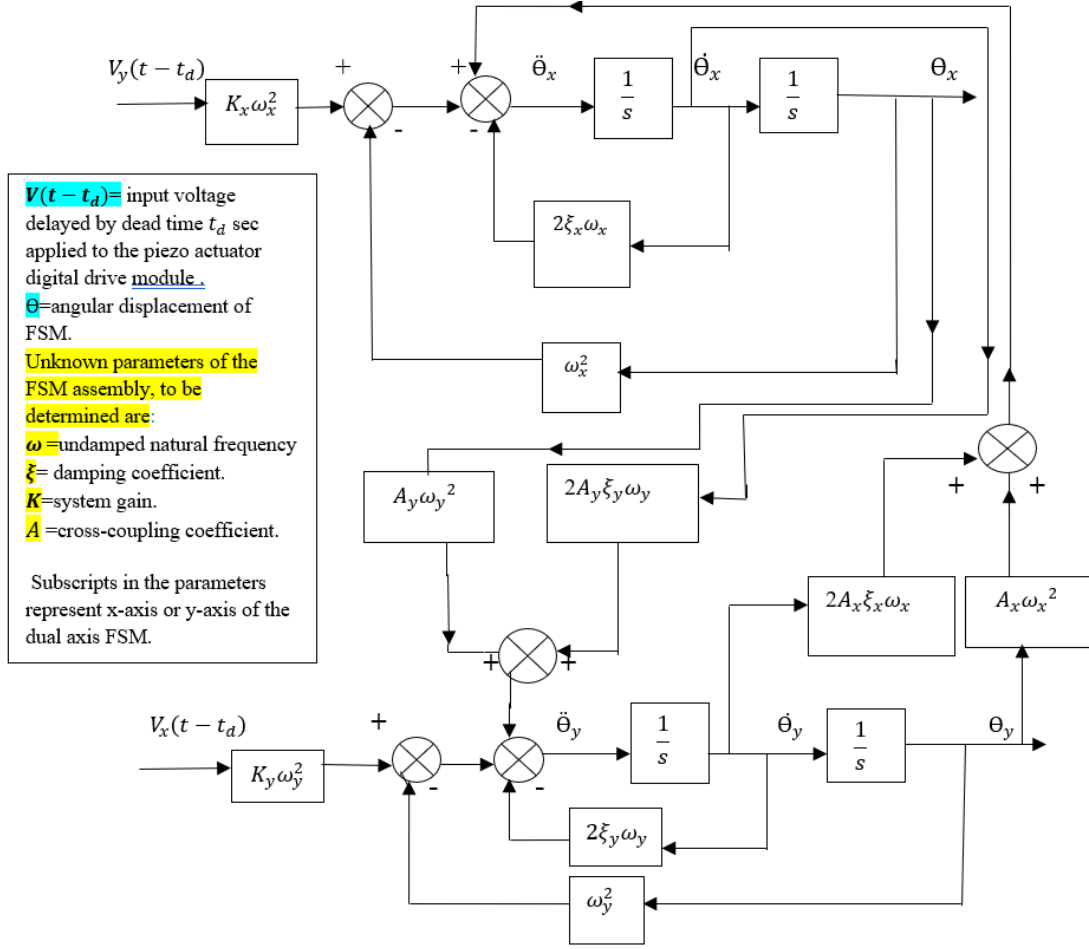


Figure 4.1 State Space Model of dual axis FSM assembly

### 4.3 Estimation Procedure

Estimating the four parameters of the FSM system of (1) is done by FR or TR algorithm depending on whether the time response of the system has mild/no oscillations (heavy damping) or strong oscillations (mild damping) respectively. Making the proper choice of FR method or TR method with an emphasis on the degree of oscillation in the step response of the FSM system demands a deeper understanding of this concept and has been taken care of in the subsequent subheadings of this section.

#### 4.3.1 Frequency Response (FR) Method

In this method frequency response data of the FSM assembly is used to generate the FSM open loop model. Using self-tuning controllers like PID, the work of [24] obtained parameters of the SOPDT closed loop

system in the frequency domain. Unlike [24], the FR method used here does parameter estimation of SOPDT system in open loop mode.

The actual open loop frequency response is obtained by driving the FSM assembly by a sinusoid whose frequency is varied from 1 Hz to 1000 Hz. The SOPDT of FSM (1) may be re-written as

$$F(s) = Ke^{-t_d s} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right) \quad (5)$$

Therefore

$$\frac{1}{F(s)} = \frac{(s^2 + 2\xi\omega_n s + \omega_n^2)e^{t_d s}}{K\omega_n^2} \quad (6)$$

Let  $|Mag(j\omega)|$  and  $Ph(j\omega)$  denote the magnitude and phase respectively at frequency  $\omega$  obtained from frequency response data and  $|F(j\omega)|$  and  $\arg(F(j\omega))$  represent the magnitude and phase respectively of the model given in (5). Applying least square estimation to the magnitude obtained from the model and actual open loop frequency response.

$$J_{mag} = \sum_{\omega} \left[ \left| \frac{1}{Mag(j\omega)} \right|^2 - \left| \frac{1}{F(j\omega)} \right|^2 \right]^2 \quad (7)$$

where

$$\frac{1}{F(j\omega)} = \frac{\omega_n^2 - \omega^2 + j2\xi\omega_n \omega}{K\omega_n^2} e^{t_d j\omega} \quad (8)$$

Since  $|e^{t_d j\omega}| = 1$ , we get

$$\left| \frac{1}{F(j\omega)} \right| = \sqrt{\frac{(\omega_n^2 - \omega^2)^2 + (2\xi\omega_n \omega)^2}{K^2 \omega_n^4}} \quad (9)$$

$$J_{mag} = \sum_{\omega} \left[ \left| \frac{1}{Mag(j\omega)} \right|^2 - \left( \frac{\omega^4}{K^2 \omega_n^4} + \left( \frac{-2\omega_n^2 + 4\xi^2 \omega_n^2}{K^2 \omega_n^4} \right) \omega^2 + \frac{1}{K^2} \right) \right]^2 \quad (10)$$

Since the process is second order the two estimators give the same result. Hence  $J_{mag} = 0$ .

Let

$$\frac{1}{K^2 \omega_n^4} = x, \left( \frac{-2\omega_n^2 + 4\xi^2 \omega_n^2}{K^2 \omega_n^4} \right) = u, \frac{1}{K^2} = z \quad (11)$$

Then we get

$$\sum_{\omega} \left| \frac{1}{Mag(j\omega)} \right|^2 = x \sum_{\omega} \omega^4 + u \sum_{\omega} \omega^2 + z \sum_{\omega} 1 \quad (12)$$

$$\sum_{\omega} \left| \frac{1}{Mag(j\omega)} \right|^2 \omega^2 = x \sum_{\omega} \omega^6 + u \sum_{\omega} \omega^4 + z \sum_{\omega} \omega^2 \quad (13)$$

$$\sum_{\omega} \left| \frac{1}{Mag(j\omega)} \right|^2 \omega^4 = x \sum_{\omega} \omega^8 + u \sum_{\omega} \omega^6 + z \sum_{\omega} \omega^4 \quad (14)$$

Solving (12-14), by selecting eight to ten frequencies in the interval  $(0, \omega_c]$  where  $\omega_c$  is the cut-off frequency, obtained from the open loop FR data, the values of u, x, z are obtained. Thus from (11) we get

$$K = \frac{1}{\sqrt{z}}, \quad \omega_n = \left( \frac{1}{xK^2} \right)^{\frac{1}{4}}, \quad \xi = \sqrt{\frac{uK^2\omega_n^4 + 2\omega_n^2}{4\omega_n^2}} \quad (15)$$

Thus, steady-state gain K, the model natural frequency  $\omega_n$  and damping coefficient  $\xi$  can be estimated from (15).

Applying least square estimation to the phase obtained from the model  $\arg(F(j\omega))$  and phase  $Ph(j\omega)$  from actual open loop frequency response, we get

$$J_{Phase} = \sum_{\omega} \left[ \frac{1}{Ph(j\omega)} - \arg \left( \frac{1}{F(j\omega)} \right) \right]^2 \quad (16)$$

$$J_{phase} = \sum_{\omega} \left[ \frac{1}{Ph(j\omega)} - t_d \omega - \arg \left( \frac{\omega_n^2 - \omega^2 + j2\xi\omega_n\omega}{K\omega_n^2} \right) \right]^2 \quad (17)$$

Since the process is second order, hence  $J_{Phase} = 0$

$$t_d = \frac{\sum_{\omega} \left[ \left( \frac{1}{Ph(j\omega)} \right) - \arg \left( \frac{\omega_n^2 - \omega^2 + j2\xi\omega_n\omega}{K\omega_n^2} \right) \right] \omega}{\sum_{\omega} \omega^2} \quad (18)$$

Therefore, the dead time can be obtained from (18).

The FR scheme cannot be applied to strongly underdamped i.e  $\xi < 0.7$  (oscillatory) cases as the frequency response of the magnitude plot will have a peak in the vicinity of the natural frequency  $\omega_n$  which may or may not be greater than  $\omega_c$  (-3dB cut-off frequency). Thus, for frequencies in the interval  $(0, \omega_c]$ , there will be one magnitude corresponding to two frequencies, making (12-14) non-unique, and hence will lead to a discrepancy in the parameter estimation.

#### 4.3.2 Time Response (TR) Method

The FSM assembly is operated in open loop mode and driven by a unipolar square wave (5Hz and 15Hz) to generate the step response data. The time response data is acquired in the DSO in comma-separated variable (.csv) format and transferred to PC and analysed using MATLAB, whence the values of dead time  $t_d$ , peak voltage  $V_{pp}$ , peak time  $t_p$ , peak overshoot  $M_p$ , and steady-state output  $V_{ss}$  are obtained. From these values, the system parameters of (1) are estimated. The TR method is applied to SOPDT system having strong oscillation so that  $V_{pp}$  and  $t_p$  can be estimated easily from the open loop time response. The dead time corresponds to the difference between the start time i.e., the time when the input starts its excitation and the

first time where the step response is greater than zero. The dead time corresponds to the difference between the start time viz the time where the input starts its excitation and the first time where the step response becomes greater than zero. The peak overshoot  $M_p$  and  $\xi$  is related by

$$\%M_p = 100 \frac{V_{pp} - V_{ss}}{V_{ss}} = e^{\frac{-\pi\xi}{\sqrt{1-\xi^2}}} \quad (19)$$

Hence  $\xi$  can be evaluated. The peak time  $t_p$  and natural frequency  $\omega_n$  are related by

$$t_p = \frac{\pi}{\omega_n(1-\xi^2)^{1/2}} \quad (20)$$

Hence  $\omega_n$  can be evaluated Also, steady state gain

$$K = \frac{V_{ss}}{V_{in}} \quad (21)$$

Thus, the four parameters  $K, \omega_n, \xi$  and  $t_d$ , are determined. The TR algorithm yields less accurate estimates if the time response of the FSM system has mild oscillations i.e.,  $\xi > 0.7$  as then peak voltage  $V_{pp}$ , peak time  $t_p$ , cannot be accurately identified.

#### 4.4 Experimental Set-up

The experimental test bench comprising of an FSM assembly, piezo-driver (d-drive), DSO and a signal generator are mounted on an optical bread board shown in Figure 4.2

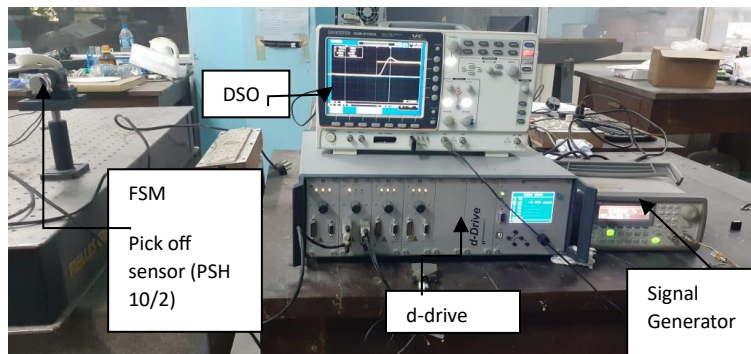


Figure 4.2: Experimental Set-up

The test-bench components are:

##### ***Fast Steering Mirror Assembly***

A two-axis FSM of PSX x/2 series [107] is used in the experimental setup. It is characterized by a maximum angular displacement of  $\pm 4$  mrad and a resonant frequency of 3500 Hz.

##### ***Digital drive (d-drive) Module***



The FSM assembly is driven by a piezo-actuator digital drive module (*d-drive*) from *M/S piezosystem jena* [107] with in-built function generator, noise injector, digital PID controller, a display module and a serial interface for PC connectivity). The d-drive has the MOD input and the MON output with special features mentioned below:

#### ***Modulation input: MOD***

The motion of the actuator can be remotely controlled using this input. The control signal applied through the MOD input must be in the range of 0 to +10 V. In this test procedure signal from a function generator is used to feed the MOD input.

#### ***Monitor Output: MON***

With a specific command, different signals can be taken out from the MON output. The voltage range of the signal taken from MON is scaled to 0 to +10 and can be monitored in an oscilloscope. In this test procedure the signal from the integrated pick-off sensor of the FSM is monitored.

#### ***Arbitrary Waveform Generator (AWG)***

An AWG, Agilent 33220A from *M/S Agilent Technology* is used to generate the input drive for the MOD input of the d-Drive unit.

#### ***Digital Storage Oscilloscope (DSO)***

A 70 MHz, 2 Gs/s, 2-channel DSO, GDS-2072A from *M/S GWINSTEK Inc* is used to acquire the response data from MON output of the d-Drive unit for storage and analysis.

To generate the time response and frequency response data of the FSM assembly the d-Drive is operated to work in open-loop mode, thereby bypassing the in-built digital PID block.

#### **4.4.1 Time Response (TR) Method [108]**

The FSM assembly is first driven by a unipolar square wave of amplitude 2 V (peak to peak) at frequencies 5 Hz and 15 Hz respectively to generate and acquire the step response data from the DSO, transferred to MATLAB environment, and the system parameters are obtained using (19)-(21). The time response performance indices (PIs) for a square wave drive at two different frequencies are listed in Table 4.1 and Table 4.2, for an x-axis drive. The y-axis drive yields an almost identical result. The Open loop FSM output with an input of 0-2V square wave of freq. 15Hz as acquired in DSO is shown in Figure 4.3.



Figure 4.3: Open loop FSM output with an input of 0-2V square wave of freq. 15Hz acquired in DSO

Hence, the overall system transfer function takes the form (23):

$$G_{FSM}(s) = (e^{-0.00057s}) \left( \frac{1304^2}{s^2 + 1304s + 1304^2} \right) \quad (22)$$

$$G_{FSM}(s) = (e^{-0.00057s}) \left( \frac{1.7004166 \times 10^6}{s^2 + 1304s + 1.700416 \times 10^6} \right) \quad (23)$$

**TABLE 4.1 Time Response data**

Freq. (Hz)	Peak Voltage $V_p$ (Volt)	Steady State Voltage $V_{ss}$ (Volt)	Dead time $t_d$ (msec)	Peak Time $t_p$ (msec)	%Peak overshoot $M_p$
5	2.28	2	0.57	2.84	14.0
15	2.28	2	0.57	2..84	14.0

**TABLE 4.2 Estimated System Parameters**

Freq. (Hz)	Dead time $t_d$ (msec)	$\omega_n$ (rad/sec)	$\xi$	K
5	0.57	1304	0.53	1
15	0.57	1304	0.53	1

The model so obtained is simulated in MATLAB to obtain its frequency response which is compared with the actual (FSM plant) frequency response to corroborate the results obtained by TR approach as shown in Figure 4.4.

Since -3dB BW gives a huge phase lag ( $<80^\circ$ ), hence usable BW is chosen at a phase lag of  $15^\circ$ . The usable bandwidth (BW) for the actual response is 28 Hz and that for the FSM model is 32 Hz. To have a closer look at the magnitude and phase errors between the actual response and FSM model response, the magnitude and phase errors are listed in Table 4.3. It is observed that up to 100 Hz, the magnitude errors (between actual response and model response) are within 0.5 dB, however the phase error gradually increases to about  $7.5^\circ$ . Since standard laser beam steering loops aim at a BW of around 50 Hz, the FSM model obtained is a good approximation of the actual FSM transfer function.

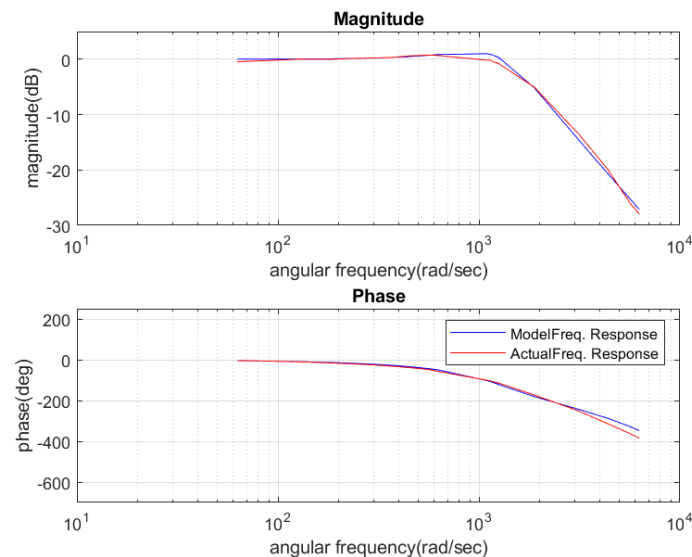


Figure 4.4: Bode plots of FSM model and actual FSM unit

**TABLE 4.3 Error between Actual and Model responses**

Freq.(Hz)	Magnitude Error	Phase Error
10	0.5	1.0
20	0.0	0.9
30	0.1	2.4
40	0.0	3.1
50	0.0	3.7
60	0.0	4.1
70	0.1	4.4
80	0.1	4.4
90	0.0	4.3
100	0.2	7.5
175	1.2	1.8
180	1.1	2.8
182	1.1	2.8
185	1.1	3.8
190	1.2	4.7
195	1.1	5.7
200	1.1	6.6
300	0.1	8.2
500	1.1	8.6
700	0.8	27.0
900	0.8	34.6
1000	0.8	37.4

The validity of the proposed algorithm is confirmed through the comparison of the TR model with actual frequency response data, revealing minimal magnitude errors and phase errors as listed in Table 4.3.

#### 4.4.2 Frequency Response (FR) Method

The FR of the FSM assembly is generated by driving the x-axis of the FSM by a sinusoid whose frequency is varied from 1Hz to 1000Hz and acquiring the angular tilt of the mirror position from the MON output of the d-drive, and then comparing it with drive input on the DSO for amplitude and phase measurement. The y-axis drive yields an almost identical result.

The cut-off frequency of the FSM from the plant's (FSM system) frequency response is found to be 1581 rad/sec (or 251 Hz).

Solving (12-14), by selecting ten frequencies in the interval (0,1581], the values of  $u$ ,  $x$ ,  $z$  are obtained as:

$$u = -5.891 \times 10^{-7}, x = 4.67 \times 10^{-13}, z = 1.011 \quad (22)$$

Using (15), we get

$$K = 0.994, \omega_n = 1212 \text{ rad/sec}, \xi = 0.53 \quad (23)$$

The dead time obtained from (18) is

$$t_d = 0.5 \text{ msec} \quad (24)$$

The model obtained by the FR method is

$$G_{\text{FSM}}(s) = (e^{-0.0005s}) \left( \frac{1212^2}{s^2 + 1212s + 1212^2} \right) \quad (25)$$

The TR of the model obtained from (25) for unit step input is corroborated by the actual TR of the FSM assembly in the MATLAB environment as shown in Figure 4.5.

The parameters obtained from the plot are a dead time of 0.57ms, a peak time of 2.8msec, and %peak overshoot of 16%. The model time response is not so close match to the actual time response (dead time of 0.57ms, peak time of 2.28 msec, and %peak overshoot of 14%) shown in Fig.4 for 15Hz. This discrepancy is due to the fact that while using the FR scheme for parameter estimation we have assumed the FSM system to be an SOPDT system with strong damping ( $\xi > 0.7$ ), although the FSM has mild damping ( $\xi < 0.7$ ).

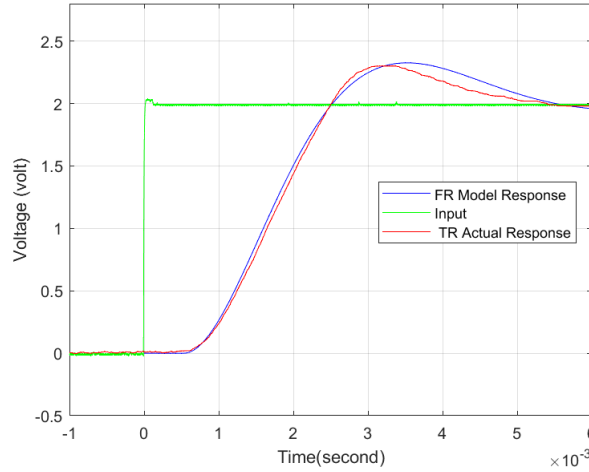


Figure 4.5 FR Model Response and TR Actual Response for 15Hz

#### 4.4.3 Cross-coupling

The x-channel of the FSM assembly is driven by a sinusoid of frequencies 1Hz to 20 Hz, and their corresponding output is observed in the y-channel. The input and output signals are acquired in the DSO in .csv format and transferred to PC where it is read and analysed in MATLAB by generating their FFT.

Cross-coupling at 15Hz, 10Hz, and 5Hz is shown in Fig. 4.6-4.8 respectively.

The cross-coupling factor is then determined easily as the ratio of the amplitude of output FFT to the amplitude of input FFT at that frequency as depicted in Table 4.5.

An identical set of readings was observed when input was given to the y-channel and output was taken from the x-channel. As the data obtained shows no definite dependency on the input drive frequency, the mean value of the cross-coupling coefficient  $A_y$  or  $A_x$  evaluated in the frequency range 1 to 20 Hz, is used as the cross-coupling coefficient of the model. From the data set listed in Table 4.4, the mean value of cross-coupling coefficient is  $9.867 \times 10^{-4}$ .

**TABLE 4.4 Cross-Coupling Coefficient**

<b>Frequency (Hz)</b>	<b>Cross-Coupling Coefficient <math>A_y</math> (or <math>A_x</math>)  (<math>\times 10^{-4}</math>)</b>
1	9.96
2	9.98
3	9.97
4	9.38
5	9.14
6	9.98
7	9.97
8	9.96
9	9.97
10	9.96
11	9.94
12	9.98
13	9.27
14	9.18
15	9.97
16	9.98
17	9.97
18	9.96
19	9.97
20	9.91

The state space model (3) is thus determined as follows:

$$\begin{bmatrix} \dot{\theta}_x \\ \ddot{\theta}_x \\ \dot{\theta}_y \\ \ddot{\theta}_y \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1700416 & -1382.24 & 1669.80 & 1.357 \\ 0 & 0 & 0 & 1 \\ 1669.80 & 1.357 & -1700416 & -1382.24 \end{bmatrix} \begin{bmatrix} \theta_x \\ \dot{\theta}_x \\ \theta_y \\ \dot{\theta}_y \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 1700416 & 0 \\ 0 & 0 \\ 0 & 1700416 \end{bmatrix} \begin{bmatrix} V_y \\ V_x \end{bmatrix} \quad (26)$$

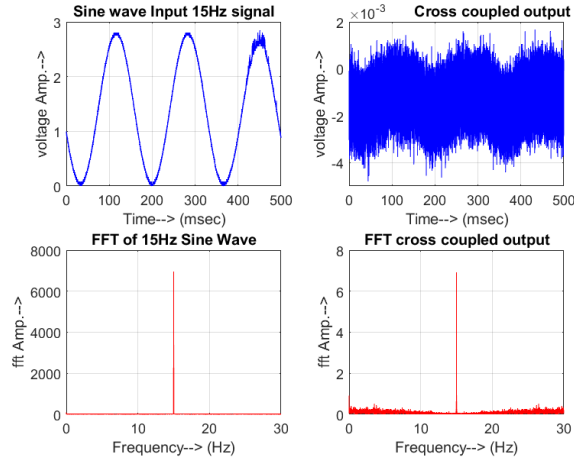


Figure 4.6 Cross-Coupling at 15 Hz

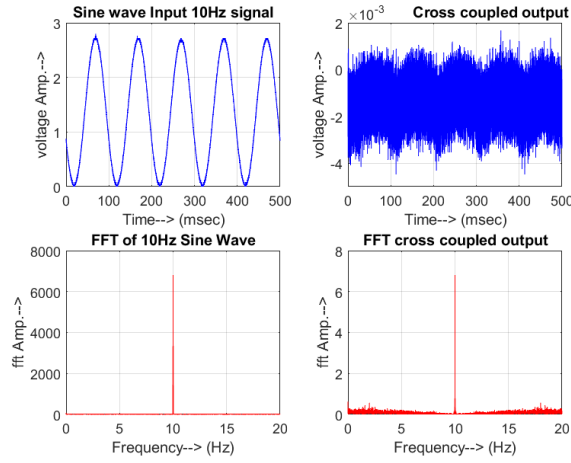


Figure 4.7 Cross-Coupling at 10 Hz

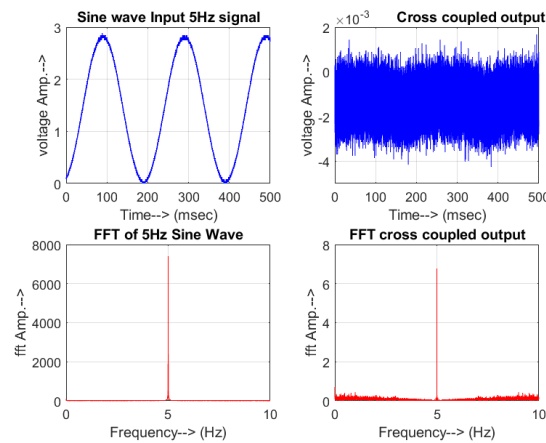


Figure 4.8 Cross-Coupling at 5 Hz

This study addresses the issue of cross-coupling between the two axes of the FSM, incorporating it into the FSM system's state space model as indicated by (26).



## 4.5 Model comparison and validation

Table 6 gives a comparative list of the parameter estimates in line with methods given in [109] and [110] and the TR method proposed in this work. However, the proposed TR method has a limitation that it cannot be applied for modelling of FSM systems with mild oscillation for which the proposed algorithm for the FR method is implemented.

**TABLE 4.5 Comparison list of parameter estimates**

Methods [109]	$w_n(\text{rad/s})$	$\xi$	$t_d(\text{ms})$
2-pts method	1373	0.58	-
3-pts method	1310	0.56	-
Methods [110]			
2-pts method	1388	0.53	0.50
3-pts method	1355	0.53	0.48
Proposed method (TR)	1304	0.53	0.57

The parameter estimation results obtained by the TR method is comparable to two established methods [109],[110] as is evident from Table 6.

### 4.5.1 Parameter estimation-based FR method for SOPDT system with strong damping

**i. e.  $\xi > 0.7$  [111]:**

To check the efficacy of FR method, the FR scheme was applied to an FSM system of [111] which has mild oscillations/strong damping. From the open loop frequency response data, the cut-off frequency  $\omega_c$  of the FSM was found to be 4211 rad/sec (or 670 Hz). Solving the (12-14), by selecting ten frequencies in the interval (0,4211], the values of  $u, x, z$  are obtained as:

$$u = 3.87 \times 10^{-8}, x = 9.77 \times 10^{-16}, z = 1.00(53)$$

Using (15) we get

$$K = 0.999, \omega_n = 5655 \text{ rad/sec}, \xi = 0.9, t_d = 67 \times 10^{-6} \text{ s} \quad (27)$$

The model of [111] so obtained by FR method is

$$G_{\text{FSM}}(s) = (e^{-0.0000006s}) \left( \frac{5655^2}{s^2 + 10179s + 5655^2} \right) \quad (28)$$

which closely matches with that obtained in [111] ( $K = 1, \omega_n = 5655 \text{ rad/sec}, \xi = 0.9, t_d = 67 \times 10^{-6} \text{ s}$ )

#### 4.5.2 Comparison of TR and FR Method with Subspace Identification Method [63]

In order to check the performance of TR and FR model, the performance index viz. variance-accounted-for (VAF) is calculated for proposed methods, and compared with the subspace identification method. The VAF signifies the consistency between the model and actual system as stated in [63]. So, a VAF of 100% ensures an exact match of the estimated system with the actual system. To check the accuracy of the model the VAF value of the recovered model is calculated as:

$$VAF = \left[ 1 - \frac{\text{var}(y_{\text{actual}} - y_{\text{model}})}{\text{var}(y_{\text{actual}})} \right] \times 100\% \quad \text{where}$$

$y_{\text{actual}}$  = output from the actual system for a given input

$y_{\text{model}}$  = output from the model for the same input.

$\text{var}$  = variance

The open loop frequency response data of the FSM for a sinusoidal input excitation from 1Hz to 1000Hz is taken as  $y_{\text{actual}}$ . The  $y_{\text{model}}$  is the simulated frequency response output for both TR and FR models for the same input. Using the MATLAB subspace identification toolbox, the  $y_{\text{model}}$  for the subspace identification method is evaluated. The VAF for the three methods tabulated in Table 4.6. clearly shows an improvement in the accuracy of the TR and FR methods over the conventional subspace method.

**TABLE 4.6 VAF Evaluation**

		Magnitude	Phase
		VAF %	VAF%
TR method		98.3	98.8
FR method		96.9	98.3
Subspace Identification method	System	Magnitude	Phase
	Order	VAF%	VAF%
	1	65	14.0
	2	95.4	56.1
	3	95.1	93.0

The use of variance accounted for (VAF) as a validation metric, along with comparisons to conventional methods, emphasizes the robustness of the proposed model as indicated in Table 4.6.

## **4.6 Conclusion**

This research introduces an algorithm for effectively modelling a two-axis FSM assembly in both the time and frequency domains. Both the TR and FR schemes demonstrate high efficacy, as evidenced by the VAF estimates exceeding 95% for both cases, ensuring the algorithm's accuracy and reliability. The accurate estimation of the cross-coupling coefficient highlights the method's success in capturing complex interactions between the assembly's axes. Additionally, the effective transformation of time response signals using the Fast Fourier Transform (FFT) showcases the method's proficiency in analysing system dynamics. Moreover, the successful application of the open-loop algorithm without specific controller insights underscores the method's versatility and potential for broader application in modelling similar complex systems. These findings collectively suggest the method's efficacy in enhancing the understanding and control of dual-axis tip-tilt fast steering mirror assemblies. The implications of this research pave the way for the future incorporation of the coupled black box model in the design of a controller, thereby enhancing the potential applicability of the algorithm in practical setting.

## **Chapter 5**

---

### **Fuzzy Logic Control**

## 5.1 Introduction

In the realm of control systems, the quest for effective and adaptive methodologies has led to the development of sophisticated techniques capable of handling complex and dynamic environments. Fuzzy Logic Control (FLC) stands out as a powerful paradigm that embraces the inherent uncertainty and imprecision present in many real-world systems. Unlike traditional control methods, which rely on precise mathematical models, FLC mimics human decision-making processes, allowing for flexible and robust control in situations where precise modeling may be challenging. The core strength of FLC lies in its capacity to handle non-linear and time-varying systems, making it an attractive choice for a wide range of industries, including manufacturing, automotive, robotics, and more. As technology advances and systems become increasingly intricate, the demand for control strategies that can adapt to uncertain and changing conditions has grown. Fuzzy Logic Control, with its ability to model and respond to uncertainty, has emerged as a valuable tool in addressing these challenges, offering a bridge between conventional rule-based systems and the complexities of real-world dynamics.

This chapter is organized as follows: The Chapter unfolds with a detailed examination of Fuzzy Logic Controller (FLC) in Section 5.2, shedding light on the essential design considerations and parameters crucial for configuring the controller. Fuzzy-logic algorithm and formulation of the rule-base is presented in Section 5.3. Following this theoretical groundwork, Section 5.4 takes a practical turn by implementing FLC in the MATLAB-Simulink environment. This section further conducts a comparative study between conventional PI-FLC, self-tuned PI-FLC and ZN-PI controllers, offering insights into the relative performance of these controllers. Moving beyond the controller's intrinsic characteristics, Section 5.5 explores the impact of measurement noise on the controllers. This investigation aims to assess the robustness of FLC and its counterparts when faced with uncertainties or disturbances in the measurement process. The Chapter culminates in Section 5.6 with concluding remarks that encapsulate the key findings and implications derived from the study. This section serves as a synthesis of the theoretical and practical aspects discussed earlier, providing a comprehensive overview of the Fuzzy Logic Controller's application, performance, and potential challenges.

## 5.2 Design of Fuzzy Logic Controller for SOPDT system

A fuzzy logic controller offers an attractive option in comparison to conventional controllers when dealing with systems that exhibit certain general characteristics, yet possess overly complex system models that prove challenging to handle through traditional methods. The fundamental characteristic of a fuzzy logic controller lies in its ability to grasp the inherent traits of the system through observation and intuitive understanding.

In systems with dead time, the control action at the present moment relies on data associated with previous instances. Consequently, any time delay in transmission could generate inaccurate data, if directly utilized for analyzing the control action. When developing a fuzzy logic controller to govern a dynamic system, several crucial elements require specification, including:

- i) The input variables and the output variables of the FLC should have a universe of discourse and be partitioned with linguistic labels.
- ii) Membership functions must be defined on each linguistic variable.
- iii) A fuzzification mapping must be specified to relate crisp input variables to fuzzy linguistic labels.
- iv) A defuzzification mapping must be specified to relate fuzzy linguistic labels to crisp output.
- v) Rules must be developed to capture the control strategy by relating the input and output of the FLC.

The block diagram of FLC is shown in Figure 5.1

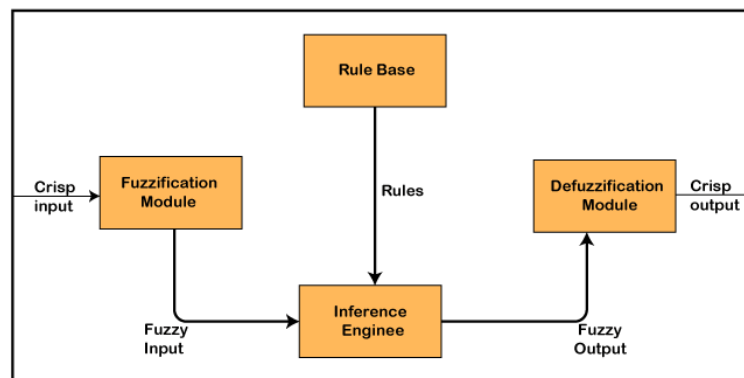


Figure 5.1 Block diagram of FLC

### 5.2.1. Rule Base

Rule Base is a component used for storing the set of rules and the If-Then conditions given by the experts are used for controlling the decision-making systems. Fuzzy if-then rules or fuzzy conditional statements are expressions of the form:

If x is A then y is B.

where x and y are input and output variables. A and B are linguistic labels of the fuzzy sets characterized by appropriate membership functions.

A is the antecedent and B is the consequent parts of the fuzzy rule. Fuzzy values A and B are described by the membership functions. The forms of membership functions are different and the problem dependent. The steps of fuzzy reasoning (inference operations upon fuzzy IF-THEN rules) performed by FISs are described as follows:

- *Compare the input variables with the membership functions on the antecedent part to obtain the membership values of each linguistic label (this step is often called fuzzification).*
- *Combine (usually OR or max.) the membership values on the antecedent part to get the firing strength (weight) of each rule.*
- *Generate the qualified consequents (either fuzzy or crisp) of each rule depending on the firing strength.*
- *Aggregate the qualified consequents to produce a crisp output (This step is called defuzzification).*

### **5.2.2. Fuzzification**

Fuzzification is a module or component for transforming the system inputs, i.e., it converts the crisp number into fuzzy steps. The crisp numbers are those inputs that are measured by the sensors and then fuzzification is passed into the control systems for further processing. This component divides the input signals into the following five states in any Fuzzy Logic system:

- Large Positive (LP)
- Medium Positive (MP)
- Small (S)
- Medium Negative (MN)
- Large negative (LN)

### **5.2.3. Inference Engine**

The Inference Engine serves as a crucial component in any Fuzzy Logic System (FLS) as it handles all the information processing. It facilitates the identification of the degree of matching between the current fuzzy input and the predefined rules. Once the matching degree is established, the system can determine which specific rule is applicable based on the provided input information. As all the relevant rules are triggered, they are combined to formulate the necessary control actions, thereby enabling the system to make informed decisions and execute appropriate actions based on the processed information.

### **5.2.4. Defuzzification**

Defuzzification plays a crucial role within a fuzzy logic system by transforming the fuzzy set inputs produced by the Inference Engine into a distinct, well-defined value. This step serves as the ultimate phase in the operation of a fuzzy logic system. The output generated is a precise value that can be easily comprehended and employed by the user. Different techniques are available to execute this conversion process, but it is vital for the user to select the most appropriate method that effectively reduces errors, guaranteeing the precision of the system's results.

### **5.2.5. Membership functions and linguistic variables**

Membership function (MF) - A function that represents the graph of fuzzy sets that allows users to quantify the linguistic term.

Degree of membership- The output of a membership function, this value is always limited to between 0 and 1. Also known as a membership value or membership grade.

A membership function for a fuzzy set A on the universe of discourse X is defined as

$\mu_A: X \rightarrow [0,1]$ . x axis represents the universe of discourse. y-axis represents the degrees of membership in the [0, 1] interval.

Linguistic variables represent crisp information in a form and precision appropriate for the problem. Each linguistic term covers a relatively wide range of numerical values. Its value is not a number but a word.

### **5.2.6 Models of Fuzzy Logic Controller**



- *Fuzzy PI controller (commonly used)- The conventional FPIC is described by the equation*

$$u(k + 1) = u(k) + \Delta u(k)$$

where  $k$  is the sampling instance and  $\Delta u(k)$  is the incremental change in controller output, determined by fuzzy rules of the form

“If  $e$  is  $E$  and  $\Delta e$  is  $\Delta E$  then  $\Delta u$  is  $\Delta U$ ”

Where  $e$  and  $\Delta e$  are error and incremental change in error signal respectively.

- *Fuzzy PD controller (not recommendable) - The fuzzy PD controller (FPDC), on the other hand uses rules of the form:*

“If  $e$  is  $E$  and  $\Delta e$  is  $\Delta E$  then  $u$  is  $U$ ”.

- *Fuzzy PID controller are rarely used due to the difficulties associated with the formulation of a comparatively larger rule-base and its tuning of more parameters.*

### 5.3 The Fuzzy Logic Algorithm

1) Initialization process: Define the linguistic variables.

Construct the fuzzy logic membership functions that define the meaning or values of the input and output terms used in the rules.

Construct the rule base (Break down the control problem into a series of IF X AND Y, THEN Z rules based on the fuzzy logic rules).

2) Convert crisp input data to fuzzy values using the membership functions (fuzzification).

3) Evaluate the rules in the rule base (inference).

4) Combine the results of each rule (inference).

5) Convert the output data to non-fuzzy values (defuzzification)

#### 5.3.1 Detailed Design Considerations

In this work PI type FLC is implemented, which is shown in Figure 5.2.

The input scaling factors (ISFs) normalize the real-world inputs to a range in which membership functions are defined. The output scaling factor (OSF) is used to change the normalized control effort to its practical value. The relation between real and normalized values of the parameters can be simply given as

$$E = e k_e; \Delta E = \Delta e k_{de}; \Delta u = \Delta U k_o$$

where  $E$  and  $\Delta E$  are the normalized inputs of the FLC controller,  $\Delta U$  is the normalized FLC output;

$e$ ,  $\Delta e$  are actual inputs to the FLC and  $\Delta u$  is actual output.

$k_e$ ,  $k_{de}$ ,  $k_o$  are the error scaling factor, the change of error scaling factor and the control effort change scaling factor, respectively.

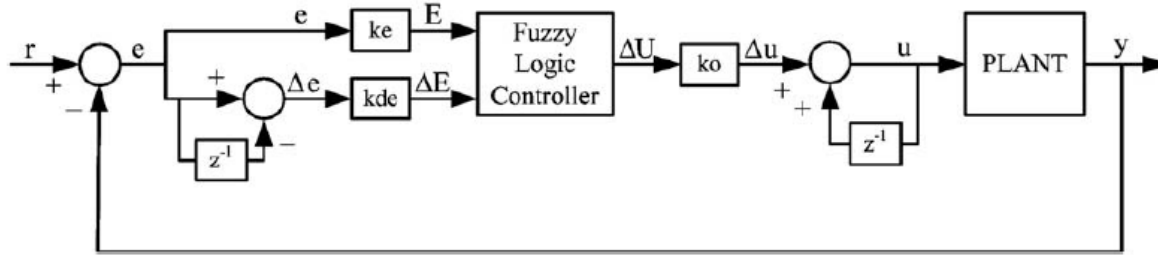


Figure 5.2 The basic control block diagram illustration with PI type FLC

Membership function for inputs (error  $e$ , change of error  $\Delta e$ ) and output ( $\Delta u$ ) are created. The input variables are decomposed into at least seven fuzzy linguistic levels (NL, NM, NS, ZE, PS, PM, PL) as shown in Figure 5.3 in order to make a considerable distinction between the fuzzy regions and, thus, to obtain fine-tuned control action. The universe of discourse is chosen to be  $[-1, 1]$  for the membership functions of input and output variables. The input and output parameters are scaled to fit this range via scaling factors. We use symmetric triangles with equal bases.

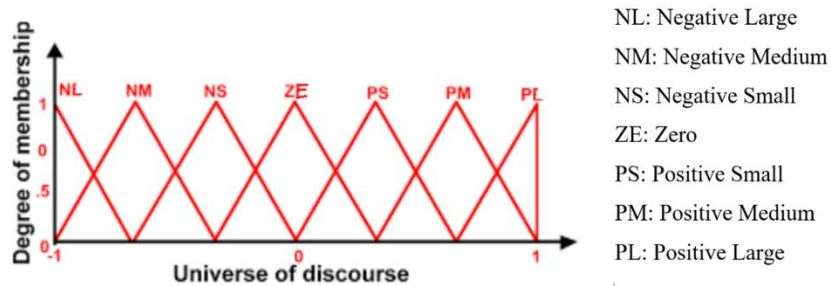


Figure 5.3 Membership function with linguistic labels in FLC

The rule base, as illustrated in Table 5.1, outlines the set of guidelines or conditions governing the system's behavior. This table likely contains a series of input-output relationships, indicating how the system responds or produces output based on different input conditions.

Meanwhile, Figure 5.4 visually represents how the system's output varies in response to changes in its inputs. It provides a graphical depiction of the relationships captured in the rule base, allowing for a more intuitive understanding of how the system behaves under different scenarios.

**Table 5.1 Fuzzy Rules for Computation of  $\Delta U$**

delE/E	NL	NM	NS	ZE	PS	PM	PL
NL	NL	NL	NL	NM	NS	NS	ZE
NM	NL	NM	NM	NM	NS	ZE	PS
NS	NL	NM	NS	NS	ZE	PS	PM
ZE	NL	NM	NS	ZE	PS	PM	PL
PS	NM	NS	ZE	PS	PS	PM	PL
PM	NS	ZE	PS	PM	PM	PM	PL
PL	ZE	PS	PS	PM	PL	PL	PL

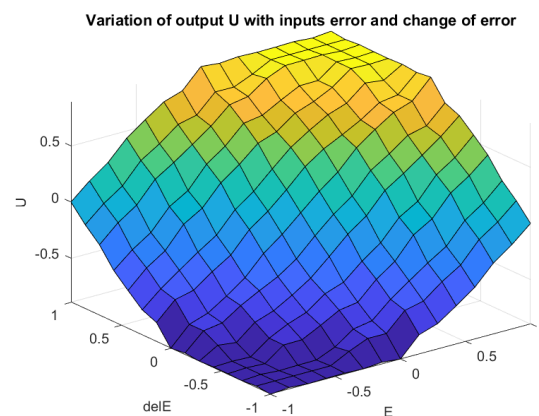


Figure 5.4 Variation of output with inputs

## Formulation of the Rule Base

The rule base for output  $\Delta u$  is formulated based on expert knowledge of the control system. Figure 5.5a shows the step response of an underdamped SOPDT system to be controlled and Figure 5.5b shows the corresponding rule-base. The error  $e$  is the difference of the desired and actual response as given in (1)

$$e = \text{desired response} - \text{actual response (at a specific time)} \quad (1)$$

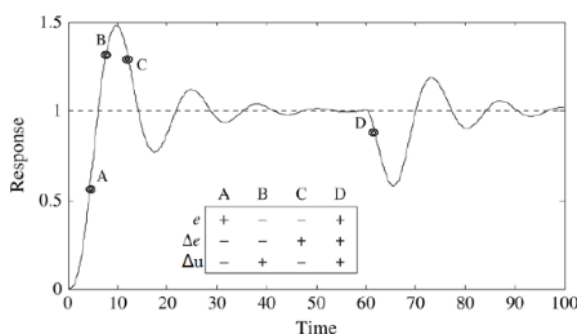
$$\Delta e = \text{present error} - \text{last error (at a specific time)} \quad (2)$$

When  $e$  and  $\Delta e$  are of opposite-sign then the response is heading in the right direction as in points A and C and small control action is required to make the response approach the set point.

Observe that at pt A the response tends to reach the set point, the error is small positive and the change of error ( $\Delta e$ ) is negative; a small control action is required to bring the response reach the set point. This has been realized by rules of the form “If  $e$  is PS and  $\Delta e$  is NM then  $\Delta u$  is NS”

When  $e$  and  $\Delta e$  are of the same sign then the response is heading in the wrong direction as in points B and D and large control action is required to make the response approach the set point.

Observe that at point B the error is large, and the response is not only far away from the set point, but also moving farther away from it, (here both  $e$  and  $\Delta e$  are of the same sign) thus the control action should be large to prevent the situation from becoming worse. This has been realized by rules of the form “If  $e$  is NB and  $\Delta e$  is NS then  $\Delta u$  is NB”



	Nbhd. B						
$\Delta e / e$	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NS	ZE
NM	NB	NM	NM	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PM	PM	PB
PB	ZE	PS	PS	PM	PB	PB	PB
	Nbhd. A						
	Nbhd. C						
	Nbhd. D						

Figure 5.5a Step response of a SOPDT system

Figure 5.5b Rule base to control the tuning

Thus, w.r.t Figure 5.5b the following remarks can be made:

Zone Orange- In this zone error has a tendency to increase further, so a large control action is required. It affects rise time and overshoot/undershoot of the system

Zone Green- In this zone error has a tendency to reduce further, so a small control action is required. It affects the rise time of the system.

Zone Pink- Both error and change of error are small, so small control action is required. It affects the stability of the system.

### 5.3.2 Tuning Parameters -Scaling Factors (SF), membership functions (MF), and rules [91]

The parameters of PI-type FLCs are tuned in order of their significance; that is, first parameters with a global effect (SF) and then ones with only local effect (MF and rules) and, hence, given the maximum importance to the tuning of SF's.

**Scaling Factor:** The values of the actual inputs are mapped onto  $[-1, 1]$  by the input SFs, and, the controller output is mapped onto the respective actual output or domain by the output SF by trial and error. Tuning the SF affects all the rules of the rule base as shown by the highlighted area in Figure 5.6

$\Delta e / e$	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NS	ZE
NM	NB	NM	NM	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PM	PM	PB
PB	ZE	PS	PS	PM	PB	PB	PB

Figure 5.6 Tuning SF affects all rules

**Membership Function:** Tuning a peak value or width of an MF affects only the rule that uses the label. It shows a marginal improvement in the transient response of a second-order system as shown in Figure 5.7a and Figure 5.7b.

$\Delta e / e$	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NS	ZE
NM	NB	NM	NM	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PM	PM	PB
PB	ZE	PS	PS	PM	PB	PB	PB

Figure 5.7a Tuning peak or width of MF of if-part of variable

$\Delta e / e$	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NS	ZE
NM	NB	NM	NM	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PM	PM	PB
PB	ZE	PS	PS	PM	PB	PB	PB

Figure 5.7b Tuning peak or width of then-part of variable

**Rule:** When the rule (either if part or then part) is tuned then only the rule involved changes as shown by highlighted area in Figure 5.8.

$\Delta e / e$	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NS	ZE
NM	NB	NM	NM	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PM	PM	PB
PB	ZE	PS	PS	PM	PB	PB	PB

Figure 5.8 Tuning a rule

## 5.4 Implementation of PI-FLC in MATLAB & Simulink environment

The plant considered is a SOPDT system whose transfer function is (3)

$$G(s) = Ke^{-t_d s} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right) \quad (3)$$

$K=1$ ,  $t_d=0.08$  sec,  $\omega_n=5$  rad/sec,  $\xi=0.75$

The PI-FLC shown in Figure 4.2 is implemented in MATLAB and Simulink environment. For this, two inputs were chosen for Error as 'E' and change of Error as 'deE' and one output as 'U'.

A triangular membership function was used in the universe of discourse  $[-1,1]$ . Each membership has seven linguistic labels as shown in Figures 5.9, 5.10, and 5.11 for E,  $\Delta E$  and  $\Delta U$  respectively.

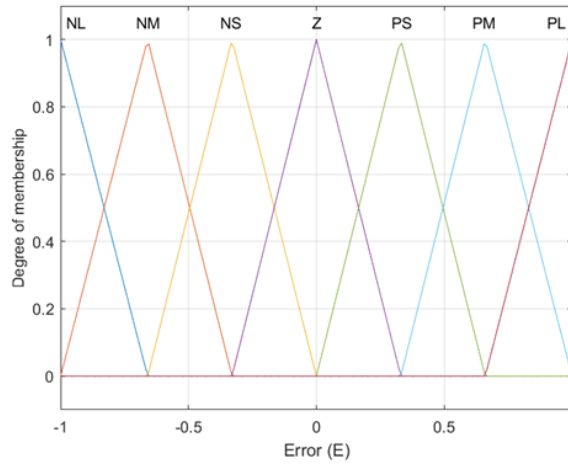


Figure 5.9 Membership function for Error(E)

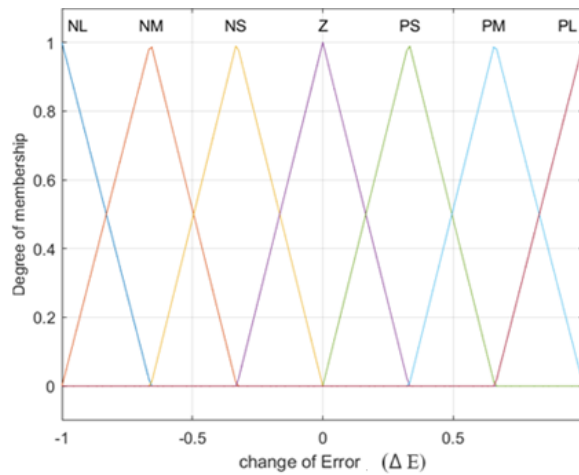
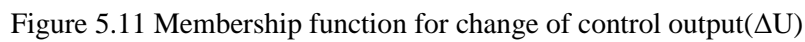
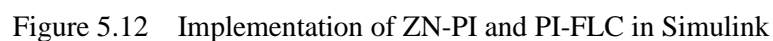


Figure 5.10 Membership function for change of Error( $\Delta E$ )



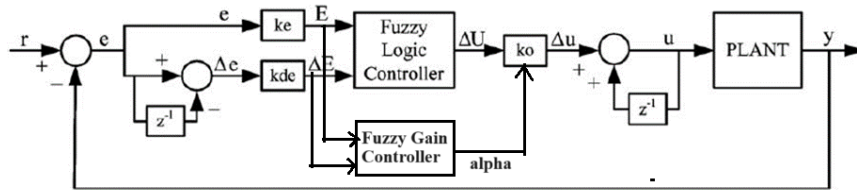
In this Simulink model shown in Figure 5.12, a Ziegler-Nichols PI controller with a proportional gain  $K_p$  of 2.0 and an integral gain  $K_i$  of 2.5 is utilized. This controller is integrated into the PID block of the model. Additionally, a Mamdani fuzzy inference system (Mamfis FIS) with specified membership functions shown in Fig.5.9-5.11 is implemented in the Fuzzy Logic Controller (FLC) block.



105



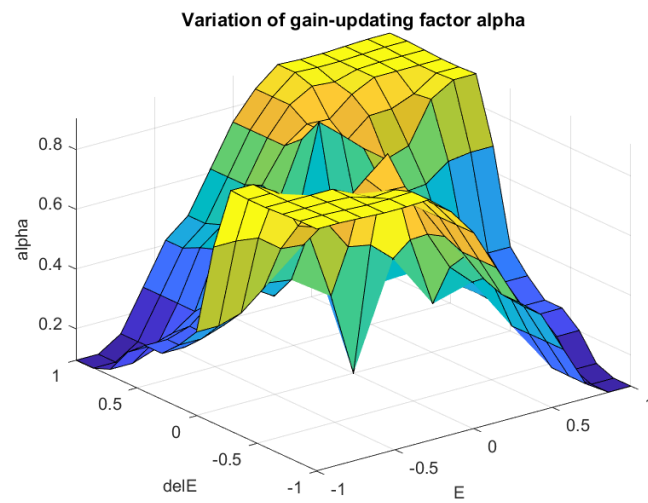
predefined rule base as shown in Figure 5.13b and the corresponding membership function shown in Figure 5.14 [91], allowing dynamic adaptation to changing system conditions. This automation enhances efficiency and responsiveness, particularly in complex systems where manual tuning may be impractical. Figure 5.15 illustrates the implementation of a ZN-PI controller and a self-tuned PI-FLC within the Simulink environment. In Simulink, these controllers are represented using blocks and connections that visually depict the control algorithm.



(a)

$\Delta e / e$	NB	NM	NS	ZE	PS	PM	PB
NB	VB	VB	VB	B	SB	S	ZE
NM	VB	VB	B	B	MB	S	VS
NS	VB	MB	B	VB	VS	S	VS
ZE	S	SB	MB	ZE	MB	SB	S
PS	VS	S	VS	VB	B	MB	VB
PM	VS	S	MB	B	B	VB	VB
PB	ZE	S	SB	B	VB	VB	VB

(b)



(c)

Figure 5.13 (a) Block diagram of self-tuned FLC

(b) Rule base for output scaling factor alpha

(c) Variation of alpha with error and change of error

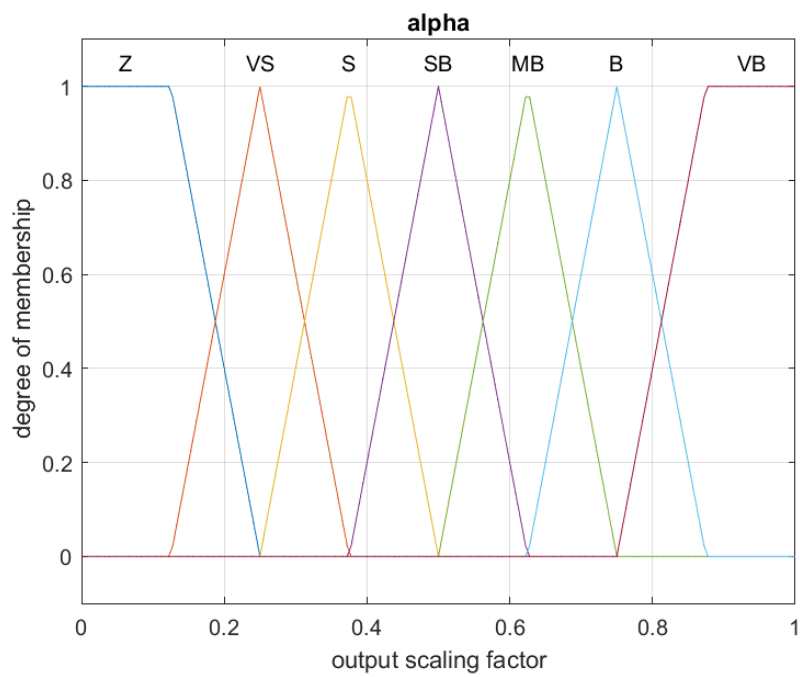


Figure 5.14 Membership function for output scaling factor alpha

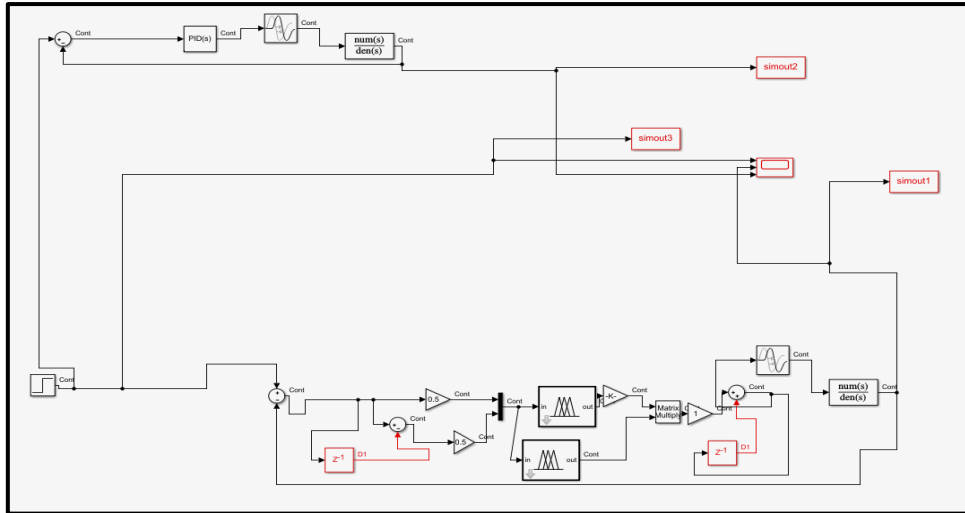


Figure 5.15 Implementation of ZN-PI and self-tuned PI-FLC in Simulink

To assess the performance of the controllers, a closed-loop test is conducted. The comparison is based on various performance indices, including Integral Time Absolute Error (ITAE), Integral Absolute Error (IAE), and Integral Squared Error (ISE). These indices provide insights into how well the controllers are able to regulate the system and minimize errors over time. Table 5.2 presents a comparison of the performance indices (IAE, ITAE, ISE) of the Fuzzy-PI, self-tuned Fuzzy-PI and ZN-tuned PI controller for the SOPDT system (3).

Furthermore, the time domain specifications of both controllers are analyzed and compared. This involves evaluating parameters such as rise time, settling time, and % overshoot. Table 5.3 presents the time-domain performance criteria associated with each of them. These metrics help in understanding the dynamic behavior of the system and how quickly it responds to changes in input. Figure 5.16 provides a visual comparison of the responses generated by the different methods under consideration.

**Table 5.2 Comparison of performance indices for SOPDT system**

Performance Index	ZN-PI	Conventional PI-FLC	Self-tuned PI-FLC
IAE	0.5597	0.8367	0.8311
ITAE	0.4399	0.4583	0.5062
ISE	0.2721	0.6189	0.6129

**Table 5.3 Time domain Performance Criteria**

	ZN-PI	Fuzzy-PI	Self-tuned Fuzzy-PI
Rise time(msec)	187.90	841.42	820.812
%Overshoot	34.45	4.737	2.577
Settling Time (sec)	7.1	5.2	4.8

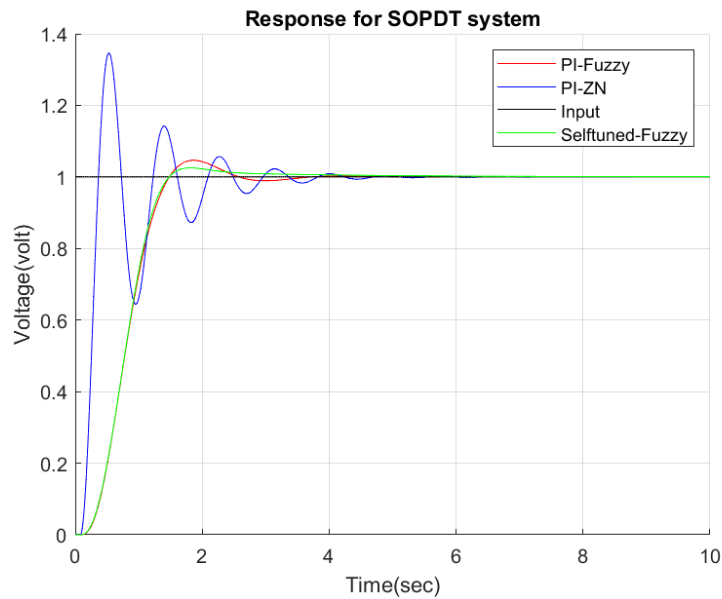


Figure 5.16 Step response by ZN-PI and PI-FLC and self-tuned PI-FLC

In summary, the Simulink model of Figure 5.15 incorporates a Ziegler-Nichols PI controller and a Mamfis FIS controller, and their performance is assessed through closed-loop testing, considering both performance indices and time domain specifications. This comprehensive evaluation provides insights into the effectiveness of each controller in regulating the system under study.

While the Ziegler Nichols PI controller ensures a faster response, it often results in a significant peak overshoot, indicating that the system's output briefly exceeds its desired value before stabilizing as is clearly evident from Figure 5.15. Conversely, a PI (Proportional-Integral) Fuzzy Logic Controller (FLC) is known for its minimal overshoot, although it may exhibit a slower response compared to the ZN-PI controller. Despite its sluggish nature, the PI-FLC's enhanced settling time further contributes to its advantages as is shown in Figure 5.15. Consequently, the FLC approach is generally considered more beneficial, offering improved performance with reduced overshoot and better settling time. The self-tuned FLC enhances the dynamic performance of the control system by improving the rise time, settling time, and peak overshoot of the conventional PI-FLC thereby contributing to a more efficient and precise control of the system.

## **5.5 Comparison of Fuzzy tuned PI control and Ziegler Nichols tuned PI Control for the SOPDT system under measurement noise**

Measurement noise usually corrupts the system output data and hence, the performance of the control algorithm needs evaluation in the presence of measurement noise to check its robustness. A well-crafted controller should possess robust characteristics to effectively handle the inevitable presence of measurement noise. This robustness ensures that the controller can maintain stability and performance even when dealing with uncertainties and variations in the sensor readings. In essence, robust design is a key element in enhancing the controller's ability to navigate real-world conditions marked by inherent measurement uncertainties.

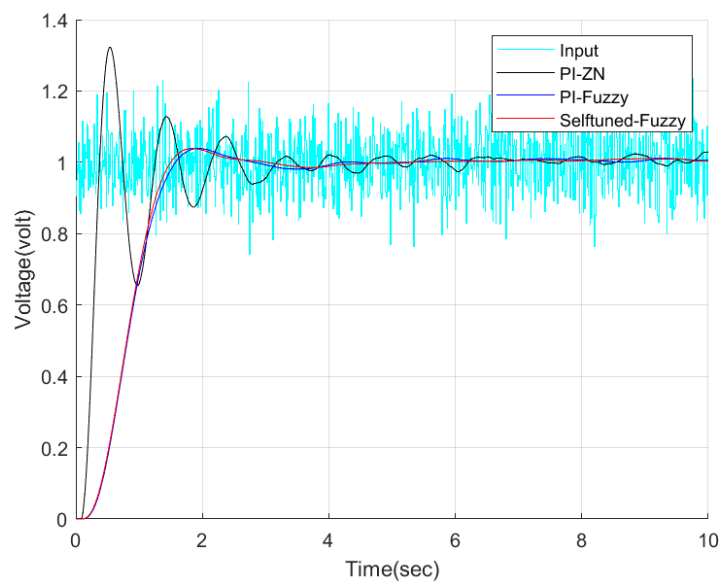
In order to evaluate the performance of both Proportional-Integral Fuzzy Logic controller (PI-FLC) and a Ziegler-Nichols Proportional-Integral (ZN-PI) controller on the Second-Order Plus Dead Time (SOPDT) system (3), a simulation experiment was conducted by incorporating measurement noise into the system's output. The nature of the measurement noise was modeled as White Gaussian, characterized by a zero mean and variance of 0.08 with a sampling time of 3 sec. This corresponds to a specified standard deviation of 28% of the steady-state value. The inclusion of this level of noise introduces variability into the system, simulating real-world conditions where external factors may affect the measurements.

The outcomes of the experiment, detailing the performance of both the PI-FLC and the ZN-PI controller under these noise conditions, are summarized in Table 5.4. Additionally, a

visual representation of the results is provided in Figure 5.17. These tables and figures serve as a comprehensive snapshot of how the controllers responded to the SOPDT system with the introduced measurement noise, offering insights into their robustness and effectiveness in practical, noisy environments.

**Table 5.4 Comparison of performance index for SOPDT system with measurement noise**

Performance Index	ZN-PI	PI-FLC	Self-tuned PI- FLC
IAE	2.987	2.725	2.698
ITAE	13.82	11.84	11.76
ISE	1.387	1.239	1.217



**Figure 5.17 Response of ZN-PI and Fuzzy-PI controller of the SOPDT system under measurement noise**

Under noisy conditions, conventional Proportional-Integral FLC (PI-FLC) and the self-tuned Fuzzy Logic Controller (FLC) demonstrate effective performance maintaining minimal overshoot. On the other hand, the Ziegler-Nichols tuned PI (ZN-PI) controller exhibits significant oscillations, indicating that the tuning parameters chosen based on the Ziegler-Nichols method might not be suitable for the specific dynamics of the system.

### 5.5.1 Comparison of Fuzzy controller performance under external disturbance

In the realm of control systems, disturbances represent external influences that can impact the stability and performance of a system. Understanding how different controllers respond to and mitigate these disturbances is crucial for designing robust and adaptive control strategies. The following block diagram is implemented in Simulink to test the robustness of the different controllers under test. An external periodic disturbance ‘d’ affects the plant

$G(s) = \frac{\omega_n^2}{s(s+2\xi\omega_n)}$  as shown in Figure 5.18.

$$\text{Clearly } T(s) = \frac{\frac{\omega_n^2}{s(s+2\xi\omega_n)}}{1 + \frac{\omega_n^2}{s(s+2\xi\omega_n)}} = \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right)$$

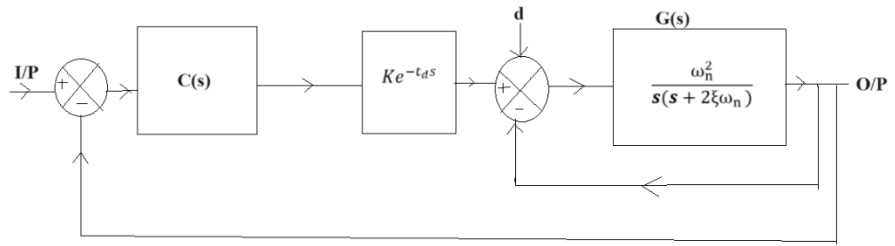


Figure 5.18 Closed Loop System with external disturbance d affecting the plant

The plant under test is the SOPDT system  $Ke^{-t_d s} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right)$ . The robustness of the controller  $C(s)$  is tested under disturbance d which is a pulse train of period 2 sec. and amplitude 0.1V.

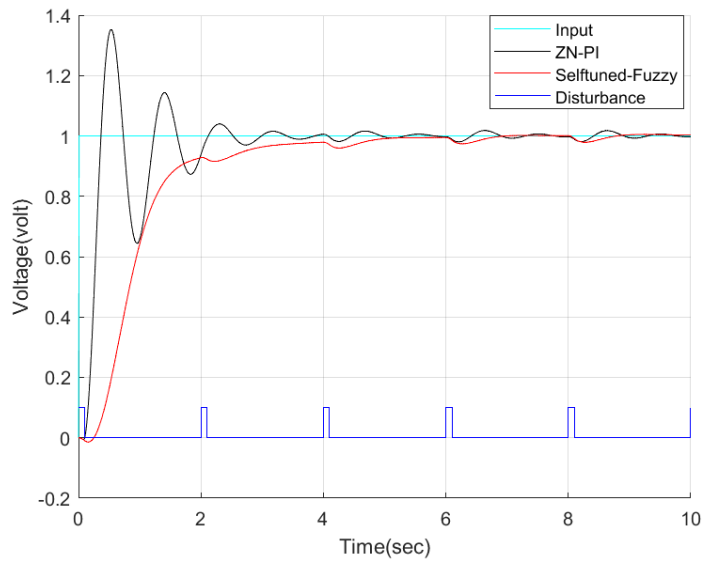


Figure 5.19 Comparison of the responses generated by the different methods (ZN-PI and Self-tuned Fuzzy) under external disturbance

**Table 5.5 Comparison of Time Domain Performance Criteria (self-tuned FLC with ZN-PI) under external disturbance**

	Self-tuned Fuzzy (PI-FLC)	ZN-PI
Rise time(msec)	1.992 sec	188.16 msec
%Overshoot	0	34.45
Settling Time (sec)	8.8	9.2

**Table 5.6 Comparison of performance index for SOPDT system with and without disturbance (self-tuned FLC with ZN-PI)**

Performance Index	ZN-PI (ITAE-value)	Self-tuned FLC (ITAE-value)
With no disturbance	1.861	0.506
With disturbance	1.921	0.596



The results presented in Figure 5.19, Table 5.5 and Table 5.6 provide a comparison of the responses between self-tuned fuzzy PI controller and ZN-PI controller under the effect of external disturbance. The self-tuned fuzzy logic demonstrates a smaller overshoot and a better settling time when compared to ZN-PI controller. It is worth noting that the Ziegler-Nichols tuned proportional-integral (ZN-PI) method yields a high overshoot, rendering it unsuitable for real-time for non-linear systems. Clearly from Table 5.6 the self-tuned FLC exhibits greater resilience to disturbance when compared to conventional ZN-PI.

## **5.6 Summary**

The Ziegler-Nichols PI controller provides a quick response but often results in significant peak overshoot. In contrast, the PI Fuzzy Logic Controller (PI-FLC) has minimal overshoot and improved settling time, despite a potentially slower response. Overall, the FLC approach is preferred for better performance with reduced overshoot. The self-tuned FLC enhances the dynamic control system performance by refining the rise time, settling time, and peak overshoot of the conventional PI-FLC, ensuring more efficient and precise control.

Under measurement noise, the conventional FLC and self-tuned FLC performs better than ZN-PI controller which exhibits large oscillations, suggesting a potential mismatch between the tuning strategy and the system's characteristics. Several factors contribute to their ability of FLC to handle noise effectively:

### **Fuzzification:**

Fuzzification is the process of converting crisp input values into fuzzy sets. This inherently allows FLCs to capture imprecision and variability in the input signals. In the presence of noise, fuzzification provides a means to represent uncertain or imprecise information, preventing the noise from having a drastic impact on the system.

### **Rule-Based Approach:**

FLCs operate on a rule-based system where linguistic rules describe the relationship between input variables and the output. These rules often incorporate human expert knowledge, providing a qualitative understanding of the system. The rule-based nature helps smooth out

the effects of noise, as the controller considers a range of possible scenarios rather than relying on precise values.

#### **Inherent Tolerance to Uncertainty:**

Fuzzy logic is designed to handle uncertainty and imprecision. The linguistic terms used in fuzzy sets, such as "high," "low," or "medium," inherently accommodate variations and uncertainties in the input data, making the controller less sensitive to noise.

#### **Membership Functions:**

The use of membership functions in FLCs allows for a gradual transition between different linguistic terms. This smooth transition helps mitigate the effects of sudden spikes or fluctuations in the input signals, which are typical characteristics of noise.

## **Chapter 6**

---

# **Genetic Algorithm-based PID control**



## **Chapter 6**

---

### **Genetic Algorithm-based PID control**

## 6.1 Introduction

In the dynamic landscape of control systems, the quest for optimal performance has spurred the integration of evolutionary computing techniques, and among these, Genetic Algorithms (GAs) have emerged as a powerful tool for tuning Proportional-Integral-Derivative (PID) controllers. PID controllers, being widely employed for their simplicity and effectiveness, play a crucial role in regulating systems across various industries. However, determining the optimal PID parameters for a given system can be a non-trivial task, especially in the presence of complex and nonlinear dynamics.

Genetic Algorithms, inspired by the mechanisms of natural selection and genetics, provide a novel approach to the PID tuning challenge. GAs operate by mimicking the process of natural evolution, employing principles such as selection, crossover, and mutation to iteratively search for the most suitable set of controller parameters. This approach holds significant promise for systems where the dynamics are poorly understood or subject to variations, as GAs can adapt and optimize the PID parameters in response to changing conditions. The synergy between Genetic Algorithms and PID controllers forms a hybrid intelligent system that combines the simplicity and familiarity of PID control with the adaptability and optimization prowess of genetic algorithms.

This Chapter is organized as follows: In Section 6.2, essential factors related to the Genetic Algorithm (GA) are discussed. Section 6.3 provides a comparison of the control actions between Ziegler-Nichols Proportional-Integral (ZN-PI) and GA-PID controllers. The impact of measurement noise on the control actions is detailed in Section 6.4. Section 6.5 outlines the description of the self-adaptive GA-PID controller. Finally, Section 6.6 offers a summary of the chapter.

## 6.2 Genetic Algorithm: - critical factors

When utilizing Genetic Algorithms (GAs), there are several critical factors that require careful consideration. These are outlined as follows:

### 6.2.1 Choice of fitness function

The choice of an objective function, also known as a fitness function or cost function, is a crucial decision in the design and success of a Genetic Algorithm (GA). Here, the fitness function  $f$  is chosen as the maximum of IAE (Integral of Absolute Error), ITAE (Integral of Time-weighted Absolute Error), and ISE (Integral of Squared Error) as shown in (1)

$$f = \max([IAE, ISE, ITAE]) \quad (1)$$

due to the following reasons:

- a. **Robustness:** Taking the maximum of these three error metrics makes the optimization more robust. It ensures that the solution GA selects performs well with respect to all three error criteria.
- b. **Worst-case scenario:** By selecting the maximum error metric, we are essentially identifying the worst-case scenario. Thus, it ensures that the system performs well under the most adverse conditions.
- c. **Balancing Trade-offs:** Different error metrics have different properties. IAE focuses on bringing the magnitude of the error signal as close to zero as possible, regardless of when the errors occurred. ITAE are time-weighted and emphasize recent errors, while ISE squares the errors, penalizing larger errors more heavily. Choosing the maximum helps strike a balance between these properties, potentially leading to solutions that perform well in various contexts.
- d. **Simplifying Decision-Making:** Using the maximum error metric as the fitness function can simplify decision-making. It provides a single, clear value to compare solutions, making it easier to choose the best-performing individuals in the GA population.
- e. **Problem-Specific Considerations:** In some applications, the choice of which error metric is most important can vary depending on the specific problem and context. By taking the maximum, you give equal weight to all three metrics, avoiding the need to make a subjective decision about their relative importance.

The PID controller is employed to reduce the error signals, or to put it in a more precise context, to minimize the values of the performance index or the fitness function  $f$ . Since smaller values of the performance index indicate better controller performance, we can establish a relationship between chromosome fitness and performance criteria. In other words, the fitness of the chromosomes is defined as in (2)

$$\text{Fitness Value} = \frac{1}{f} \quad (2)$$

In this context, a higher fitness value corresponds to a better chromosome or controller, as it signifies a lower value for the associated performance index. Conversely, lower fitness values indicate chromosomes with poorer controller performance.

### 6.2.2 Encoding

When using Genetic Algorithms (GA) in MATLAB's Genetic Algorithm Toolbox to optimize PID controller parameters, we start by encoding the problem into GA chromosomes and creating an initial population of typically 20 to 100 chromosomes. We use 50 chromosomes for a balance between efficiency and accuracy. These chromosomes represent real-number values for  $K_p$ ,  $K_i$  and  $K_d$  within specific bounds derived from system characteristics, such as delay, and established control methods Ziegler-Nichols rule. Careful choice of bounds is essential for successful convergence, as arbitrary bounds may impede finding optimal solutions. The lower bound chosen for  $[K_p, K_i, K_d]$  are  $[0,0,0]$  and the upper bound as  $[200, 200, 200]$ .

### 6.2.3 Population

Each generation's population is represented as a matrix  $A$  of size  $(50 \times 4)$  as given in (3) where each row corresponds to a chromosome. These chromosomes contain values for proportional, integral, and derivative gains vis  $K_p$ ,  $K_i$ ,  $K_d$  respectively and an additional column to store their fitness values  $f$  as in (4). The GA algorithm iteratively evolves these populations to find the best PID controller parameters for the given problem.

$$A = \begin{bmatrix} K_{p1} & K_{i1} & K_{d1} & f_1 \\ K_{p2} & K_{i2} & K_{d2} & f_2 \\ K_{p3} & K_{i3} & K_{d3} & f_3 \\ K_{p4} & K_{i4} & K_{d4} & f_4 \\ \vdots & \vdots & \vdots & \vdots \\ K_{p50} & K_{i50} & K_{d50} & f_{50} \end{bmatrix} \quad (3)$$

### 6.2.4 Reproduction

Reproduction serves as a fundamental operation within a genetic algorithm, functioning in accordance with the principle of "survival of the fittest." During each generation of the algorithm, chromosomes from the current population are duplicated or copied into the subsequent generation. This copying process is guided by a reproduction probability that is defined as in (4).

$$Probability = \frac{f_j(K_{pj}, K_{ij}, K_{dj})}{\sum_{j=1}^{J=50} f_j(K_{pj}, K_{ij}, K_{dj})} \quad (4)$$

### **6.2.5 Crossover (Recombination):**

Crossover is performed on pairs of parent chromosomes to create offspring. Crossover involves swapping genetic information between parents to produce one or more children. The goal is to combine the favorable characteristics of the parents.

### **6.2.6 Mutation**

Mutation involves making small random changes in the genetic information of a chromosome. It is applied to some of the offspring chromosomes. This introduces genetic diversity into the population and can help explore new regions of the solution space.

### **6.2.7 Replacement**

Replace the old population with the new population of offspring. Some genetic algorithms use generational replacement, where the entire old population is replaced, while others use steady-state replacement, where a portion of the old population is retained

### **6.2.8 Termination/ Solution Extraction**

The termination condition could be a maximum number of generations, a target fitness level, or a time limit. If the termination condition is met, the algorithm stops.

If the algorithm has converged to a satisfactory solution, the best chromosome(s) found in the final population is extracted as the solution to the optimization problem.

### **6.2.9 The Genetic Algorithm**

#### **BEGIN**

Generate Initial Population given as in (3)

Determine Fitness from (2)

#### **REPEAT:**

Perform Crossover

Perform Mutation

Perform Replacement

Compute Fitness



**UNTIL** the population converges to the minimum fitness function in the specified generation.

**END**

### 6.3 Comparison of GA-tuned PID control and Ziegler Nichols-tuned PI Control for SOPDT system

A Genetic Algorithm was implemented for PID tuning in the control of a SOPDT system (5).

$$G(s) = Ke^{-t_d s} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right) \quad (5)$$

$$K=1, t_d=0.08 \text{ sec}, \omega_n=5 \text{ rad/sec}, \xi=0.75$$

The fitness function utilized aims to minimize the worst-case value among three key performance metrics: Integral of Absolute Error (IAE), Integral of Squared Error (ISE), and Integral of Time-weighted Absolute Error (ITAE). The optimization process was conducted with a population size of 50 individuals over a span of 20 generations.

The well known Ziegler Nichols tuning rule was applied to the SOPDT system (5) and it was compared with the GA algorithm stated in Section 6.2.

The study assessed the cost functions ITAE, ISE, and IAE within a closed-loop testing scenario. This evaluation was carried out using a PID controller configuration with equal proportional gain  $K_p$ , integral gain  $K_i$ , and derivative gains  $K_d$ , (such that  $K_p = K_i = K_d=0.5$ ) in Simulink, as illustrated in Figure 6.1. The resulting values of these cost functions, namely ITAE, ISE, and IAE, are presented in the second column of Table 6.1. Among these values, the maximum or most unfavorable outcome belongs to the cost function ITAE. The Genetic Algorithm (GA) was employed separately for each of the cost functions. The objective was to determine the optimal tuning values for  $K_p$ ,  $K_i$ ,  $K_d$ . The results of this optimization process, which corresponds to the minimized cost function, is documented and presented in Table 6.1.

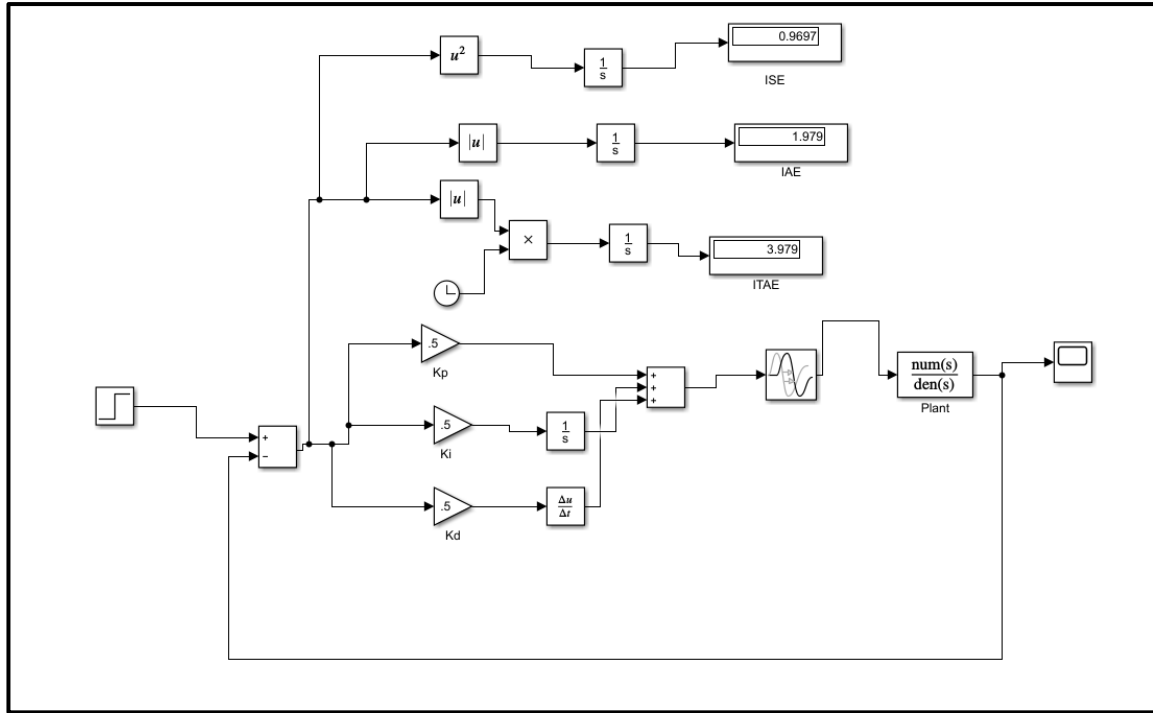


Figure 6.1 Simulation in Simulink for evaluation of cost function when  $K_p = K_i = K_d=0$

**Table 6.1 Optimal tuning vlues of  $K_p, K_i, K_d$  for different cost function**

Cost function	$K_p = K_i = K_d=0.5$ (initial values)	GA-Minimized	Best tuning values for minimized cost function		
			$K_p$	$K_i$	$K_d$
ITAE	3.979	0.0568	0.2564	1.0253	0.0627
IAE	1.979	0.2593	0.4939	2.1161	0.0623
ISE	0.9697	0.5675	0.3594	2.0860	0.0163

Figure 6.2 provides a visual comparison of the responses generated by the different methods under consideration. To further evaluate and compare these methods, Table 6.2 presents the time-domain performance criteria associated with each of them, and Table 6.3 presents a comparison of the performance indices (IAE, ITAE, ISE) of GA tuned PID and ZN tuned PI controller for the SOPDT system (5). Interestingly, even though ITAE represents the worst-case scenario among the evaluated cost functions, it outperforms both ISE and IAE in this context.

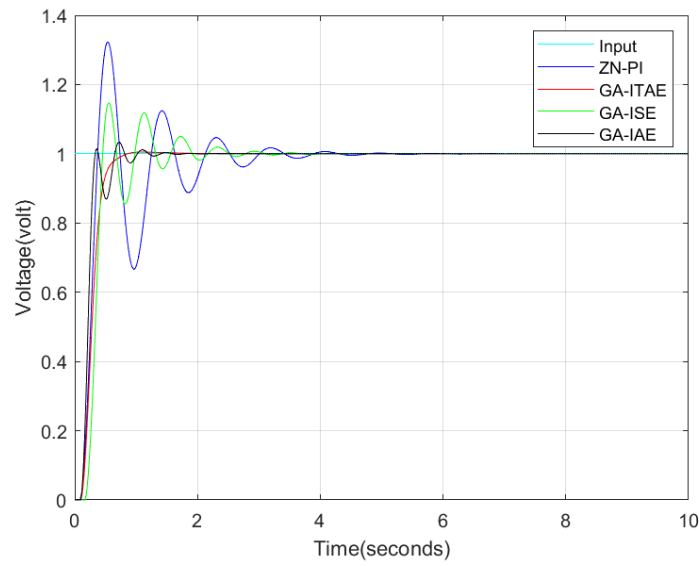


Figure 6.2 Unit step response using Ziegler Nichols and GA with different cost functions (IAE, ITAE and ISE)

**Table 6.2 Time domain Performance Criteria**

	GA-ITAE	GA-IAE	GA-ISE	ZN-PI
Rise time(msec)	299.214	170.017	234.28	187.90
%Overshoot	0.505	11.798	13.08	34.45
Settling Time (sec)	1.4	4.8	5.5	7.1

**Table 6.3 Comparison of performance index for SOPDT system**

Performance Index (Cost function)	ZN-PI	PID tuned by GA
IAE	0.5597	0.2593
ITAE	0.4399	0.0568
ISE	0.2721	0.5675

In the comparison (from Figure 6.1 and Table 6.2) between PID tuning methods, Genetic Algorithms (GA) outperforms Ziegler-Nichols, highlighting GA's effectiveness in optimizing PID controller parameters for improved control system performance.

Moreover the minimizing the worst case cost function ITAE, performs better than ISE and IAE. Ziegler-Nichols is a classical method relying on manual oscillation induction, often providing only decent initial settings, which might not work well for complex or nonlinear systems, assuming linear behavior. In contrast, Genetic Algorithms, inspired by natural selection, evolve potential solutions over generations to find optimal PID parameters. The key to GA's success lies in its fitness function, quantifying system performance and customization to specific system dynamics and objectives. This function ensures efficient, and adaptable PID tuning, making GA a preferred choice for control system optimization.

#### **6.4 Comparison of GA tuned PID control and Ziegler Nichols tuned PI Control for SOPDT system under measurement noise**

Measurement noise usually corrupts the system output data and hence performance of the control algorithm needs evaluation in presence of measurement noise to check its robustness. A well-crafted controller should possess robust characteristics to effectively handle the inevitable presence of measurement noise. This robustness ensures that the controller can maintain stability and performance even when dealing with uncertainties and variations in the sensor readings. In essence, robust design is a key element in enhancing the controller's ability to navigate real-world conditions marked by inherent measurement uncertainties.

In order to evaluate the performance of both a Genetic Algorithm (GA) tuned PID controller and a Ziegler-Nichols Proportional-Integral (ZN-PI) controller on the Second-Order Plus Dead Time (SOPDT) system (5), simulation experiment was conducted by incorporating measurement noise into the system's output. The nature of the measurement noise was modeled as White Gaussian, characterized by a zero mean and Variance of 0.0064. This corresponds to a specified standard deviation of 8% of the steady-state value. The inclusion of this level of noise introduces variability into the system, simulating real-world conditions where external factors may affect the measurements.

The outcomes of the experiment, detailing the performance of both the GA controller and the ZN-PI controller under these noise conditions, are summarized in Table 6.4. Additionally, a visual representation of the results was provided in Figure 6.3, 6.4 and 6.5. These tables and figures serve as a comprehensive snapshot of how the controllers responded to the SOPDT system with the introduced measurement noise, offering insights into their robustness and effectiveness in practical, noisy environments.

**Table 6.4 Comparison of performance index for SOPDT system with measurement noise (mean zero and standard deviation 8% of steady state value)**

Performance Index (Cost function)	ZN-PI	PID tuned by GA	Best tuning values for minimized cost function by GA		
			K <sub>p</sub>	K <sub>I</sub>	K <sub>D</sub>
IAE	0.8008	0.5534	2.0010	4.2900	0.2135
ITAE	1.867	1.776	0.3184	1.3894	2.8750
ISE	0.2876	0.3184	1.3894	2.8750	0.1953

The performance evaluation as shown in Table 6. considered three different cost functions: IAE, ITAE, and ISE and they show superior performance of GA-Control over traditional ZN-PI control in the presence of measurement noise.

Further the comparison of the response graphs of the SOPDT system highlights the effectiveness of GA-tuned PID control over traditional ZN-PI control as is evident from Figures 6.3, 6.4, and 6.5. Among the three evaluated cost functions—IAE, ITAE, and ISE—the ITAE stands out as the most effective criterion for assessing and comparing the system responses. A reduction in ITAE values suggests that the GA controller not only minimized errors but also did so in a more time-efficient manner. This implies improved control response in presence (see Figure 6.3) measurement noise.

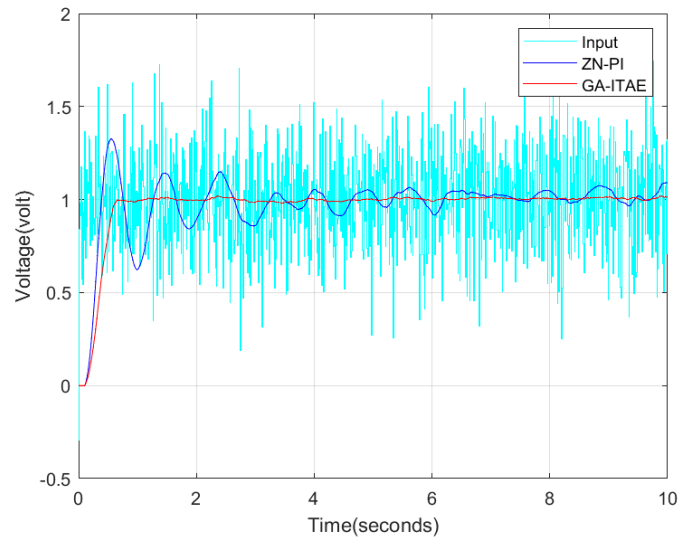


Figure 6.3 Response of ZN-PI and GA-ITAE controller of the SOPDT system under measurement noise.

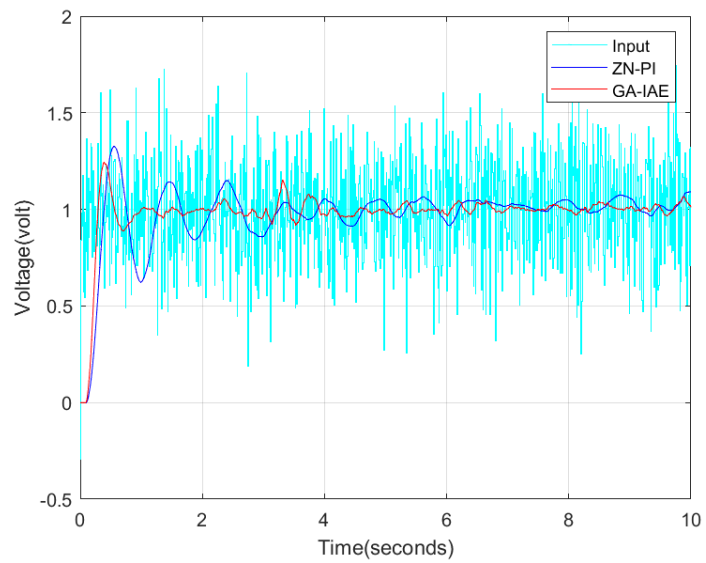


Figure 6.4 Response of ZN-PI and GA-IAE controller of the SOPDT system under measurement noise

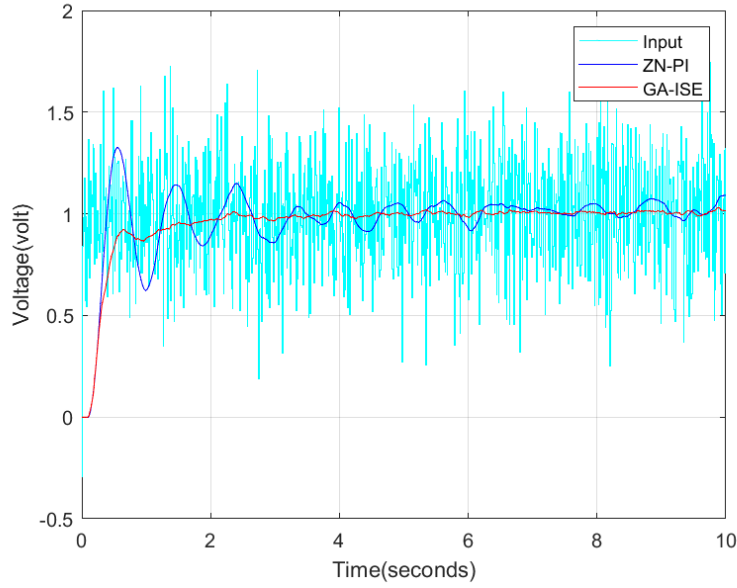


Figure 6.5 Response of ZN-PID and GA-ISE controller of the SOPDT system under measurement noise

Notably, the graphical representation of Figure 6.6 (which shows all the cost-functions) clearly demonstrates the superior performance of GA-tuned PID control over ZN-PID control, with ITAE emerging as the most reliable cost function for this particular system evaluation. The Figure 6.6 shows the responses for the different controllers (all in one) under measurement noise.

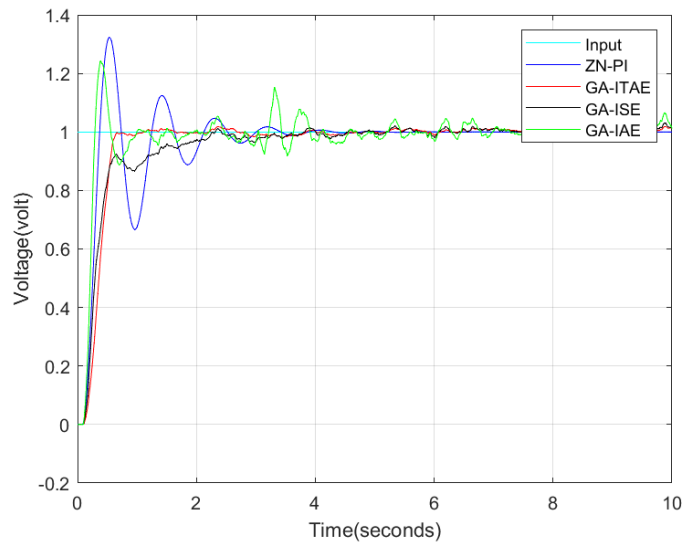


Figure 6.6 Response of ZN-PID and GA controllers (ITAE, IAE and ISE) of the SOPDT system under measurement noise

## 6.5 Self-Adaptive Genetic Algorithm

In the realm of Genetic Algorithms (GA), where optimization processes are employed for tuning parameters like proportional ( $K_p$ ), integral ( $K_i$ ), and derivative ( $K_d$ ) gains in control systems, the selection of an appropriate cost function is crucial. Commonly used cost functions include ITAE (Integral of Time-weighted Absolute Error), IAE (Integral of Absolute Error), and ISE (Integral of Squared Error). Through experimentation across various systems, it has been observed that when the proportional, integral, and derivative gains are held constant, the performance of the GA is most effectively assessed by utilizing the worst-case cost function. Remarkably, in this context, the worst-case cost function consistently yields the best results.

To enhance the adaptability of the GA, a self-adaptive mechanism has been introduced as shown in Figure 6.7. This mechanism dynamically adjusts the optimization process based on the choice of the worst-case cost function, ensuring that the GA optimally tunes the control system parameters for improved performance across diverse scenarios.

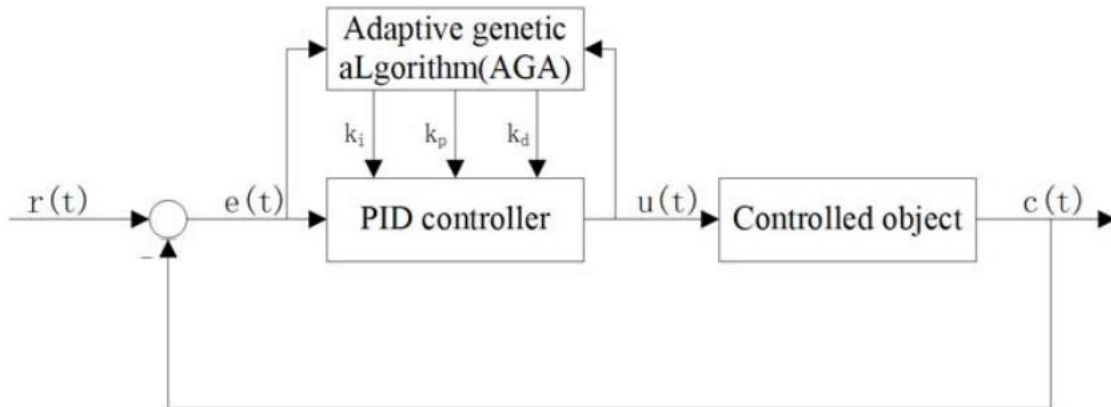


Figure 6.7 Block diagram of Self Adaptive GA controller

### 6.5.1 Choice of the cost function for self-adaptive GA

Evaluation of ITAE, ISE, and IAE within a closed-loop testing scenario was carried out using a PID controller configuration with equal proportional gain  $K_p$ , integral gain  $K_i$ , and derivative gains  $K_d$ , such that  $K_p = K_i = K_d = 0.5$  in Simulink, as illustrated in Figure 6.1. The resulting values of these cost functions, namely ITAE, ISE, and IAE, are presented in the second column of Table 6.1. Among these values, the maximum or most unfavorable outcome belongs to the cost function ITAE. The Genetic Algorithm (GA) was employed separately for each of the cost functions. The objective was to determine



the optimal tuning values for  $K_p$ ,  $K_i$ ,  $K_d$ . The results of this optimization process, which corresponds to the minimized cost function, is documented and presented in Table 6.1.

In the evaluation of closed-loop performance using a PID controller with equal gains ( $K_p = K_i = K_d=0.5$ ), the study focused on key cost functions: Integral of Time-weighted Absolute Error (ITAE), Integral of Squared Error (ISE), and Integral of Absolute Error (IAE). The results of this assessment, depicted in Table 6.1(second column), highlighted ITAE as the most critical metric.

To optimize the PID controller's performance, a Genetic Algorithm (GA) was employed separately for each cost function. The goal was to find the best tuning values for  $K_p, K_i, K_d$  to minimize these cost functions. The optimized results, representing the values of  $K_p, K_i, K_d$  that minimized the cost functions, are summarized in Table 6.1 showing that ITAE cost function gives best results. This table provides a concise overview of the enhanced performance achieved through GA optimization, showcasing the optimal parameter values for an improved PID controller within the specified closed-loop system.

The self-adaptive Genetic Algorithm (GA) is designed to dynamically select the cost function that yields the least favorable outcome when PID gains (proportional, integral, and derivative with values of  $K_p = K_i = K_d=0.5$ ) are kept constant. In this scenario, the chosen cost function is ITAE, as it results in the most unfavorable value. This decision is supported by the Table 6.2 which illustrates that ITAE provides the best time domain performance criteria among the considered cost functions.

### **6.5.2 Comparison of self-adaptive GA controller performance under external disturbance**

In the realm of control systems, disturbances represent external influences that can impact the stability and performance of a system. Understanding how different controllers respond to and mitigate these disturbances is crucial for designing robust and adaptive control strategies. The following block diagram is implemented in Simulink to test the robustness of the different controllers under test. An external disturbance 'd' affects the plant  $G(s) = \frac{\omega_n^2}{s(s+2\xi\omega_n)}$  as shown in Figure 6.8.

Clearly  $T(s) = \frac{\frac{\omega_n^2}{s(s+2\xi\omega_n)}}{1 + \frac{\omega_n^2}{s(s+2\xi\omega_n)}} = \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right)$

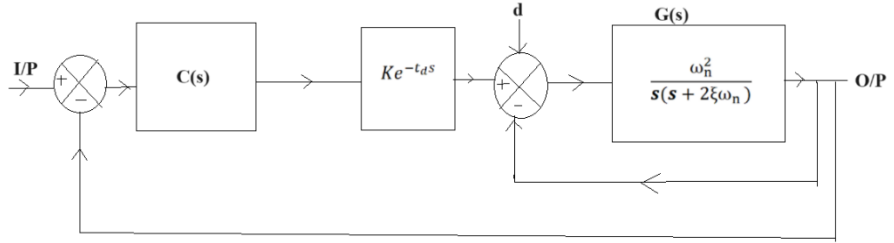


Figure 6.8 Closed Loop System with external disturbance d affecting the plant

The plant under test is the SOPDT system  $Ke^{-t_d s} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right)$ . The robustness of the controller  $C(s)$  is tested under disturbance d which is a pulse train of period 2 sec. and amplitude 0.1V.

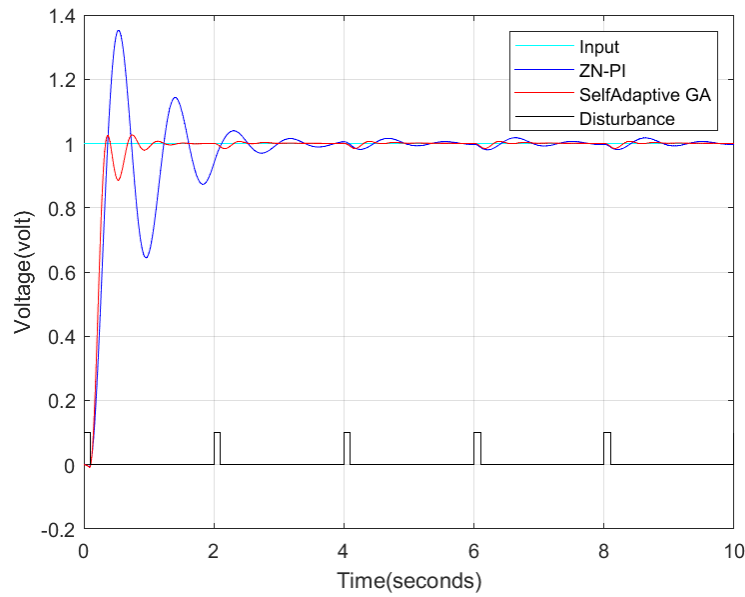


Figure 6.9 Comparison of the responses generated by the different methods( ZN-PI and Self-adaptive GA) under external disturbance

**Table 6.5 Comparison of Time Domain Performance Criteria (self-adaptive GA with ZN-PI) under external disturbance**

	Self-adaptive GA (GA-IAE)	ZN-PI
Rise time(msec)	150.201 msec	188.16 msec
%Overshoot	2.57	34.45
Settling Time (sec)	8.8	9.2

**Table 6.6 Comparison of performance index for SOPDT system with and without disturbance (self-adaptive GA with ZN-PI)**

Performance Index	ZN-PI (ITAE-value)	Self-adaptive GA (ITAE-value)
With no disturbance	1.867	1.776
With disturbance	1.921	1.782

The results presented in Figure 6.9, Table 6.5 and Table 6.6 provide a comparison of the responses between self-adaptive genetic algorithm (GA) tuned PID controller and conventional ZN-PI controller under the effect of external disturbance. The self-adaptive genetic algorithm demonstrates a smaller overshoot and a quicker response when compared to ZN-PI. It is worth noting that the Ziegler-Nichols tuned proportional-integral (ZN-PI) method yields a high overshoot, rendering it unsuitable for real-time for non-linear systems. Clearly from Table 6.6 the self-adaptive GA exhibits greater resilience to disturbance when compared to conventional ZN-PI.

## 6.6 Summary

The PID controller fine-tuned by the Genetic Algorithm demonstrates superior performance criteria for control of the Second-Order Plus Dead Time (SOPDT) system, outperforming the Ziegler Nichols-PI controller. The Ziegler-Nichols method, a classical tuning approach, is thus eclipsed by the advanced optimization capabilities of the Genetic Algorithm when applied to PID

controllers. The GA-tuned PID controller exhibits greater robustness compared to the Ziegler Nichols-PI controller.

The chapter also highlights that the GA-tuned PID controller exhibits greater robustness to measurement noise. Robustness, in the context of control systems, refers to the controller's ability to maintain satisfactory performance despite variations in system parameters or external disturbances. The Genetic Algorithm-tuned PID controller is deemed more resilient and adaptable to changes and uncertainties in the system, highlighting its stability and efficacy under diverse conditions.

In summary, the results emphasize the enhanced performance achieved by employing GA-tuned PID control over ZN-PI control, with ITAE emerging as the most reliable cost function for this particular system evaluation.



## **Chapter 7**

---

### **Modelling and Control of Trainer 37-100 -A Case Study**

## 7.1 Introduction

The Process Control Trainer 37-100 stands as an integrated marvel, encompassing both the plant, representing the system under control, and the control equipment within a singular unit. This case study embarks on a comprehensive exploration of the intricate dynamics involved in the modelling and control of this all-in-one training device. This Chapter delves into the modelling aspect, scrutinizing the dynamic behaviour of the plant within Trainer 37-100. This involves a thorough examination of its response to step inputs, transient behaviours, and the overall system dynamics. Understanding these intricacies lays the foundation for developing accurate models that mirror real-world scenarios, enhancing the effectiveness of the training experience.

Many studies on the Trainer 37-100 have approached its control design by modelling it as a first-order system [118-119], often without verifying the existence of an inflection point. In this case study, we initially identify the inflection point. Subsequently, we introduce a model that represents the system as a second-order plus dead time (SOPDT) overdamped system, which offers superior accuracy compared to the conventional first-order representation.

In parallel, the case study focuses on the control aspects, investigating the design and optimization of control algorithms governing Trainer 37-100. The challenge lies in creating control strategies that adapt seamlessly to the dynamic nature of the plant, ensuring stability, responsiveness, and efficiency in the face of changing conditions.

The rest of this Chapter is organized as follows: in Section 7.2, the focus is on delineating the dynamics of Trainer 37-100, with subsequent sections delving into the system's identification and modeling. Moving on to Section 7.3, a Proportional Integral (PI) fuzzy logic controller (FLC) is developed as part of the study. Following that, in Section 7.4, attention is directed toward the development of a Genetic Algorithm (GA) tuned Proportional-Integral-Derivative (PID) controller. The FLC and the GA-tuned PID are subjected to a comparative analysis against a Ziegler-Nichols (ZN) tuned Proportional-Integral (PI) controller. The evaluation extends further in Section 7.5, where the control actions of the FLC, GA, and ZN controllers are scrutinized under the influence of measurement noise. This investigation provides valuable insights into the robustness and performance of each controller in the presence of real-world challenges, such as measurement noise. By systematically comparing the responses of the FLC, GA, and ZN controllers in Section 7.6, the study aims to contribute

to a nuanced understanding of their respective effectiveness and limitations in practical control scenarios for Trainer 37-100.

## 7.2 Process Trainer 37-100

The Process Control Trainer 37-100 is an all-in-one training device used for process control. It consists of both the plant (the system being controlled) and the control equipment within a single unit. One of the key components in this equipment is a centrifugal blower, which draws air from the surrounding atmosphere and passes it through a heater grid. A bead thermistor is positioned downstream of the heater grid to monitor the air temperature before the air is released back into the atmosphere. The thermistor detects the temperature of the air. The distance between the thermistor and the heater allows for the investigation of "lag" time, which refers to the delay in temperature response. The air velocity within the system can be adjusted using an inlet throttle connected to the blower. Additionally, users can set the desired temperature within a range of 30°C to 60°C. A toggle switch enables an internal step increase to the desired temperature signal, allowing for simulated changes in temperature.

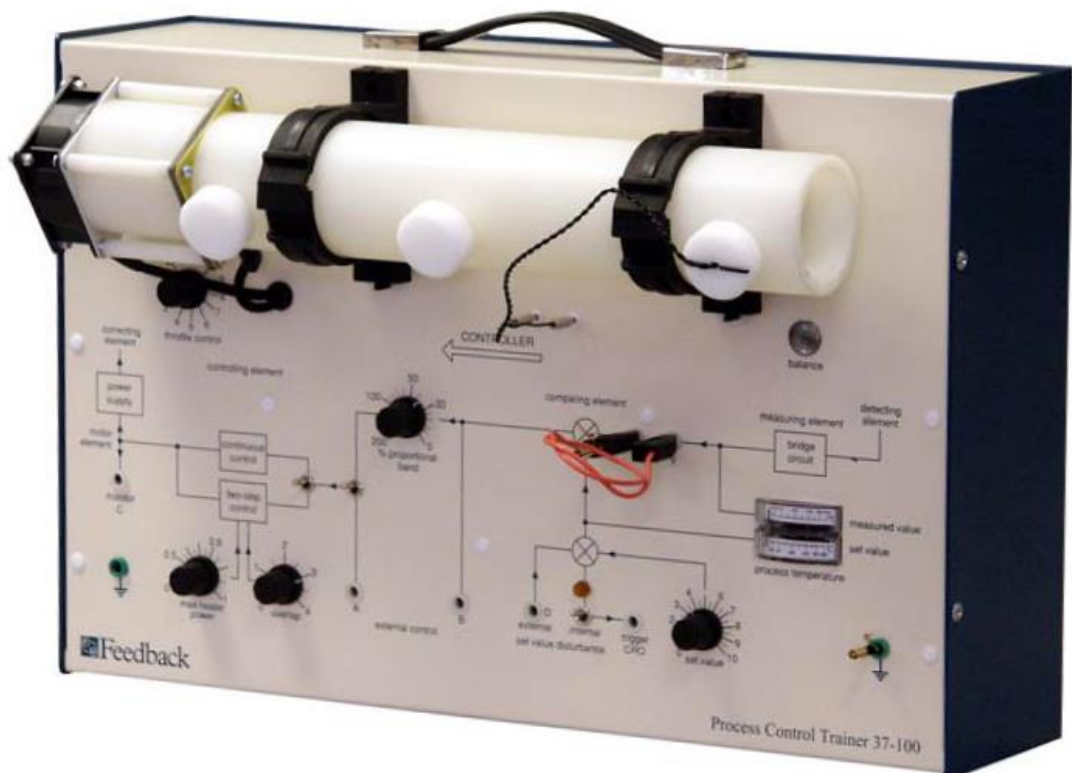


Figure 7.1: Process Trainer 37-100



The Trainer 37-100 offers the flexibility to operate in either open-loop or closed-loop control configurations. In open-loop control, there is no feedback mechanism, while in closed-loop control, the system adjusts itself based on feedback from the temperature measurements. The Trainer has an inbuilt Proportional controller. Proportional control may be varied on the Trainer 37-100 from 5% to 200% corresponding to a gain of 20 to 0.5 respectively where the percent proportional control reflects the gain  $K_P$  applied to the deviation or error signal (difference between desired signal and measured signal) and is given as in (1).

$$K_P = \frac{100}{\text{Proportional}\%} \quad (1)$$

Therefore, a 100% proportional setting provides unit gain to the deviation signal. Furthermore, the trainer allows for the connection of an external controller.

To visualize the overall process, refer to Figure 7.2. It provides a visual representation of how the components of the 37-100 Trainer work together to achieve temperature control in an open-loop configuration. The Trainer 37-100's open-loop transfer function can be located in reference [120], and it is visualized as shown in Figure 7.2.

In Figure 7.2, the product of three factors:  $k_1$ ,  $k_2$ , and  $k_3$ , represents the DC (direct current) gain of the system, symbolized as  $K$ . The DC gain signifies the amplification or attenuation of the input signal by the system in the absence of any time-varying or dynamic effects. The term  $e^{-st_d}$  mentioned in (2) arises due to air transport and is referred to as a transport delay. It represents the time it takes for air or material to be transported through a system, causing a delay in the system's response. This transport delay can affect the overall dynamics and response characteristics of the system. Furthermore, the term  $H(s)$  mentioned in Figure 7.2 refers to the transfer function arising due to the heat transfer dynamics.

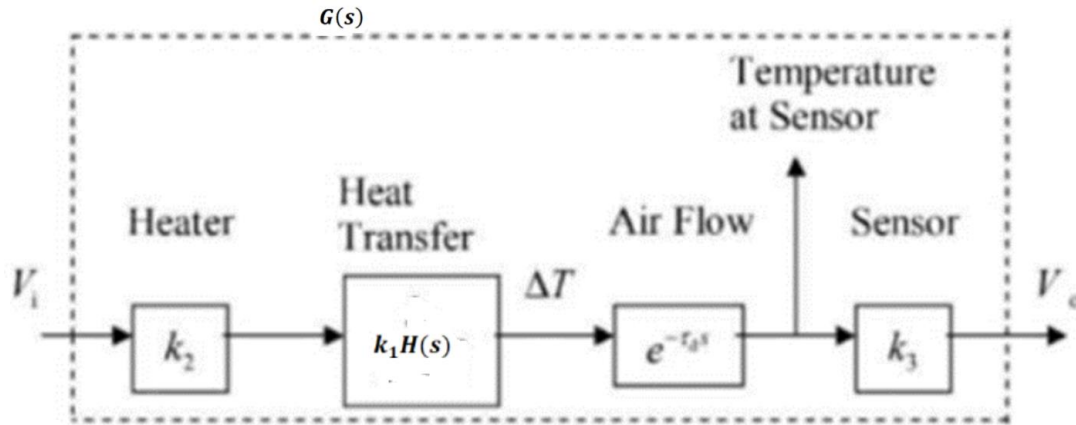


Figure 7.2: Open loop configuration of the process

### 7.2.1 Open loop test

To obtain the process characteristics, open loop tests were performed. A 4V (peak to peak) square wave signal of frequency 0.1 Hz with offset signal -2V was applied externally to the Trainer 37-100 by a signal generator (AFG-2005) and it was observed at Channel 2 of the DSO (GDS-1054B). Keeping the bead thermistor sensor at the last position (279mm), and throttle speed at 3 (i.e 30%), the measured signal was observed at Channel 1 of the DSO (GDS-1054B). Figure 7.3a shows the experimental set-up and Figure 7.3b shows the screenshot of the acquired signals.

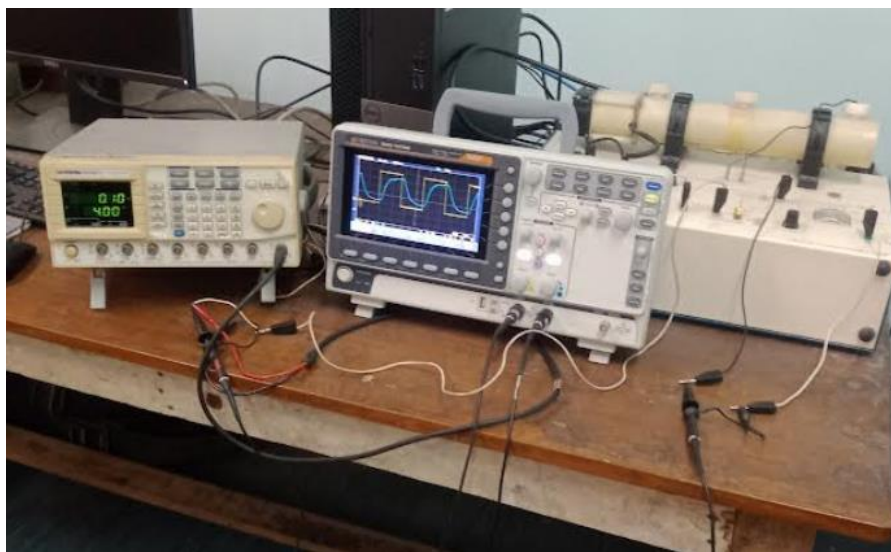


Figure 7.3a: Experimental set-up

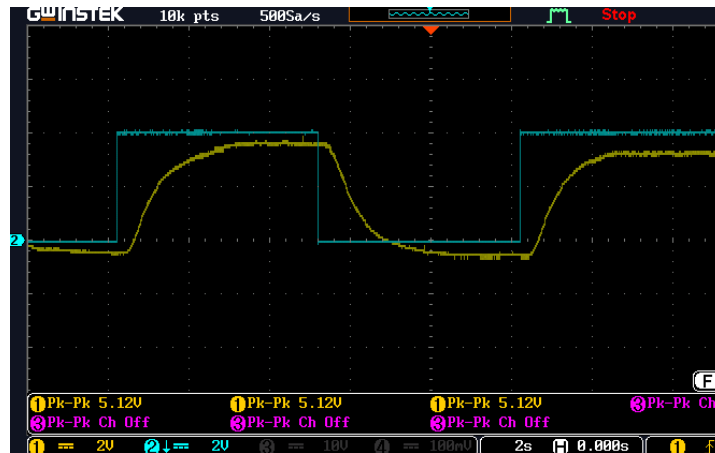


Figure 7.3b: Acquired signal in DSO: Ch1 is the measured signal and Ch2 is the input signal

### 7.2.2 System identification and modelling of Trainer 37-100

The .csv format of the acquired signals is read in MATLAB to determine if the inflection point exists. The identification of inflection points in the unit step response of a system serves as a valuable indicator for understanding its dynamic behaviour. When an inflection point is present, it signifies characteristics akin to a second-order system. In such cases, an overdamped Second-Order Plus Dead Time (SOPDT) approximation is commonly employed. This model, with its negative real poles in the Laplace domain, is particularly suitable for systems displaying slower responses and without overshoot in their step responses. On the contrary, when no inflection point is observed, the system is often simplified and approximated as a first-order system. A first-order system, characterized by a dominant time constant, exhibits a more straightforward response with a single exponential rise or decay and lacks oscillations. These model approximations, whether SOPDT or first-order, play a crucial role in system identification and the design of control strategies, aiding engineers in effectively analyzing and understanding the dynamics of the system at hand.

Mathematically, we analyze the inflection point by calculating the derivative of the step response with respect to time and identifying where the second derivative becomes zero. Alternatively, we may assess the concavity of the curve at different points, recognizing that an inflection point is where the concavity changes. Visual inspection of the graph for a shift from concave up to concave down, or vice versa, can also guide the identification. The inflection point determined is shown in Figure 7.4a.

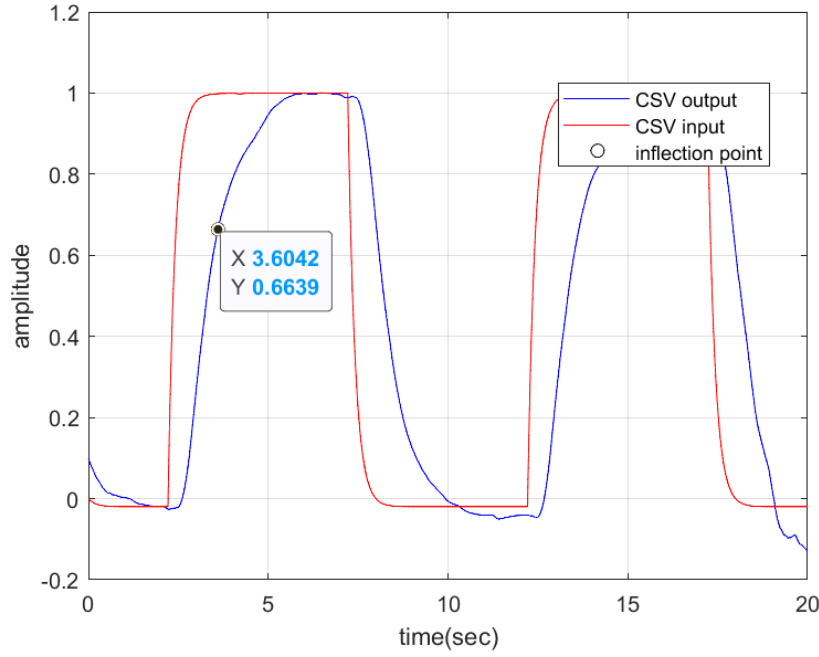


Figure 7.4a: Inflection point identified from acquired output (.CSV file) read in MATLAB

Thus overall transfer function  $G(s)$  of the Trainer 37-100 (shown in Figure 7.2) is a second order plus dead time (SOPDT) system as identified in (2)

$$G(s) = Ke^{-t_d s} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right) \quad (2)$$

(where  $K$  is the DC gain,  $t_d$  is the dead time,  $\omega_n$  is the undamped natural frequency and  $\xi$  is the damping factor of the system).

The sluggish response of the trainer without any oscillation as shown in Figure 7.3 suggests that its dynamics can be characterized as Second-Order Plus Dead Time (SOPDT) with overdamped/critically damped ( $1 \leq \xi \leq 2$ ) behavior. Referring to Chapter 3,  $\xi$ ,  $\omega_n$ ,  $t_d$  and  $K$  may be determined as follows:

$$\xi_{\text{est}} = (2.0489 \times 10^{-3})T_R^2 - (4.9765 \times 10^{-2})T_R + 1.18660 \quad (3)$$

where  $T_R = \frac{t_{90} - t_5}{t_{10} - t_5}$  and  $\xi_{\text{est}}$  is estimated damping ratio.

$$\omega_{n_{\text{est}}} = (4.869\xi - 1.477)f_r \quad (4)$$

where  $f_r = \frac{1}{t_{90} - t_{10}}$  and  $\omega_{n_{\text{est}}}$  is estimated undamped natural frequency.

Here  $t_{90}$  is the time when step response attains 90% of its final value,  $t_{10}$  is the time when step response attains 10% of its final value and  $t_5$  is the time when step response attains 5% of its final value.

These values are tabulated in Table 7.1

With these values of  $\xi_{\text{est}}$  and  $\omega_{n_{\text{est}}}$  we simulate the step response of the system with zero dead time and again find  $t_{90}$  from the simulated time response. The difference of the  $t_{90}$  of the SOPDT system and that of the simulated system will give a measure of the dead time  $t_d$ . The dc gain  $K$  is the ratio of the steady state output to the steady state input. The estimated parameters of the system are tabulated in Table 7.2

**Table 7.1 Estimation of  $T_R$  and  $f_r$**

$t_5(\text{sec})$	$t_{10}(\text{sec})$	$t_{90}(\text{sec})$	$T_R$	$f_r$ (/sec)
2.720	2.790	4.688	25.260	0.5291

**Table 7.2 Estimation of the parameters of the system**

$K$	$\xi_{\text{est}}$	$\omega_{n_{\text{est}}}$	$t_d$
1	1.236	2.405	0.423

Thus, the Trainer model derived is identified as in (5a)

$$G(s) = e^{-0.42s} \left( \frac{5.78}{s^2 + 5.95s + 5.78} \right) \quad (5a)$$

If we overlook the inflection point and represent the Trainer as a first-order model, the resulting model is as follows:

$$G(s) = \frac{e^{-0.42s}}{1.02s + 1} \quad (5b)$$

Figure 7.4b provides a visual comparison of the step response between the FOPDT (first order with dead time system) and the overdamped SOPDT system, juxtaposed with the original response. It's evident from the visual representation that the overdamped SOPDT model offers a more accurate depiction. This observation is further supported by the ISE and IAE assessments presented subsequently in Table 7.3

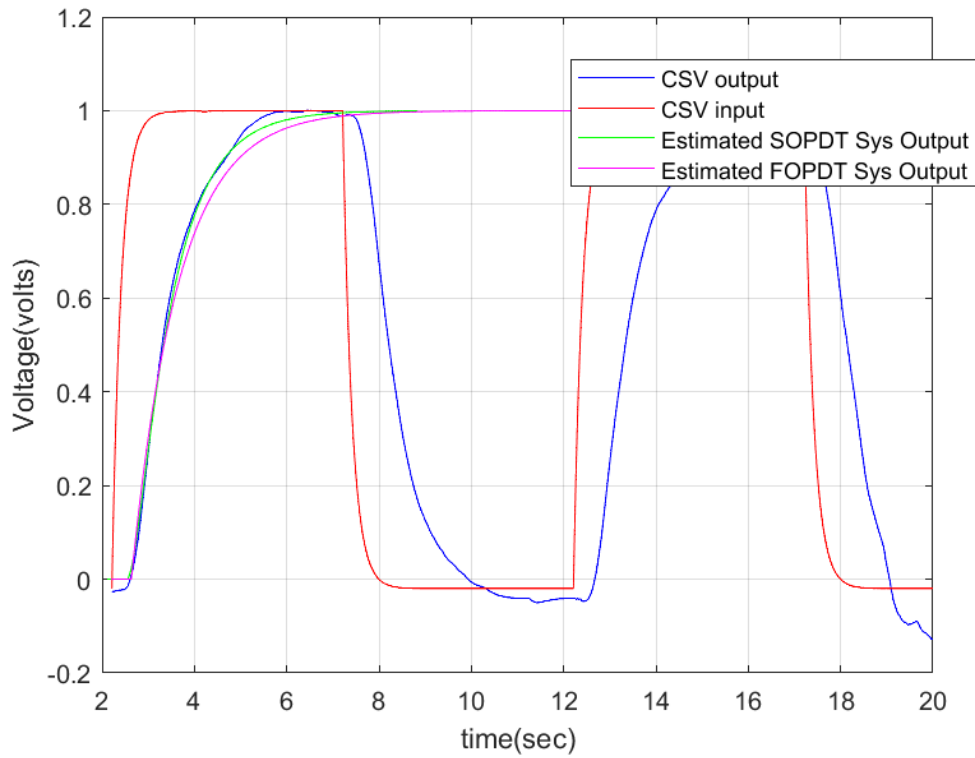


Figure 7.4b: Comparison of the responses by FOPDT model and SOPDT overdamped model with response of original system

**Table 7.3 Comparison of IAE and ISE of the estimated models**

Model	%IAE	%ISE
SOPDT Model (Eqn. 5a)	2.24	1.98
FOPDT Model (Eqn. 5b)	2.68	2.55

Given that the SOPDT overdamped model Eqn.(5a) demonstrates superior accuracy, we will utilize this model for the upcoming experimental simulations.

### 7.3 Design of Fuzzy logic Controller for the Trainer 37-100

In this work PI type FLC is implemented, which is shown in Figure 7.5.

The input scaling factors (ISFs)  $k_e$  &  $k_{de}$  normalize the real-world inputs to a range in which membership functions are defined. The output scaling factor (OSF) is used to change the normalized control effort to its practical value. The relation between real and normalized values of the parameters can be simply given as:

$$E = ek_e; \Delta E = \Delta ek_{de}; \Delta u = \Delta Uk_o$$

where  $E$  and  $\Delta E$  are the normalized inputs of the FLC controller,  $\Delta U$  is the normalized FLC output;

$e, \Delta e$  are actual inputs to the FLC and  $\Delta u$  is actual output.

$k_e$ ,  $k_{de}$ ,  $k_o$  are the error scaling factor, the change of error scaling factor and the control effort change scaling factor, respectively.

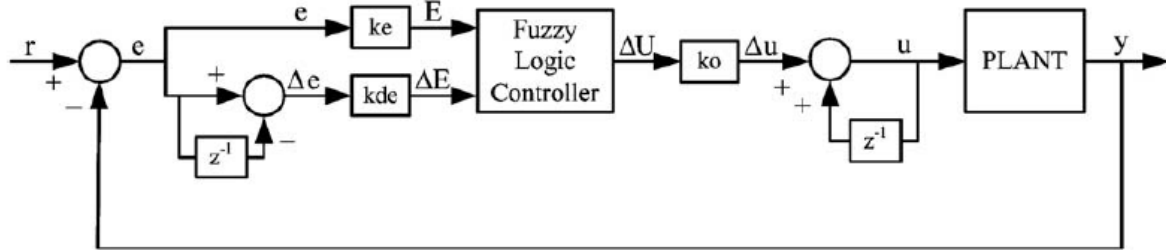


Figure 7.5: The basic control block diagram illustration with PI type FLC

Membership function for inputs (error  $e$ , change of error  $\Delta e$ ) and output ( $\Delta u$ ) are created. The input variables are decomposed into at least seven fuzzy linguistic levels (NL, NM, NS, ZE, PS, PM, PL) as shown in Figure 7.6a in order to make a considerable distinction between the fuzzy regions and, thus, to obtain fine-tuned control action. The universe of discourse is chosen to be  $[-1, 1]$  for the membership functions of input and output variables. The input and output parameters are scaled to fit this range via scaling factors. We use symmetric triangles (except the two membership functions at the extreme ends) with equal bases.

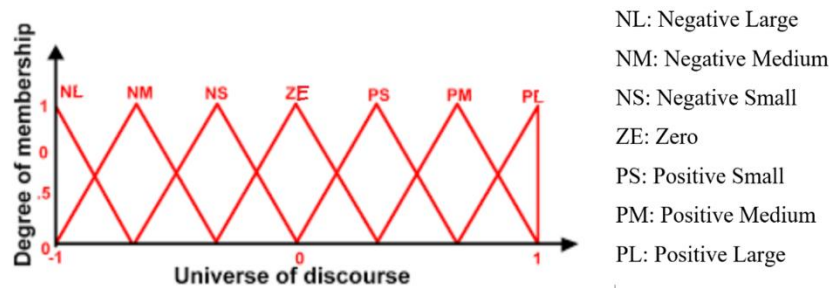


Figure 7.6a: Membership function with linguistic labels in FLC

The rule base, as illustrated in Figure 7.6b, outlines the set of guidelines or conditions governing the system's behavior. This table likely contains a series of input-output relationships, indicating how the system responds or produces output based on different input conditions.

Meanwhile, Figure 7.7 visually represents how the system's output varies in response to changes in its inputs. It provides a graphical depiction of the relationships captured in the rule base, allowing for a more intuitive understanding of how the system behaves under different scenarios

delE/E	NL	NM	NS	ZE	PS	PM	PL
NL	NL	NL	NL	NM	NS	NS	ZE
NM	NL	NM	NM	NM	NS	ZE	PS
NS	NL	NM	NS	NS	ZE	PS	PM
ZE	NL	NM	NS	ZE	PS	PM	PL
PS	NM	NS	ZE	PS	PS	PM	PL
PM	NS	ZE	PS	PM	PM	PM	PL
PL	ZE	PS	PS	PM	PL	PL	PL

Figure 7.6b: Fuzzy Rules for Computation of  $\Delta U$

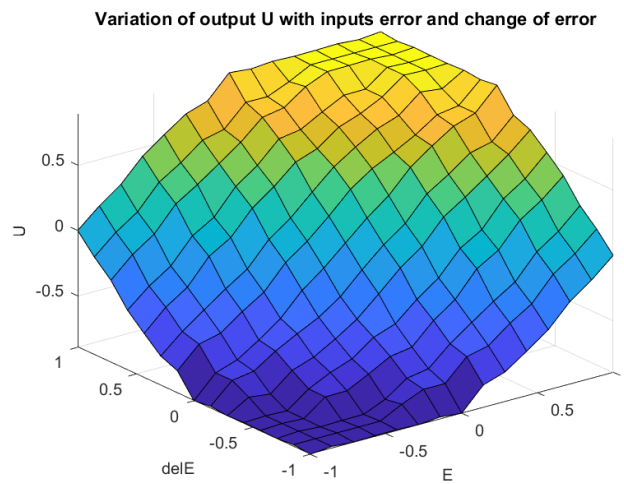


Figure 7.7: Variation of output with inputs

### 7.3.1 Implementation of PI-FLC in MATLAB & Simulink environment

The plant considered is a SOPDT system whose transfer function is (5). The PI-FLC shown in Figure 7.5 is implemented in MATLAB and Simulink environment. For this, two inputs were chosen for Error as 'E' and change of Error as ' $\Delta E$ ' and one output as 'U'. A triangular membership function was used in the universe of discourse  $[-1,1]$ . Each membership has seven linguistic labels as shown in Figures 7.8, 7.9, and 7.10 for E,  $\Delta E$  and  $\Delta U$  respectively.



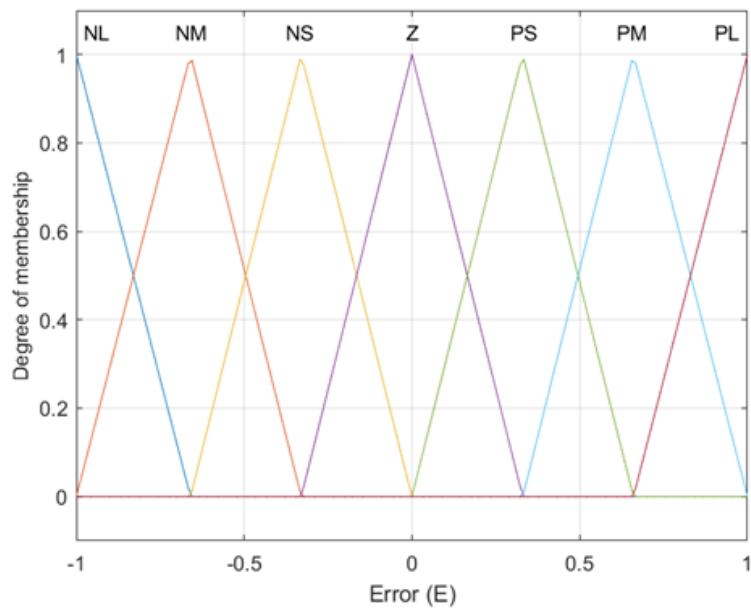


Figure 7.8: Membership function for Error(E)

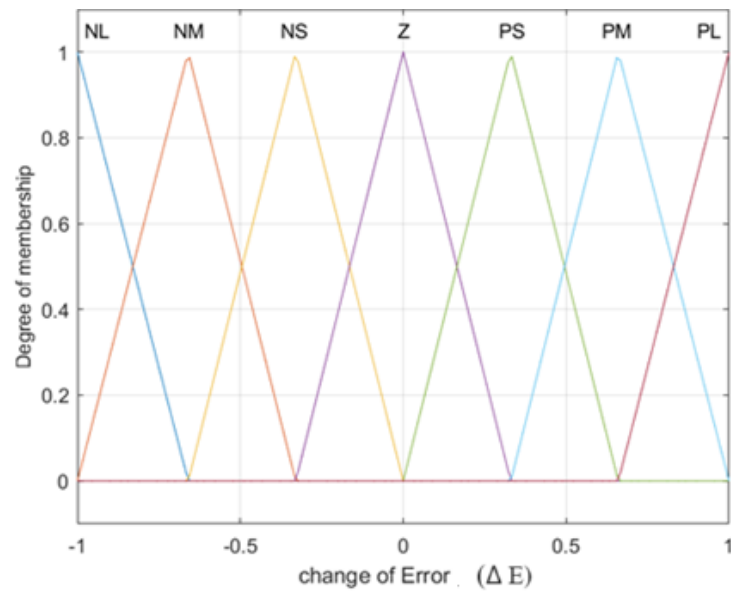


Figure 7.9: Membership function for change of Error ( $\Delta E$ )



In a conventional Fuzzy Logic Controller (FLC), scaling factors are manually tuned through a trial-and-error process. To avoid the trial and error method a self-tuned FLC is designed in which a gain updating factor ‘alpha’ is incorporated which automatically adjusts its parameters based on a predefined rule base as shown in Figure 7.12(a) and the corresponding membership function shown in Figure 7.13 [95], allowing dynamic adaptation to changing system conditions. Figure 7.12(b) visually represents how the ‘alpha’ varies in response to changes in its inputs. This automation enhances efficiency and responsiveness, particularly in complex systems where manual tuning may be impractical. Figure 7.14(a) illustrates the implementation of a ZN-PI controller and a self-tuned PI-FLC controller within the Simulink environment. In Simulink, these controllers are represented using blocks and connections that visually depict the control algorithm. Also Figure 7.14(b) illustrates PI-FLC and self-tuned PI-FLC in Simulink environment with their evaluated IAE, ISE and ITAE values.

$\Delta e / e$	NB	NM	NS	ZE	PS	PM	PB
NB	VB	VB	VB	B	SB	S	ZE
NM	VB	VB	B	B	MB	S	VS
NS	VB	MB	B	VB	VS	S	VS
ZE	S	SB	MB	ZE	MB	SB	S
PS	VS	S	VS	VB	B	MB	VB
PM	VS	S	MB	B	B	VB	VB
PB	ZE	S	SB	B	VB	VB	VB

Figure 7.12: (a)Rule base for output scaling factor alpha

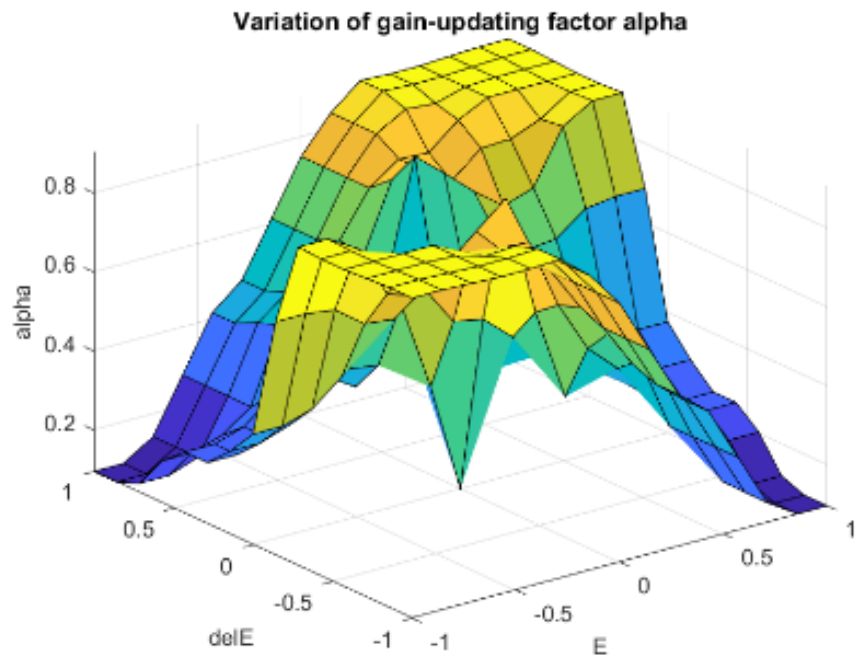


Figure 7.12 (b) Variation of alpha with error and change of error

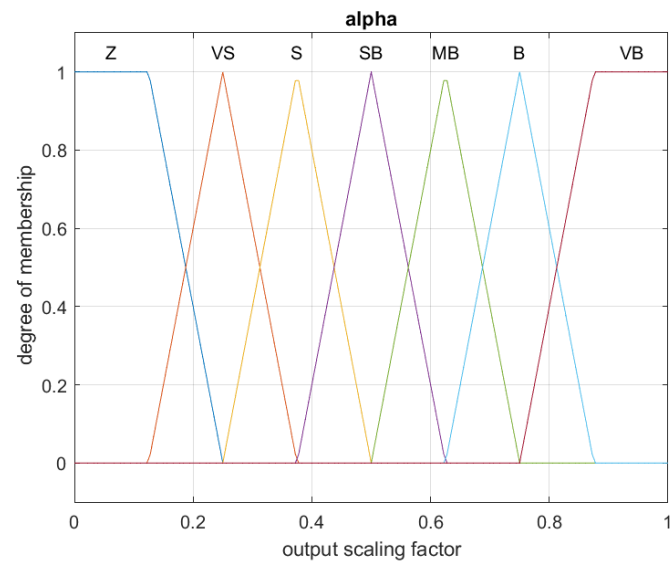


Figure 7.13: Membership function for output scaling factor alpha

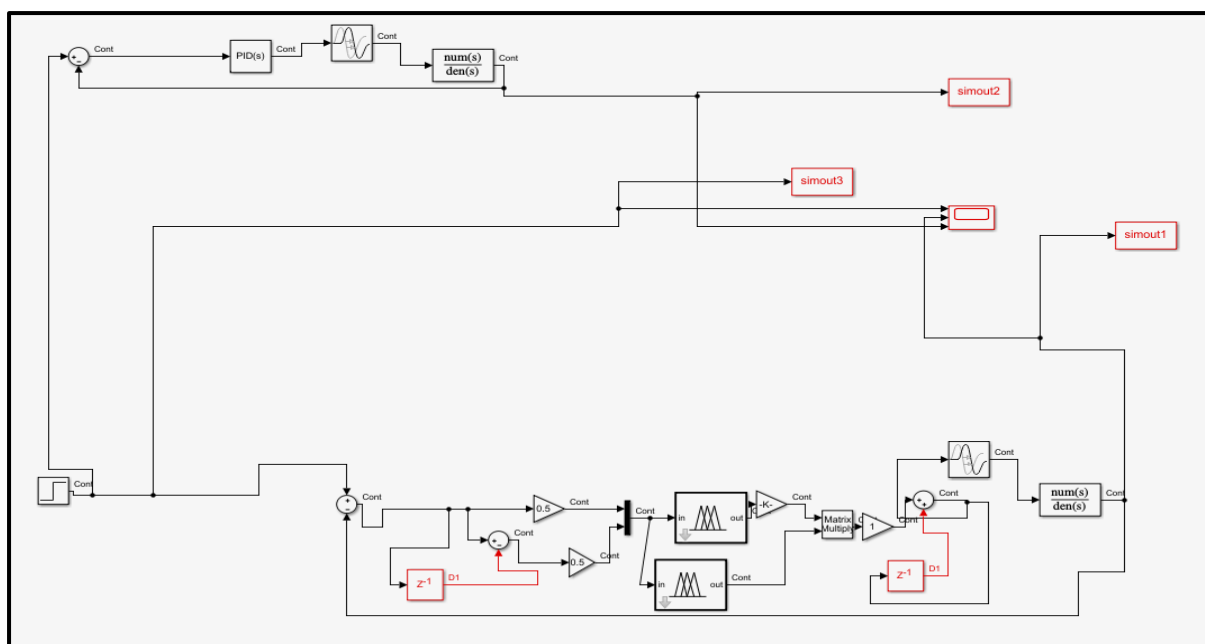


Figure 7.14(a): Implementation of ZN-PI and self-tuned PI-FLC in Simulink

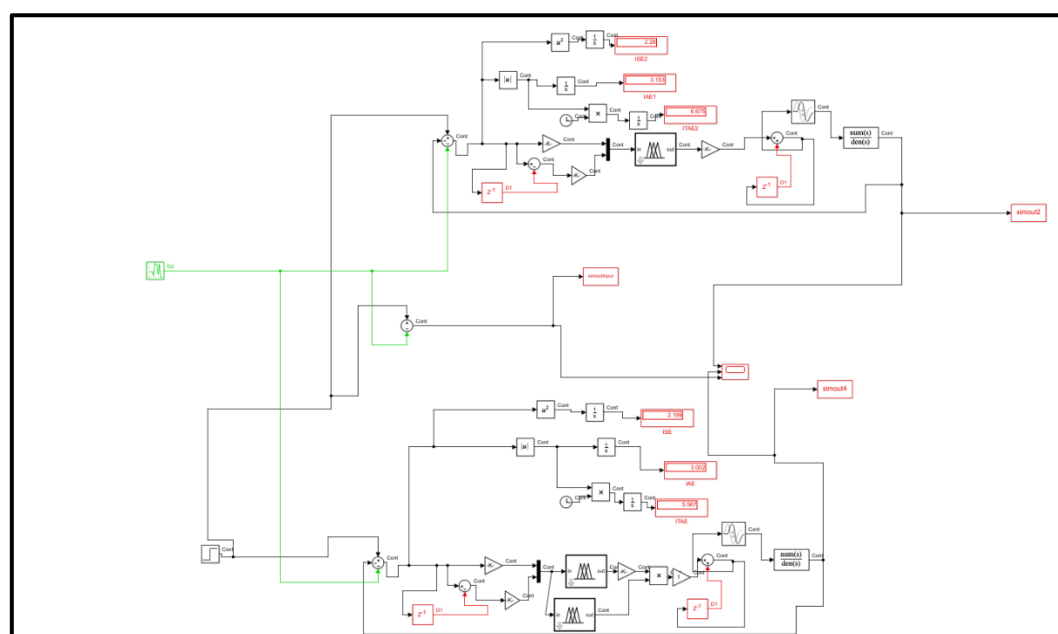


Figure 7.14(b): Implementation of PI-FLC and self-tuned PI-FLC in Simulink with display of IAE, ISE and ITAE values

To assess the performance of the controllers, a closed-loop test is conducted. The comparison is based on various performance indices, including Integral Time Absolute Error (ITAE), Integral Absolute Error (IAE), and Integral Squared Error (ISE). These indices provide insights into how well the controllers are able to regulate the system and minimize errors over time. Table 7.4 presents a comparison of the performance indices (IAE, ITAE, ISE) of the PI-FLC, self-tuned PI-FLC and ZN-tuned PI controller for the SOPDT system (5).

Furthermore, the time domain specifications of both controllers are analyzed and compared. This involves evaluating parameters such as rise time, settling time, and % overshoot. Table 7.5 presents the time-domain performance criteria associated with each of them. These metrics help in understanding the dynamic behavior of the system and how quickly it responds to changes in input. Figure 7.15 provides a visual comparison of the responses generated by the different methods under consideration.

**Table 7.4 Comparison of performance index for the Trainer Model**

Performance Index	ZN-PI	PI-FLC	Self-tuned PI- FLC
IAE	1.286	2.98	3.002
ITAE	1.682	6.67	8.829
ISE	0.824	2.06	2.199

**Table 7.5 Time domain Performance Criteria for FLC and ZN-PI tuned system**

	ZN-PI	PI-FLC	Self-tuned PI- FLC
Rise time(sec)	0.411	2.82	2.91
%Overshoot	21.05	7.82	0
Settling Time (sec)	6.3	9.9	7.9

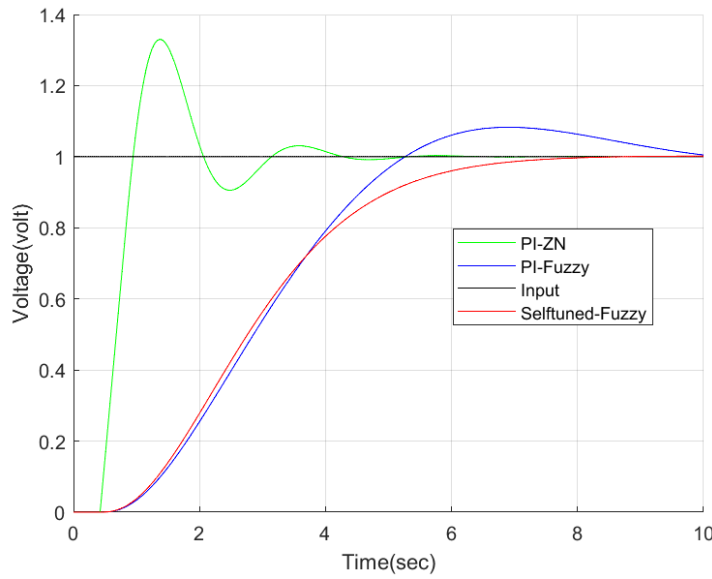


Figure 7.15: Comparison of the responses generated by the different methods( PI-ZN and Fuzzy)

The Ziegler-Nichols Proportional-Integral (ZN-PI) controller is effective in achieving a quick response in a system. However, it often leads to a notable peak overshoot, where the system briefly surpasses the desired output before stabilizing, as illustrated in Figure 7.15. In contrast, a Proportional-Integral (PI) Fuzzy Logic Controller (FLC) is renowned for minimizing overshoot, even though its response may be slower compared to the ZN-PI controller. Despite its slower nature, the PI-FLC demonstrates improved settling time, as depicted in Figure 4.15. Consequently, the FLC approach is generally considered more advantageous, offering enhanced performance with reduced overshoot and improved settling time. The self-tuned PI-FLC contributes to the dynamic performance of the control system by refining the rise time, settling time, and peak overshoot of the conventional PI-FLC, thereby promoting a more efficient and precise control of the system.

## 7.4 Design of Genetic Algorithm based PID Controller for the Trainer 37-100

A Genetic Algorithm was implemented for PID tuning in the control of the Trainer 37-100. The fitness function utilized aimed to minimize the worst-case value among three key performance metrics: Integral of Absolute Error (IAE), Integral of Squared Error (ISE), and Integral of Time-weighted Absolute Error (ITAE). The optimization process was conducted with a population size of 50 individuals over a span of 20 generations.

The well known Ziegler Nichols tuning rule was also applied to the estimated model (5) of the Trainer 37-100 and it was compared with the GA algorithm tuned PID controller.

The study assessed the cost functions ITAE, ISE, and IAE within a closed-loop testing scenario. This evaluation was carried out using a PID controller configuration with equal proportional gain  $K_p$ , integral gain  $K_i$ , and derivative gains  $K_d$ , such that  $K_p = K_i = K_d=0.5$ ) in Simulink, as illustrated in Figure 7.16. The resulting values of these cost functions, namely ITAE, ISE, and IAE, are presented in the second column of Table 7.6. Among these values, the maximum or most unfavorable outcome belongs to the cost function ISE.

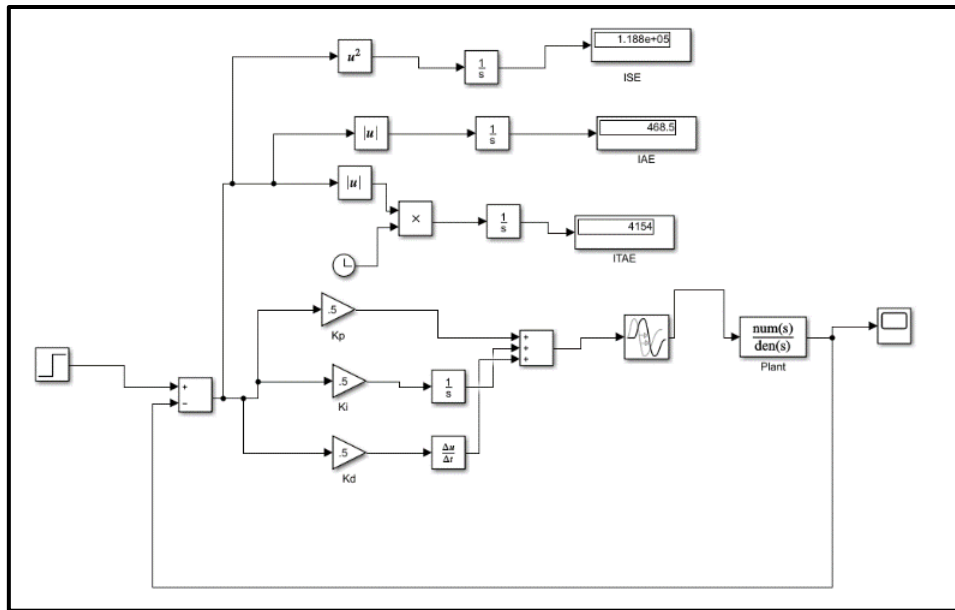


Figure 7.16 Simulation in Simulink for evaluation of cost function when  $K_p = K_i = K_d=0.5$

In accordance with the description provided in Chapter 4, the Genetic Algorithm (GA) was employed separately for each of the cost functions. The objective was to determine the optimal tuning values for  $K_p$ ,  $K_i$ ,  $K_d$ . The results of this optimization process, which corresponds to the minimized cost function, is documented and presented in Table 7.6.

**Table 7.6 Optimal tuning values of  $K_p, K_i, K_d$  for different cost functions**

Cost function	$K_p = K_i = K_d=0.5$ (initial values)	GA-Minimized	Best tuning values for minimized cost function		
			$K_p$	$K_i$	$K_d$
ITAE	4154	0.5062	1.663	1.128	0.429
ISE	1188	0.7108	1.92	1.95	0.141
IAE	4680	0.4142	1.89	1.86	0.547



Figure 7.17 provides a visual comparison of the responses generated by the different methods under consideration. To further evaluate and compare these methods, Table 7.7 presents the time-domain performance criteria associated with each of them. Interestingly, even though IAE represents the worst-case scenario among the evaluated cost functions, it outperforms both ITAE and ISE in this context.

**Table 7.7 Time domain Performance Criteria of GA-tuned and ZN-PI tuned system**

	GA-ITAE	GA-IAE	GA-ISE	ZN-PI
Rise time(msec)	600.1	520.2	426.3	411.7
%Overshoot	3.64	6.98	15.45	21.05
Settling Time (sec)	6.4	6.3	6.5	6.3

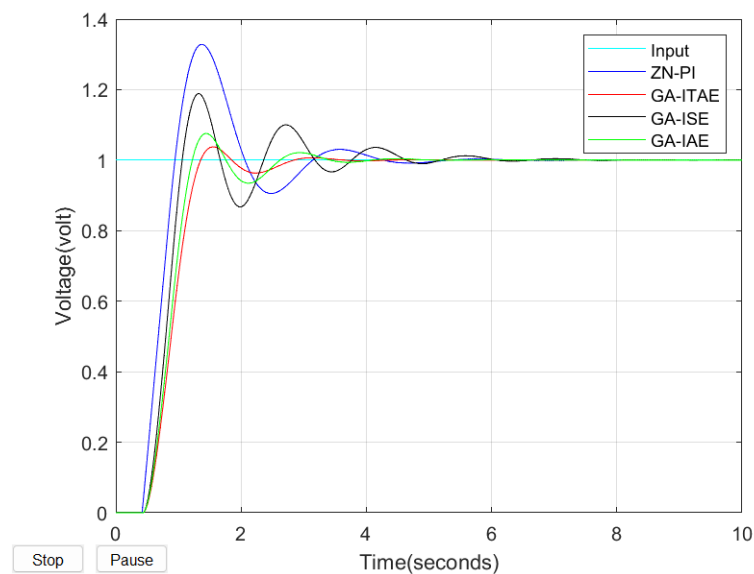


Figure 7.17 Response of Ziegler Nichols and GA with different cost functions (IAE, ITAE and ISE)

In the comparison (from Figure 7.17 and Table 7.7) between PID tuning methods, Genetic Algorithms (GA) outperforms Ziegler-Nichols, highlighting GA's effectiveness in optimizing PID controller parameters for improved control system performance.

Moreover the minimizing the worst case cost function IAE, performs better than ITAE and ISE. Ziegler-Nichols is a classical method relying on manual oscillation induction, often providing only decent initial settings, which might not work well for complex or nonlinear systems, assuming linear behavior. In contrast, Genetic Algorithms, inspired by natural selection, evolve potential solutions over generations to find optimal PID parameters. The key to GA's success lies in its fitness function,

quantifying system performance and customization to specific system dynamics and objectives. This function ensures objective, efficient, and adaptable PID tuning, making GA a preferred choice for control system optimization.

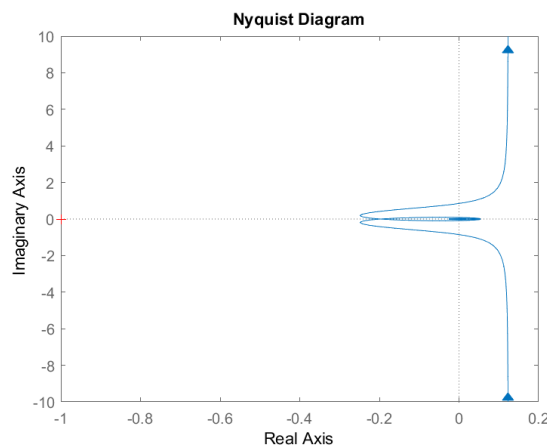
### 7.4.1 Sensitivity and stability analysis

The Table 7.8 presents a detailed comparison between the sensitivity and stability analyses of controllers: GA-tuned PID controllers and Ziegler-Nichols (ZN) PI controllers. The findings clearly demonstrate that the GA-tuned PID controllers exhibit lower sensitivity in comparison to the ZN method, with GA-IAE being the least sensitive among them.

Furthermore, the stability metrics, namely Gain Margin (GM) and Phase Margin (PM), support this observation. Both the GM and PM values for the GA-tuned PID controllers are higher than those for the ZN PI controller. Additionally, when examining the Nyquist plots (Figures 7.18 a,b,c,d), it becomes evident that the GA-tuned PID controllers are more stable. This is evidenced by the greater distance observed between the Nyquist plots and the (-1.0) point for the GA controllers as compared to the ZN controller.

**Table 7.8 Sensitivity and stability comparison**

	Sensitivity	Gain Margin (dB)	Phase Margin(degree)
GA-ITAE	1.270	14.1	92.1
GA-IAE	1.242	14.1	97.3
GA-ISE	1.284	13.0	97.4
ZN-PI	1.781	5.79	54.7



**Figure 7.18a Nyquist Plot for GA-ITAE**

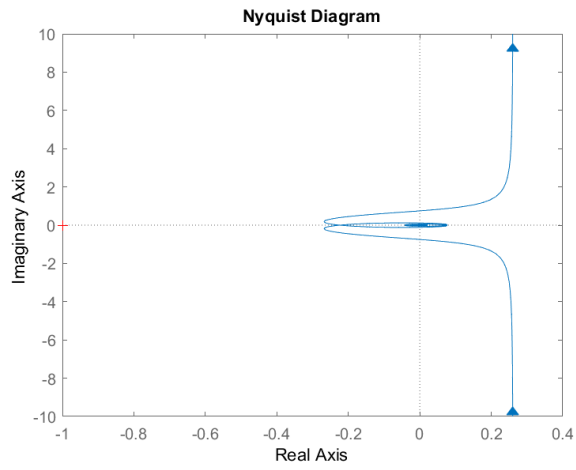


Figure 7.18b Nyquist Plot for GA-IAE

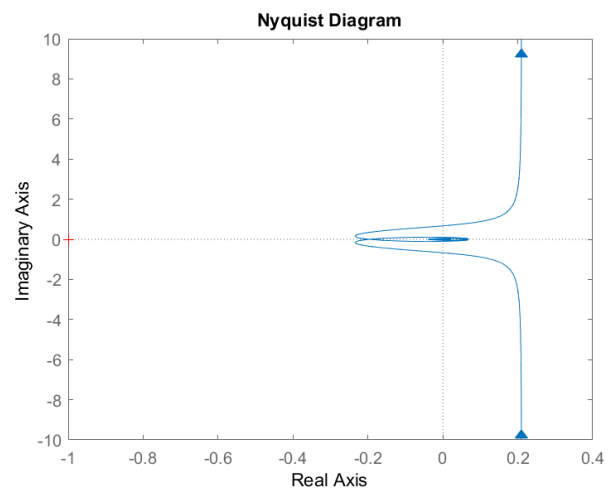


Figure 7.18c Nyquist Plot for GA-ISE

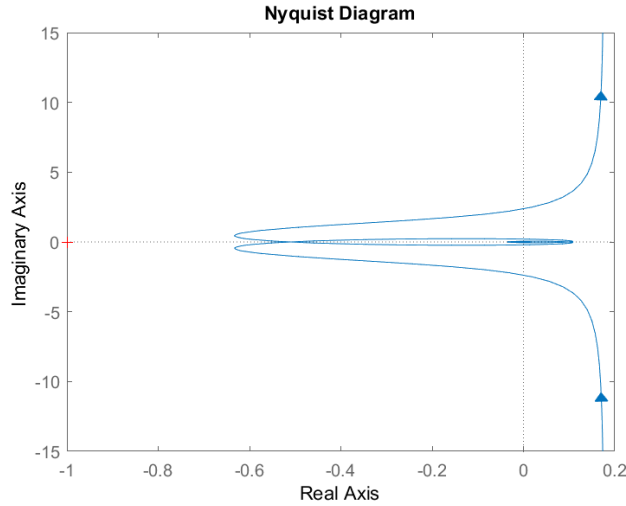


Figure 7.18d Nyquist Plot for ZN-PI

## 7.5 Effect of Measurement Noise

Measurement noise usually corrupts the system output data and hence, the performance of the control algorithm needs evaluation in the presence of measurement noise to check its robustness. A well-crafted controller should possess robust characteristics to effectively handle the inevitable presence of measurement noise. This robustness ensures that the controller can maintain stability and performance even when dealing with uncertainties and variations in the sensor readings. In essence, robust design is a key element in enhancing the controller's ability to navigate real-world conditions marked by inherent measurement uncertainties.

### 7.5.1 Comparison of Fuzzy tuned PI (PI-FLC) control and Ziegler Nichols tuned PI (ZN-PI) Control for the SOPDT system under measurement noise

In order to evaluate the performance of both Proportional-Integral Fuzzy Logic controller (PI-FLC) and a Ziegler-Nichols Proportional-Integral (ZN-PI) controller on the Second-Order Plus Dead Time (SOPDT) system (5), a simulation experiment was conducted by incorporating measurement noise into the system's output as shown in Figure 7.19. The nature of the measurement noise was modeled as White Gaussian, characterized by a zero mean and variance of 0.0064 which is 8 with a sampling time of 0.01 sec. This corresponds to a specified standard deviation of 8% of the steady-state value. The inclusion of this level of noise introduces

variability into the system, simulating real-world conditions where external factors may affect the measurements.

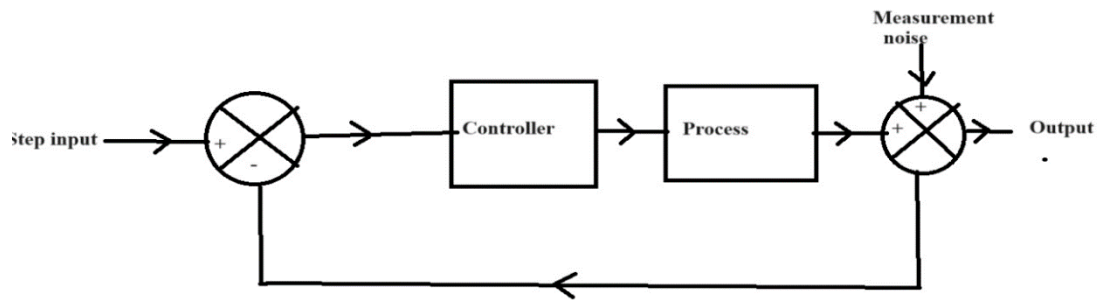


Figure 7.19: Closed Loop system with measurement noise added

The outcomes of the experiment, detailing the performance of both the PI-FLC and the ZN-PI controller under these noise conditions, are summarized in Table 7.9. Additionally, a visual representation of the results is provided in Figure 7.20. These tables and figures serve as a comprehensive snapshot of how the controllers responded to the SOPDT system with the introduced measurement noise, offering insights into their robustness and effectiveness in practical, noisy environments.

**Table 7.9 Comparison of performance index for SOPDT system with measurement noise**

Performance Index	ZN-PI	PI-FLC	Self-tuned PI- FLC
IAE	1.656	1.725	1.198
ITAE	4.263	3.092	2.939
ISE	0.875	0.569	0.436

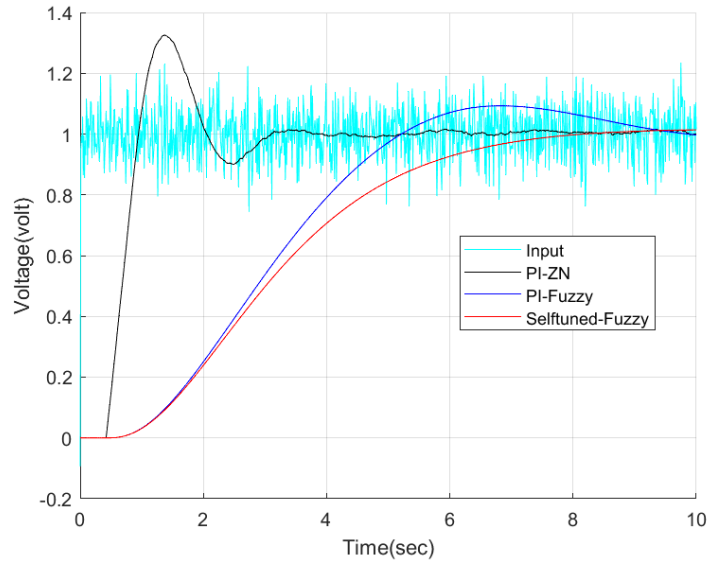


Figure 7.20 Response of ZN-PI, PI-FLC and self-tuned PI-FLC under measurement noise (white Gaussian noise with zero mean and standard deviation 8% of steady state value and sampling time 0.01 sec)

The self-tuned FLC outperforms both the conventional FLC and Ziegler-Nichols Proportional-Integral (ZN-PI) controller as it achieves minimal overshoot and improved settling time. This enhanced performance comes at the expense of a somewhat sluggish response.

### 7.5.2 Comparison of GA tuned PID control and Ziegler Nichols tuned PI Control for the Trainer Model under measurement noise

The outcomes of the experiment, detailing the performance of both the GA controller and the ZN-PI controller under these noise conditions, are summarized in Table 7.10. Additionally, a visual representation of the results was provided in Figure 7.21-7.23. These tables and figures serve as a comprehensive snapshot of how the controllers responded to the SOPDT system with the introduced measurement noise, offering insights into their robustness and effectiveness in practical, noisy environments.

**Table 7.10 Comparison of performance index for SOPDT system with measurement noise and the tuning parameters**

Performance Index (Cost function)	ZN-PI	PID tuned by GA	Best tuning values for minimized cost function by GA		
			$K_p$	$K_i$	$K_d$

IAE	1.656	0.476	1.500	1.116	0.375
ITAE	4.263	3.896	1.246	1.031	0.187
ISE	0.875	0.7151	2.064	1.412	0.531

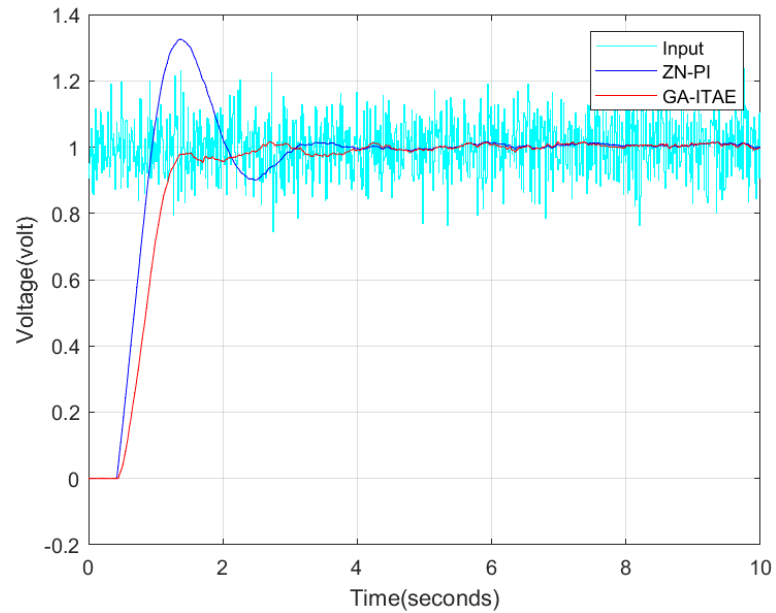


Figure 7.21 Response of ZN-PI and GA-PID (using ITAE cost-function) under measurement noise

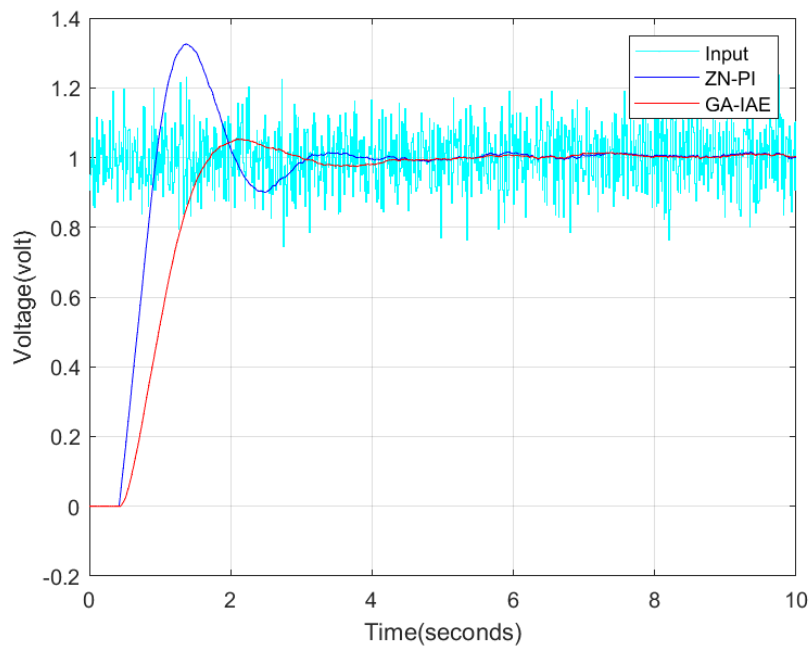


Figure 7.22 Response of ZN-PI and GA-PID (using IAE cost-function) under measurement noise

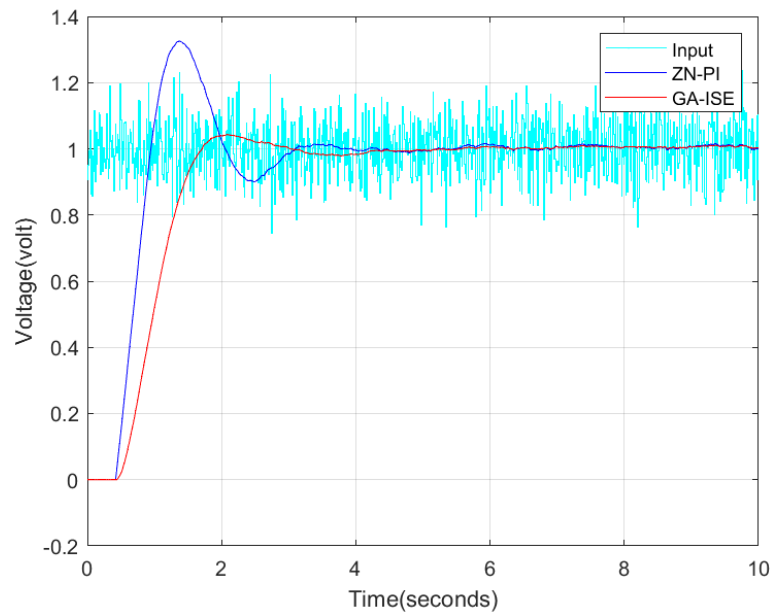


Figure 7.23 Response of ZN-PI and GA-PID (using ISE cost-function) under measurement noise

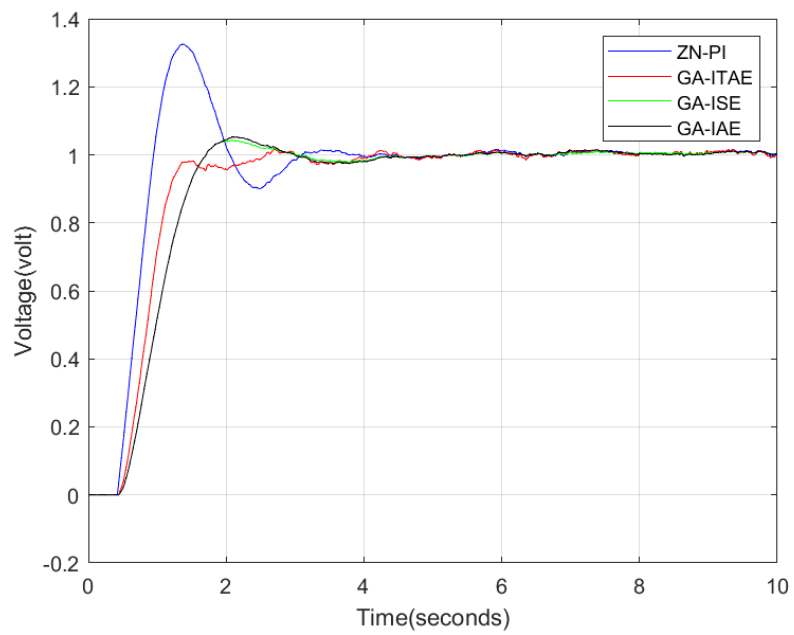


Figure 7.24 Response of Ziegler Nichols and GA with different cost functions (IAE, ITAE and ISE) under measurement noise



Table 7.10 indicates that the GA-IAE controller exhibits the lowest error values when compared to other GA-tuned PID controllers and the conventional ZN-PI controller. This observation aligns with the sensitivity analysis, which highlighted that the GA-IAE controller is the least affected by noise. Additionally, the worst-case analysis further confirms the superior performance of the GA-IAE controller over other GA controllers and the ZN-PI co

## 7.6 Comparison of Self tuned Fuzzy-PI Controller and Self Adaptive GA based PID controller

The self-adaptive Genetic Algorithm (GA) is designed to dynamically select the cost function that yields the least favorable outcome when PID gains (proportional, integral, and derivative with values of  $K_p = K_i = K_d=0.5$ ) are kept constant as shown in Figure 7.16 In this scenario, the chosen cost function is IAE, as it results in the most unfavorable value. This decision is supported by the Table 7.7 which illustrates that IAE provides the best time domain performance criteria among the considered cost functions.

In Table 7.10, a comprehensive analysis of time domain performance criteria is presented, drawing a comparison between the self-tuned Fuzzy Logic Controller (FLC) and the self-adaptive Genetic Algorithm (GA) tuned PID controller with Ziegler-Nichols tuned proportional-integral (ZN-PI) controller. Concurrently, Figure 7.25 provides a visual representation of this comparison. Notably, the self-tuned FLC stands out by showcasing a zero overshoot, indicating a precise and stable response. However, this achievement comes at the cost of an elevated rise-time. The Table 7.11 and Figure 7.24 collectively underscore the trade-offs inherent in the system dynamics, emphasizing the need to balance overshoot and rise-time based on specific application requirements.

**Table 7.11 Comparison of Time Domain Performance Criteria (self-tuned FLC and self-adaptive GA with ZN-PI)**

	Self-tuned Fuzzy (PI-FLC)	Self-adaptive GA (GA-IAE)	ZN-PI
Rise time(msec)	1.992 sec	520.2 msec	411.7 msec
%Overshoot	0	6.98	21.05
Settling Time (sec)	7.9	6.3	6.3

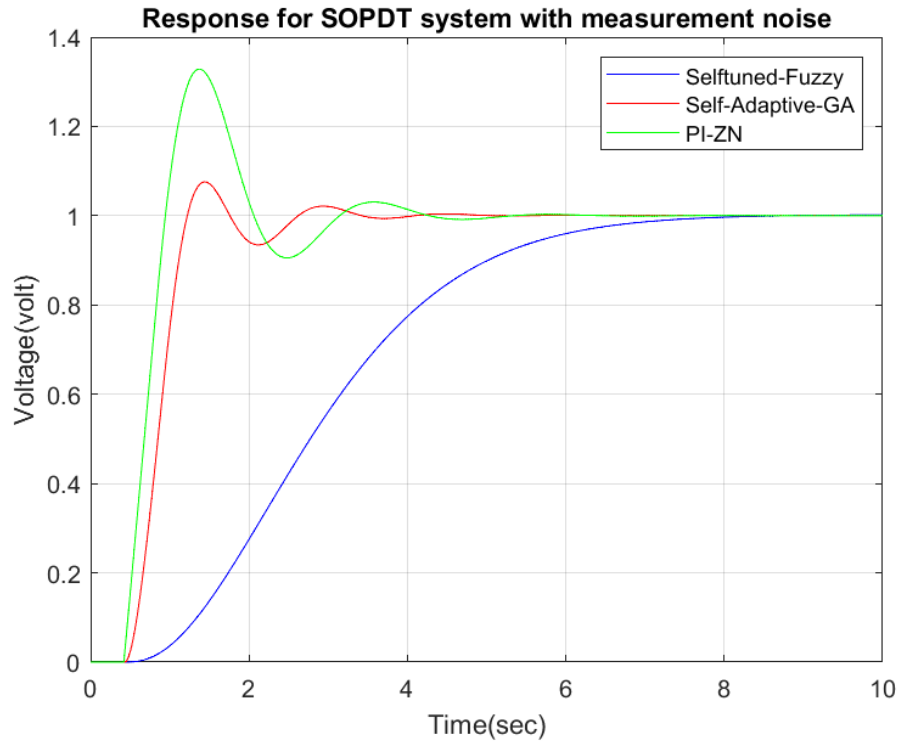


Figure 7.25: Comparison of the responses generated by the different methods (PI-ZN and Self-tuned Fuzzy and Self-adaptive GA)

The results presented in Figure 7.26 and Table 7.12 provide a comparison of the responses between self-tuned fuzzy PI controller and self-adaptive genetic algorithm (GA) tuned PID controller under the effect of measurement noise. The self-tuned genetic algorithm demonstrates a smaller overshoot and a quicker response when compared to its fuzzy counterpart. It is worth noting that the Ziegler-Nichols tuned proportional-integral (ZN-PI) method yields a high overshoot, rendering it unsuitable for real-time for non-linear systems.

**Table 7.12 Comparison of performance index for SOPDT system with and without measurement noise (self-tuned FLC and self-adaptive GA with ZN-PI)**

Performance Index	ZN-PI (ISE-value)	Self-adaptive GA (IAE-value)	Self-tuned FLC (IAE-value)
With no measurement noise	1.286	0.414	0.486
With measurement noise	1.656	0.976	0.496

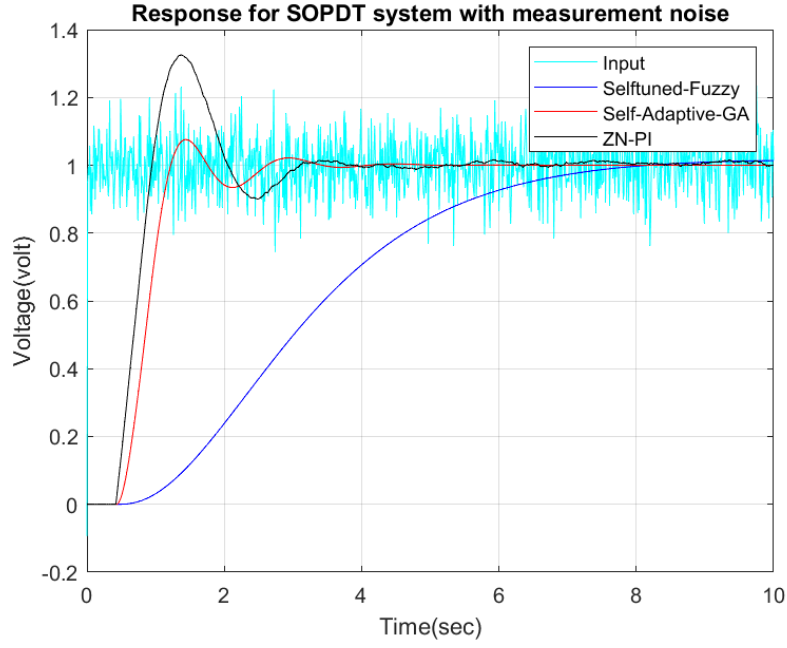


Figure 7.26: Comparison of the responses generated by the different methods (PI-ZN and Self-tuned Fuzzy and Self-adaptive GA) under measurement noise

Clearly from Table 7.12 the self-tuned FLC exhibits greater resilience to noise when compared to self-adaptive GA and conventional ZN-PI.

### 7.6.1 Comparison of controller performance under external disturbance

In the realm of control systems, disturbances represent external influences that can impact the stability and performance of a system. Understanding how different controllers respond to and mitigate these disturbances is crucial for designing robust and adaptive control strategies. The following block diagram is implemented in Simulink to test the robustness of the different controllers under test. An external disturbance 'd' affects the plant  $G(s) = \frac{\omega_n^2}{s(s+2\xi\omega_n)}$  as shown in Figure 7.27.

$$\text{Clearly } T(s) = \frac{\frac{\omega_n^2}{s(s+2\xi\omega_n)}}{1 + \frac{\omega_n^2}{s(s+2\xi\omega_n)}} = \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right)$$

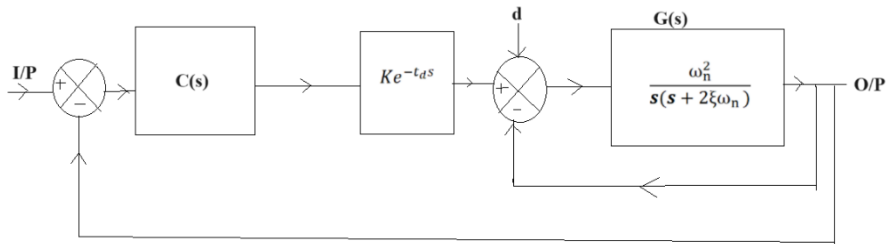


Figure 7.27 Closed Loop System with external disturbance d affecting the plant

The plant under test is the SOPDT system  $Ke^{-t_d s} \left( \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \right)$ . The robustness of the controller  $C(s)$  is tested under disturbance  $d$  which is a pulse train of period 2 sec. and amplitude 0.1V.

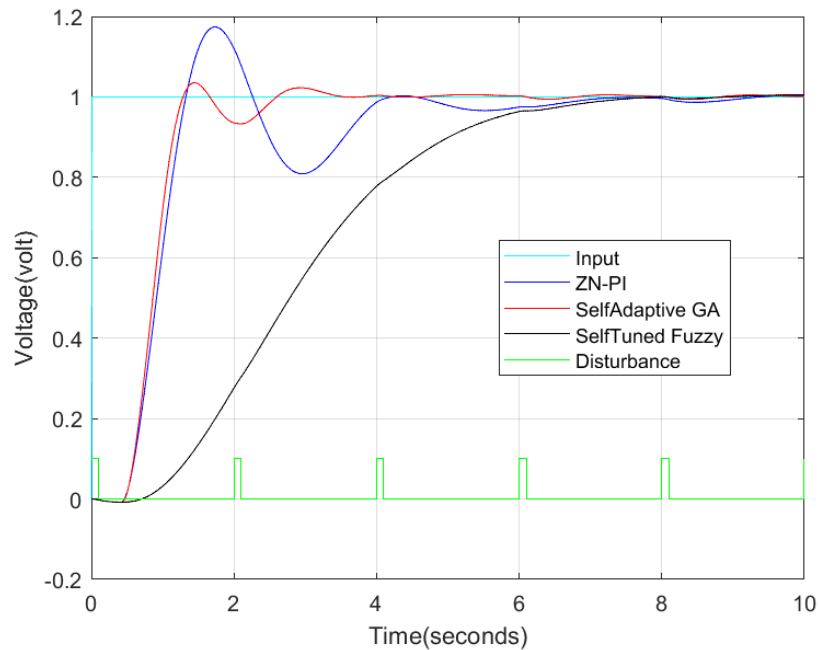


Figure 7.28 Comparison of the responses generated by the different methods( PI-ZN andSelf-tuned Fuzzy and Self-adaptive GA) under external disturbance

**Table 7.13 Comparison of Time Domain Performance Criteria (self-tuned FLC and self-adaptive GA with ZN-PI) under external disturbance**

	Self-tuned Fuzzy (PI-FLC)	Self-adaptive GA (GA-IAE)	ZN-PI
Rise time(msec)	1.992 sec	553.59 msec	614.67 msec
%Overshoot	0	3.63	18.45
Settling Time (sec)	7.5	6.8	9.1

**Table 7.14 Comparison of performance index for SOPDT system with and without disturbance (self-tuned FLC and self-adaptive GA with ZN-PI)**

Performance Index	ZN-PI (ISE-value)	Self-adaptive GA (IAE-value)	Self-tuned FLC (IAE-value)
With no disturbance	1.286	0.414	0.486
With disturbance	1.692	0.653	0.596

The results presented in Figure 7.28, Table 7.13 and Table 7.14 provide a comparison of the responses between self-tuned fuzzy PI controller and self-adaptive genetic algorithm (GA) tuned PID controller under the effect of external disturbance. The self-tuned genetic algorithm demonstrates a smaller overshoot and a quicker response when compared to its fuzzy counterpart. It is worth noting that the Ziegler-Nichols tuned proportional-integral (ZN-PI) method yields a high overshoot, rendering it unsuitable for real-time non-linear systems. Clearly from Table 7.14, the self-tuned FLC exhibits greater resilience to disturbance when compared to self-adaptive GA and conventional ZN-PI.

It may also be noted that the controller performance index and the closed loop time response performance indices for the proposed self-tuned fuzzy and self-adaptive GA controllers are superior compared to those of the Generalized Predictive Controller for a similar process kit as reported in [118].

## **7.7 Factors that contribute to the ability of FLC to handle noise effectively**

1. Fuzzification is the process of converting crisp input values into fuzzy sets. This inherently allows FLCs to capture imprecision and variability in the input signals. In the presence of noise, fuzzification provides a means to represent uncertain or imprecise information, preventing the noise from having a drastic impact on the system.
2. FLCs operate on a rule-based system which helps smooth out the effects of noise, as the controller considers a range of possible scenarios rather than relying on precise values.
3. The linguistic terms used in fuzzy sets, such as "high," "low," or "medium," inherently accommodate variations and uncertainties in the input data, making the controller less sensitive to noise.
4. The use of membership functions in FLCs allows for a gradual transition between different linguistic terms. This smooth transition helps mitigate the effects of sudden spikes or fluctuations in the input signals, which are typical characteristics of noise.

## **7.8 Summary**

This case study comprehensively explored the dynamics, identification, and modelling of Trainer 37-100 followed by the development and comparison of a fuzzy logic controller (FLC) and a Genetic Algorithm (GA) tuned PID controller. Both controllers were

systematically compared with a Ziegler-Nichols (ZN) tuned Proportional-Integral (PI) controller. The subsequent analysis, involving the evaluation of control actions under the influence of measurement noise and external disturbances, provided valuable insights into the robustness of the controllers. These findings contribute to a nuanced understanding of the strengths and limitations of FLC and GA-tuned PID controllers in real-world scenarios, offering a foundation for informed decision-making in the application of control strategies for similar processes. Through this case study, we seek not only to enhance our understanding of the Trainer 37-100 but also to contribute to the broader discourse on process control. The insights gained from this investigation hold the potential to improve advancements in control systems, offering a practical lens through which one may view the symbiotic relationship between modelling and control in integrated systems. This case study is a stepping stone toward a deeper comprehension of process control systems and their role in shaping future applications.

## **Chapter 8**

---

### **Conclusion and Future Work**

## 8.1 Summary of contributions

In conclusion, the exploration of the modelling and control of Second-Order Plus Dead Time (SOPDT) systems has provided valuable insights into the intricacies of dynamic systems characterized by delay and higher-order dynamics. Through the course of this study, various modelling techniques is examined, including analytical approaches and system identification methods, to capture the essential features of SOPDT systems accurately.

Two independent modelling techniques, based on time response transients, for modelling underdamped and critically/overdamped SOPDT systems have been proposed. Simulated SOPDT time responses are used to estimate the 2<sup>nd</sup> order equivalent canonical transfer function parameters using the proposed techniques. The results are compared with existing modelling methods to establish the efficacy of the proposed modelling techniques.

The modelling of an available two-axis coupled system, viz. a fast steering mirror assembly, using both time domain and frequency domain methods, is presented as a case study. The SOPDT parameters and the cross-coupling coefficients are estimated from measurement data.

The application of intelligent control strategies, with a comparison to traditional proportional-integral-derivative (PID) controllers, has been a central theme in addressing the challenges posed by controllers for SOPDT systems. Various tuning methodologies are explored, considering both classical and modern approaches, to optimize PID controller parameters for robust and efficient performance. The analysis of closed-loop responses, and transient behaviour has shed light on the nuances of control design for SOPDT systems.

The proposed modelling technique and the control strategies for SOPDT systems are used for a temperature control process kit and the performance analysis data are reported.

## 8.2 Future Research

As we conclude this study, several avenues for future research and development in the domain of modelling and control of SOPDT systems emerge. Some potential areas for exploration include:

- Many real-time systems exhibit characteristics of high-order systems. Therefore, there is a prospective avenue for exploration to investigate whether these real-time higher-order systems can be effectively modelled using the proposed Second-Order Plus Dead Time (SOPDT) modelling methods.



- The state-space technique for designing optimal controllers, like the Linear Quadratic Regulator (LQR) and Linear Quadratic Gaussian (LQG) control strategies for multiple-input multiple-output (MIMO) SOPDT systems may be explored.
- Mamdani's fuzzy logic controller was implemented in this work. Considering the future scope, exploring alternative fuzzy logic control paradigms, such as Sugeno-type controllers, can offer various advantages and open up new possibilities. Unlike Mamdani controllers, Sugeno controllers provide crisp output values rather than fuzzy sets. The output is a mathematical function of the input variables, making it more amenable to mathematical analysis and control system synthesis.
- Using Mamdani-style rules for interpretability and Sugeno-style rules for computational efficiency can provide a balanced approach. This hybrid approach combines the strengths of both Mamdani and Sugeno controllers is another future scope of this work.
- Real-Time Implementation: Translating theoretical findings into practical applications by exploring real-time implementation and validation of control strategies on physical SOPDT systems can bridge the gap between simulation and actual system behaviour.
- Robustness Analysis: Conducting a comprehensive analysis of the robustness of control strategies under uncertainties, disturbances, and variations in system parameters will contribute to the development of controllers capable of handling real-world challenges.
- Hybrid control strategies that combine the strengths of different algorithms to achieve improved performance and robustness can be another future scope.
- Address challenges associated with real-time implementation, such as computational limitations and hardware constraints, to facilitate the practical deployment of sophisticated control strategies for physical SOPDT systems



## References

- [1] Rake H. "Step response and frequency methods", Automatica 1980; Vol.16. 519-526.  
<https://doi.org/10.1016/B978-0-08-027583-3.50006-0>
- [2] Jr. Huang C.T., Clements W.C. "Parameter Estimation for Second Order plus Dead Time Model", Ind. Eng. Chem. Process Des. Dev.9 ,1982; 21,4, 601-603 801 <https://doi.org/10.1021/i200019a011>
- [3] Hwang S.H, Lai S.T. "Use of two stage least squares algorithms for identification of continuous systems with time delay based on pulse responses", Automatica, 2004; 40(9),1561-1568.  
<https://doi.org/10.1016/j.automatica.2004.03.017>
- [4] Wang Q., Zhang Y. "Robust identification of continuous systems with dead time from step responses", Automatica, 37 ,2001; 3,77-390. [https://doi.org/10.1016/S0005-1098\(00\)00177-1](https://doi.org/10.1016/S0005-1098(00)00177-1)
- [5] Sundaresan K.R., Prasad C.C., Krishnaswamy, P.R. "Evaluating Parameters from Process Transients", Ind. Eng. Chem. Process Des. Dev., 1978; Vol. 17. No. 3, 237. DOI: 10.1021/i260067a004
- [6] Rangaiah G. P, Krishnaswamy P.R. "Estimating Second-Order plus Dead Time Model Parameters", Ind. Eng. Chem. Res. 33, 1994; 1867-1871 1867. <https://doi.org/10.1021/ie00031a029>
- [7] Messner W.C. "The Last Jedi: Step Response Identification of Overdamped Second Order Systems", ACC, 2018; 978-1-5386-5428-6 DOI: <https://doi.org/10.23919/ACC.2018.8431709>
- [8] Tsung C., Huang M.F. "Estimation of Second-Order Parameters from the Process Transient by Simple Calculation" ; Ind. Eng. Chem. Res.1993; 32,228-230. DOI: <https://doi.org/10.1021/ie00013a030>
- [9] Gawthrop P.J, Nihtila M.T. "Identification of time delays using a polynomial identification method", Systems and Control Letters, 1985; 5(4),267-271. [https://doi.org/10.1016/0167-6911\(85\)90020-9](https://doi.org/10.1016/0167-6911(85)90020-9)
- [10] Nath U.M., Dey C., Mudi R.K. "Designing of internal model control proportional, integral, and derivative controller with second-order filtering for lag- and delay-dominating processes based on suitable dead time approximation", Asia-Pac J Chem Eng., 2019, Vol.14, Issue 6, e2359.<https://doi.org/10.1002/apj.2359>
- [11] Tuma M., Jura P. "Dead Time Estimation in Generalized Laguerre function basis", International Conference of Numerical Analysis and Applied Mathematics ICNAAM AIP Conf. Proc.,2020; 2293, 340005-1–340005-4. <https://doi.org/10.1063/5.0026749>
- [12] Ljung L. "On the Estimation of Transfer Functions", IFAC,1985; Vol.21 pp 677-696, No.6.  
[https://doi.org/10.1016/S1474-6670\(17\)60805-X](https://doi.org/10.1016/S1474-6670(17)60805-X)
- [13] Ferretti G. Malezzoni C., Scattolini R. "Recursive estimation of time delay in sampled systems", Automatica 1991; Vol.27 (4)653-661 [https://doi.org/10.1016/0005-1098\(91\)90056-8](https://doi.org/10.1016/0005-1098(91)90056-8)
- [14] Yuwana M., Seborg D.E. "A New Method for On-Line Tuning" AIChE Journal ,1982; Page 434, Vol. 28, No. 3, ISSN-0001-1541-82-5666-0 <https://doi.org/10.1002/aic.690280311>
- [15] Jutan A., Rodriguez E.S. II. "Extension of a New Method for On-line Controller", The Canadian Journal of Chemical Engineering, 1984; Vol.62.<https://doi.org/10.1002/cjce.5450620610>
- [16] Lee J. "On-Line PID Controller Tuning from a Single Closed-Loop Test"; AIChE Journal,1989 February Vol. 35, No.2.<https://doi.org/10.1002/aic.690350221>
- [17] Lee J., Cho W., Edgar T.F. An Improved Technique for PID Controller Tuning from Closed-Loop Tests. AIChE Journal 1990; Vol. 36, No.12.<https://doi.org/10.1002/AIC.690361212>
- [18] Suganda, P., Rangaiah, G.P., Krishnaswamy, P.R. "Online process identification from closed loop tests under PI control", Trans IChemE, ,1998; Vol 76, 0263-8762/98 <http://scholarbank.nus.edu.sg/handle/10635/91610>

- [19] Cobos J.B. “Parameter estimation for an overdamped dynamic second order system, a new approach”, IEEE 5th Colombian Conference on Automatic Control (CCAC),2021; 978-1-6654-1883-6/21 <https://doi.org/10.1109/CCAC51819.2021.9633265>
- [20] Rodríguez-Abreo O.,Rodríguez-Reséndiz J., Velásquez F.A.C., Ortiz Verdín, A.A.,Garcia-Guendulain, J.M.,Garduño-Aparicio M. “Estimation of Transfer Function Coefficients for Second-Order Systems via Metaheuristic Algorithms”. Sensors, 2021; 21, 4529. <https://doi.org/10.3390/s21134529>
- [21] Reddy S. N., Chidambaram M. “Model identification of critically damped second order plus time delay systems”, Indian Chemical Engineer, 2019; DOI:10.1080/00194506.2019.1629842
- [22] Harriott, P., Process Control, McGraw Hill, New York, NY, pp. 48–49 (1964).
- [23] Young G.E., Suresh K.S., Rao, V., Chatufale, R. “Block-Recursive Identification of Parameters and Delay in the Presence of Noise” Transactions of ASME 600, 1995; Vol. 117. <https://doi.org/10.23919/ACC.1992.4792579>
- [24] Cheres E., Eydelzon A., “Parameter Estimation of a second Order Model in the Frequency Domain from Closed Loop Data”, Trans IChemE, Vol 78, Part A, pp. 293-298, March 2000 <https://doi.org/10.1205/026387600527158>
- [25] Chen L, Han L, Huang B, Liu F. “Parameter estimation for a dual-rate system with time delay”, ISA Transactions, 2014; <http://dx.doi.org/10.1016/j.isatra.2014.01.001i>
- [26] Honório L.M, Costa E.B, Oliveira E.J, Fernandes D. A, Moreira Antonio P.G.M “Persistently-exciting signal generation for Optimal Parameter Estimation of constrained nonlinear dynamical systems”, ISA Transactions, 2018; <https://doi.org/10.1016/j.isatra.2018.03.024>
- [27] Anbarasan K, Srinivasan K. “Design of RTDA controller for industrial process using SOPDT model with minimum or non-minimum zero” ISA Transactions 57 2015; 231–244 <http://dx.doi.org/10.1016/j.isatra.2015.02.016> 0019-0578/& 2015
- [28] Box G., Tiao G. Bayesian Inference in Statistical Analysis, Massachusetts Addison Wesley Reading ,1973
- [29] Takagi T, Sugeno M. “Fuzzy identification of systems and its applications to modeling and control”, IEEE Transactions on Systems Man Cybernetics, 1985; 15:116–32.
- [30] Prakash J., Srinivasan K. “Design of nonlinear PID controller and nonlinear model predictive controller for a continuous stirred tank reactor”, ISA Transactions 48,2009; 273–282 doi:10.1016/j.isatra.2009.02.001
- [31] Nagy-Kiss AM, Schutz G., Ragot J “Parameter estimation for uncertain systems based on fault diagnosis using Takagi– Sugeno model” ISA Transactions ,2015;<http://dx.doi.org/10.1016/j.isatra.2014.11.022i>
- [32] Cui R, Wei Y., Cheng S., Wang Y. “An innovative parameter estimation for fractional order system with impulse noise”, ISA Transactions 2017<http://dx.doi.org/10.1016/j.isatra.2017.06.025i>
- [33] Rahili S, Ren W. “Distributed Continuous-Time Convex Optimization with Time-Varying Cost Function”, IEEE Transactions on Automatic Control, 2016; 0018-9286 (c) doi10.1109/TAC.2016.2593899, IEEE
- [34] Bernstein A., Dall’Anese E., Simonetto A. “Online Primal-Dual Methods with Measurement Feedback for Time-Varying Convex Optimization”, IEEE Transactions on Signal Processing, Vol 67, No.8, April 1, 2019 <https://doi.org/10.1109/TSP.2019.2896112>
- [35] Cox C., Tindle J., Burn K. “A comparison of software-based approaches to identifying FOPDT and SOPDT model parameters from process step response data”, Applied Mathematical Modelling, Elsevier 2016; 0307-904X/\_ 2015 Elsevier Inc. <https://doi.org/10.1016/j.apm.2015.05.007>

- [36] Rodríguez-Abreo O., Rodríguez-Reséndiz J., Velásquez F.A.C., Ortiz Verdín, A.A.,Garcia-Guendulain,J.M.,Garduño-Aparicio M. “Estimation of Transfer Function Coefficients for Second-Order Systems via Metaheuristic Algorithms”, *Sensors*, 2021; 21, 4529. <https://doi.org/10.3390/s21134529>
- [37] W. Byrski. “A new method of multi-inertial system identification by the Strejc model”. In (W. Mitkowsli, J. Kacprzyk, K. Oprzedkiewicz, and P. Skruch, editors), *Trends in Advanced Intelligent Control, Optimization and Automation, Advances in Intelligent Systems and Computing* 577. Springer International Publishing AG, 2017.
- [38] Víctor M. Alfaro & Ramon Vilanova “Control of high-order processes: repeated-pole plus dead-time models' identification”, *International Journal of Control*, 2021;DOI: 10.1080/00207179.2021.1954240
- [39] Rudin, W. *Principles of Mathematical Analysis*, 3rd ed. New York: McGraw-Hill, 1976.
- [40] Gradshteyn, I. S., Ryzhik, I. M. *Tables of Integrals, Series, and Products*, 6th ed. San Diego, CA: Academic Press, p. 1132, (2000)
- [41] Rao S.S. *Engineering Optimization Theory and Practice*.John Wiley pg363-398
- [42] Ljung L., Chen T. “Convexity Issues in System Identification”, 10th IEEE International Conference on Control and Automation (ICCA), 2013; doi:10.1109/ICCA.2013.6565206
- [43] Raj A., Selvi J., Kumar D. and Raghavan S., “Design of cognitive decision-making controller for autonomous online adaptive beam steering in free space optical communication system”, *Wireless Personal Communications*, 2015 ;84(1), pp.765-799,  
DOI : <https://doi.org/10.1007/s11277-015-2660-3>
- [44] Willke B., “Stabilized lasers for advanced gravitational wave detectors”, *Laser & Photonics Reviews*, 2010; 4(6), pp.780-794,  
DOI : <https://doi.org/10.1002/lpor.200900036>
- [45] Ulander K., “Two-axis beam steering mirror control system for precision pointing and tracking applications” (No. UCRL-TH-219069). Lawrence Livermore National Lab (LLNL), Livermore, CA (United States),2006  
DOI : <https://doi.org/10.2172/893570>
- [46] Peev G., Simidjievski N. and Džeroski S., “Modeling of dynamical systems: a survey of tools and a case study”, In 20th International Multiconference Information Society-IS, 2017; (pp. 15-18).
- [47] Hei M., Zhang L.C., Zhou Q.K., Lu Y.F. and Fan D.P., “Model-based design method of two-axis four-actuator fast steering mirror system”, *Journal of Central South University*, 2015; 22(1), pp.150-158,  
DOI : <https://doi.org/10.1007/s11771-015-2505-y>
- [48] Zhou Q., Ben-Tzvi P. and Fan D., “Design and analysis of a fast steering mirror for precision laser beams steering”, *Sensors & Transducers*, 2009; 5p.104.
- [49] Lu Y., Fan D. and Zhang Z., “Theoretical and experimental determination of bandwidth for a two-axis fast steering mirror” *Optik*, 2013; 124(16), pp.2443-2449,  
DOI : <https://doi.org/10.1016/j.ijleo.2012.08.023>
- [50] Kluk D.J., Boulet M.T. and Trumper D.L., “A high-bandwidth, high-precision, two-axis steering mirror with moving iron actuator”, *Mechatronics*, 2012.; 22(3), pp.257-270,  
DOI : <https://doi.org/10.1016/j.mechatronics.2012.01.008>

- [51] Wei C., Sihai C., Xin W. and Dong L., “A new two-dimensional fast steering mirror based on piezoelectric actuators”, In 4th IEEE International Conference on Information Science and Technology, 2014; (pp. 308-311). IEEE,  
DOI : [10.1109/ICIST.2014.6920390](https://doi.org/10.1109/ICIST.2014.6920390)
- [52] Chen G., Liu P. and Ding H., “Structural parameter design method for a fast-steering mirror based on a closed-loop bandwidth”, *Frontiers of Mechanical Engineering*, 2020;15(1), pp.55-65,  
DOI : <https://doi.org/10.1007/s11465-019-0545-y>
- [53] Tang T., Huang Y., Fu C. and Liu S., “Acceleration feedback of a CCD-based tracking loop for fast steering mirror”, *Optical Engineering*, 2009; 48(1), p.013001  
DOI : <https://doi.org/10.1117/1.3065500>
- [54] Tang T., Ge R., Ma J. and Fu C., “Compensating for some errors related to time delay in a charge-coupled-device-based fast steering mirror control system using a feedforward loop” *Optical Engineering*, 2010; 49(7), p.073005,  
DOI : <https://doi.org/10.1117/1.3467453>
- [55] Tian J., Yang W., Peng Z. and Tang T., “Inertial sensor-based multiloop control of fast steering mirror for line of sight stabilization”, *Optical Engineering*, 2016; 55(11), p.111602,  
DOI : <https://doi.org/10.1117/1.OE.55.11.111602>
- [56] Sweeney M.N., Rynkowski G.A., Ketabchi M. and Crowley R., “Design considerations for fast-steering mirrors (FSMs)”, In *Optical Scanning 2002* (Vol. 4773, pp. 63-73). SPIE,  
DOI : <https://doi.org/10.1117/12.469197>
- [57] Wang G., Wang Y., Zhou H., Bai F., Chen G. and Ma J., “Comprehensive approach to modeling and identification of a two-axis piezoelectric fast steering mirror system based on multi-component analysis and synthesis”, *Mechanical Systems and Signal Processing*, 2019.;127, pp.50-67,  
DOI : <https://doi.org/10.1016/j.ymssp.2019.03.002>
- [58] Watkins R.J., Chen H.J., Agrawal B.N. and Shin Y.S., “Optical beam jitter control”, In *Free-Space Laser Communication Technologies XVI*, 2004, June; (Vol. 5338, pp. 204-213). SPIE,  
DOI : <https://doi.org/10.1117/12.529457>
- [59] Somaschini R., Bianchi G., Scaccabarozzi D., Cinquemani S., Zocchi F. and Marioni F., “Characterization of commercial fast steering mirrors for space application”, *IEEE 5th International Workshop on Metrology for AeroSpace (MetroAeroSpace)* (pp. 468-472). IEEE, 2019, June.  
DOI : [10.1109/MetroAeroSpace.2019.8869640](https://doi.org/10.1109/MetroAeroSpace.2019.8869640)
- [60] Zhang S., Zhang B., Li X., Wang Z. and Qian F., “A method of enhancing fast steering mirror’s ability of anti-disturbance based on adaptive robust control”, *Mathematical Problems in Engineering*, 2019;  
DOI: <https://doi.org/10.1155/2019/2152858>
- [61] Xia Y.X., Bao Q.L. and Wu Q.Y., “Internal model control of a fast steering mirror for electro-optical fine tracking”, In *High-Power Lasers and Applications 2010*, November; (Vol. 7843, pp. 119-125). SPIE.  
DOI : <https://doi.org/10.1117/12.868901>
- [62] Chen N., Potsaid B., Wen J.T., Barry S. and Cable A., “Modeling and control of a fast-steering mirror in imaging applications”, In *IEEE International Conference on Automation Science and Engineering 2010*, August. (pp. 27-32). IEEE

DOI : [10.1109/COASE.2010.5584424](https://doi.org/10.1109/COASE.2010.5584424)

- [63] Song H., Zhang J.H., Yang P., Huang H.C., Zhan S.Y., Liu T.J., Guo Y.L., Wang H.Z., Huang H., Mu Q.Q. and Fang M.F., “Modeling of a dynamic dual-input dual-output fast steering mirror system”, *Frontiers of Information Technology & Electronic Engineering*, 2017;18(10), pp.1488-1498  
DOI : <https://doi.org/10.1631/FITEE.1601221>
- [64] Wang G. and Rao C., “Adaptive control of piezoelectric fast steering mirror for high precision tracking application”, *Smart Materials and Structures*, 2015; 24(3), p.035019.
- [65] Csencsics E. and Schitter G., “System design and control of a resonant fast steering mirror for Lissajous-based scanning”, *IEEE/ASME Transactions on Mechatronics*, 2017; 22(5), pp.1963-1972,  
DOI : [10.1109/TMECH.2017.2722578](https://doi.org/10.1109/TMECH.2017.2722578)
- [66] Csencsics E., Schlarp J. and Schitter G., “High-performance hybrid-reluctance-force-based tip/tilt system: design, control, and evaluation”. *IEEE/ASME Transactions on Mechatronics*, 2018; 23(5), pp.2494-2502,  
DOI : [10.1109/TMECH.2018.2866272](https://doi.org/10.1109/TMECH.2018.2866272)
- [67] Tapos F.M., Edinger D.J., Hilby T.R., Ni M.S., Holmes B.C. and Stubbs D.M., “High bandwidth fast steering mirror”, In *Optomechanics 2005* (Vol. 5877, pp. 60-73). SPIE,  
DOI : <https://doi.org/10.1117/12.617960>
- [68] Di Lin., Yi-Ming W. and Fan Z., “Research on precision tracking on fast steering mirror and control strategy”. In *IOP Conference Series: Earth and Environmental Science*, 2018. Vol. 114, No.1, p.012009.IOP Publishing.
- [69] Hilkert J.M., “Inertially stabilized platform technology concepts and principles”. *IEEE control systems magazine* 2008; 28(1), pp.26-46,  
DOI : [10.1109/MCS.2007.910256](https://doi.org/10.1109/MCS.2007.910256)
- [70] Germann L.M., “Specification of fine-steering mirrors for line-of-sight stabilization systems”, In *Active and Adaptive Optical Components*, 1992 January; (Vol. 1543, pp. 202-212). SPIE,  
DOI : <https://doi.org/10.1117/12.51182>
- [71] Tang T., Tan Y. and Ren G., “Unload control of double fast steering mirrors of the control system”, In *XX International Symposium on High-Power Laser Systems and Applications 2014*; (Vol. 9255, pp. 851-854). SPIE,  
DOI : <https://doi.org/10.1117/12.2064900>
- [72] Han W., Shao S., Zhang S., Tian Z. and Xu M., “Design and modeling of decoupled miniature fast steering mirror with ultrahigh precision”, *Mechanical Systems and Signal Processing*, 167, 2022; p.108521,  
DOI : <https://doi.org/10.1016/j.ymssp.2021.108521>
- [73] McKelvey T., Akçay H. and Ljung L., 1996. “Subspace-based multivariable system identification from frequency response data”, *IEEE Transactions on Automatic control*, 41(7), pp.960-979 DOI: 10.1109/9.508900
- [74] Abid M., Yu J., Xie Y., Salam A. “Conceptual design, modelling and compliance characterization of a novel 2-DOF rotational pointing mechanism for fast steering mirror”, *Chinese Journal of Aeronautics*, Volume 33, Issue 12, December 2020, Pages 3564-357 <https://doi.org/10.1016/j.cja.2020.03.032>
- [75] Cao K., Hao G., Liu Q., Tan L., Ma J., “Hysteresis Modelling and Compensation of Fast Steering Mirrors with Hysteresis Operator Based Back Propagation Neural Networks”, *Micromechanics* 2021,12(7),732  
<https://doi.org/10.3390/mi12070732>

- [76] Zadeh L.A. "Fuzzy sets", *Inf Control* 1965;8:338–53
- [77] Mamdani E. "Applications of fuzzy algorithms for simple dynamic plant", *Proc IEE* 1974;121(12):1585–8.
- [78] Palm R. "Sliding mode fuzzy control", In: *Proc fuzz IEEE*, San Diego, CA; 1992. p.519–26.
- [79] Savran A, Kahraman G. "A fuzzy-model-based adaptive PID controller design or nonlinear and uncertain processes", *ISA Trans* 2014; 53:280–8.
- [80] Lee CC. "Fuzzy logic in control systems: fuzzy logic controller", *IEEE Trans Syst Man Cybern* 1990; 20:404–34.
- [81] Shen D, Sun W, Sun Z. "Adaptive PID formation control of non-holonomic robots without leader's velocity information". *ISA Trans* 2014; 53:474–80
- [82] Abdo M.M, Vali A.R, Toloei A.R, Arvan M.R. "Stabilization loop of at two axes gimbal system using self-tuning PID type fuzzy controller". *ISA Trans* 2014; 53:591–602.
- [83] F. G. Shinsky, *Process Control Systems—Application, Design, and Tuning*. New York: McGraw-Hill, 1998.
- [84] ] He S. Z., Tan S., Xu F. L., and Wang P. Z., "Fuzzy self-tuning of PID controller," *Fuzzy Sets Syst.*, vol. 56, pp. 37–46, 1993 Santos MP, Ferreira JAF. Novel intelligent real-time position tracking system using FPGA and fuzzy logic. *ISA Trans* 2014; 53:402–14.
- [85] Maeda M. and Murakami S., "A self-tuning fuzzy controller," *Fuzzy Sets Syst.*, vol. 51, pp. 29–40, 1992
- [86] Nomura H., Hayashi I., and Wakami N., "A self-tuning method of fuzzy control by decent method," in *Proc. Int. Fuzzy Syst. Assoc.*, Brussels, Belgium, 1991, pp. 155–158.
- [87] Zheng L., "A practical guide to the tuning of proportional and integral (PI)like fuzzy controllers," in *Proc. Fuzz IEEE*, San Diego, CA, Mar. 1992, pp. 633–641.
- [88] Yoshida M., Tsutsumi Y., and Ishida T., "Gain tuning method for design of fuzzy control systems," in *Proc. Int. CoHolland J. H* 1994 *Adaptation in natural and artificial systems. An introductory analysis with application to biology, control and artificial intelligence* (London: Bradford book edition)
- [89] Hayashi S., "Auto-tuning fuzzy PI controller," in *Proc. Int. Fuzzy Syst. Assoc.*, Brussels, Belgium, 1991, pp. 41–44.
- [90] Iwasaki T. and Morita A., "Auto-tuning controller with fuzzy identification," *Proc. Int. Conf. Fuzzy Logic Neural Networks*, Fukuoka, Japan, 1990, pp. 401–404.
- [91] Mudi Rajani K., Pal Nikhil R., "A Robust Self-Tuning Scheme for PI- and PD-Type Fuzzy Controllers". *IEEE Transactions on Fuzzy Systems*, Vol.. 7, NO. 1 February 1999
- [92] Dorf R. C and Bishop R. H. 2011; *Modern control systems* (Prentice Hall)
- [93] Besekersky V. A, Popov E. P. *Theory of Automatic Control Systems* (Saint Petersburg: Professiya Publ) 2003;752 p
- [94] Denisova L. A. "Automatic feed control of steam generator in the power unit of a nuclear power plant: modeling and optimization", *Automation and Remote Control*. 2016; Vol 77 no 6 pp 1084-92
- [95] Blondin MJ, Sanchis Sáez J, Pardalos PM. "Control Engineering from Classical to Intelligent Control Theory—An Overview", *Computational Intelligence and Optimization Methods for Control Engineering Springer Optimization and Its Applications* Vol. 150. Cham: Springer. 2019.  
DOI: 10.1007/978-3-030-25446-9\_1



- [96] Borase RP, Maghade D, Sondkar S, Pawar S. "A review of PID control, tuning methods and applications", *International Journal of Dynamics and Control*, 2021;9(2):818-827
- [97] Lloyds Raja G, Ali A. "New pi-pd controller design strategy for industrial unstable and integrating processes with dead time and inverse response", *Journal of Control, Automation and Electrical Systems*, 2021;32(2):266-280
- [98] Patel VV. "Ziegler-Nichols tuning method. Resonance", 2020;25(10):1385-1397
- [99] Ekinici S, Hekimoğlu B, Kaya S. "Tuning of PID Controller for AVR System Using Salp Swarm Algorithm", *International Conference on Artificial Intelligence and Data Processing (IDAP)*, Malatya, Turkey. 2018. pp. 1-6. DOI: 10.1109/IDAP.2018.8620809
- [100] Senthil Kumar S, Anitha G, "A novel self-tuning fuzzy logic-based PID controllers for two-axis gimbal stabilization in a missile seeker", *International Journal of Aerospace Engineering* 2021:1-12
- [101] Anand A, Aryan P, Kumari N, Raja GL., "Type-2 fuzzy-based branched controller tuned using arithmetic optimizer for load frequency control. *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*", 2022; 44(2):4575-4596
- [102] El-Samahy AA, Shamseldin MA. "Brushless dc motor tracking control using self-tuning fuzzy PID control and model reference adaptive control", *Ain Shams Engineering Journal*. 2018; 9(3):341-352
- [103] Somwanshi D, Bunde M, Kumar G, Parashar G., "Comparison of fuzzy-pid and pid controller for speed control of dc motor using LabVIEW", *Procedia Computer Science*. 2019;152:252-260
- [104] Potnuru D, Mary KA, Babu CS, "Experimental implementation of flower pollination algorithm for speed controller of a bldc motor", *Ain Shams Engineering Journal*. 2019;10(2):287-295
- [105] Bari S, Zehra Hamdani SS, Khan HU, Rehman Mu, Khan H., "Artificial Neural Network Based Self-Tuned PID Controller for Flight Control of Quadcopter", *International Conference on Engineering and Emerging Technologies (ICEET)*, Lahore, Pakistan. 2019. pp. 1-5. DOI: 10.1109/CEET1.2019.8711864
- [106] Rodr'iguez-Abreo O, Rodr'iguez-Res'endiz J, Fuentes-Silva C, Hern'andez-Alvarado R, Falc'on MDCPT. "Self-tuning neural network PID with dynamic response control", *IEEE Access*. 2021;9:65206-65215
- [107] PiezoSystemjena PSH x/2 Series <https://www.piezosystem.com/wp-content/uploads/2022/03/PSH-X-Slash-2-Handling-Instructions.pdf>
- [108] M. Pal, D. Biswas, A. Das, K. Banerjee, B. Dam, "Rapid Modelling of Fast Steering Mirror Assembly from Time Response Data," *Proceedings of Fourth International Conference MCCS19*, pp683-696. Springer 2021  
DOI: 10.1007/978-981-15-5546-6\_57
- [109] Huang, C.T, Chou C.J. , "Estimation of the Under damped Second-Order Parameters from the System Transient", *Ind. Eng. Chem*, 1994, Res.32,228-230, <https://doi.org/10.1021/ie00025a024>
- [110] G. P. Rangaiah, P. R. Krishnaswamy. "Estimating second-order dead time parameters from underdamped process transients" *Chemical Engineering Science*, 1996, 51(7), 1149-1155
- [111] R. Joseph Watkins, Brij N. Agarwal, "Use of Least Means Squares Filter in Control of Optical Beam Jitter", *Journal of Guidance, Control, and Dynamics*, Vol 30, no. 4, (July-August 2007), pp. 1116-1122  
DOI:10.2514/1.26778
- [112] Adegbenro, A., Short, M. and Williams, S., 2019, "Development of a Digital PID-like Adaptive Controller and its Application in HVAC systems", *International Conference on Innovative Applied Energy*.

- [113] M. S. Aftab and M. Shafiq, "Adaptive PID controller based on Lyapunov function neural network for time delay temperature control," 2015 IEEE 8th GCC Conference & Exhibition, 2015, pp. 1-6, doi:10.1109/IEEEGCC.2015.7060094.
- [114] Valenzuela, P., Ebadat, A., Everitt, N. and Parisio, A., "Closed Loop Identification for Model Predictive Control of HVAC Systems: From Input Design to Controller Synthesis", 2019 IEEE Transactions on Control Systems Technology, 28(5), pp.1681-1695.
- [115] Nam Khoa Huynh, Hua Li, Yeng Chai Soh and Wenjian Cai, "Numerical analysis and model-based control of energy recovery ventilator in HVAC system", 2016 World Congress on Sustainable Technologies (WCST).
- [116] Sayehi, I., Touali, O., Bouallegue, B., and Tourki, R., "A comparative study of two kernel methods: Support Vector Regression (SVR) and Regularization Network (RN) and application to a thermal process PT326", 2015 16th International Conference on Sciences and Techniques.
- [117] Saha, S., Rudra, S., Dasgupta, T. and Maitra, M., "Design of a novel adaptive variable structure control law for industrial heat exchanger process representing a class of nonminimum process plant", 2016 IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering (UPCON).
- [118] Ibias J.A.C., Rodriguez J.B.N., Salinas R.I.G, "Generalized Predictive Control Applied to the PT326 Process Trainer", 2021 IEEE XXVIII International Conference on Electronics, Electrical Engineering and Computing (INTERCON).
- [119] Salvador J.A., Ramirez D.R., Alamo T., Pena D.M., "Offset free data-driven control: Application to a process control trainer", 2018 IET Control Theory and Applications <http://dx.doi.org/10.1049/iet-cta.2019.0376>
- [120] B. Ribeiro B., Cardoso A. "A Model-based Neural Network Controller for a Process Trainer Laboratory Equipment", Artificial Neural Nets and Genetic Algorithms. Springer, Vienna. [https://doi.org/10.1007/978-3-7091-6492-1\\_133](https://doi.org/10.1007/978-3-7091-6492-1_133)
- [121] Meena D.C., Devanshu, "A. Genetic algorithm tuned PID controller for process control", 2017 International Conference on Inventive Systems and Control (ICISC) <https://doi.org/10.1109/ICISC.2017.8068639>
- [122] M. A. Unar, D. J. Murray-Smith and S. F. A. Shah, "Design and tuning of fixed structure PID controllers: A survey".
- [123] T. Ray Chaudhuri and L. G. C. Hamey, "From conventional control to intelligent neuro control methods –A survey of the literature", Macquarie University Technical Report No. 95/170C, Department of computing school of MPCE, Macquarie University, New South Wales, Australia, 1995.
- [124] K. J. Åström, C. C. Hang, P. Persson and W. K. Ho, "Towards intelligent PID control", Automatica, 28(1), pp.1-9, 1992.
- [125] W. K. Ho, C. C. Hang and L. S. Kao, "Tuning of PID controllers based on gain and phase margin specifications", Automatica, 31(3), pp. 497-502, 1995.
- [126] P. J. Gawthrop, "Self-tuning PID control structures", CSC report: CSC-96011, 1996.
- [127] D. Driankov, H. Hellendorn and M. Reinfrank, "An Introduction to Fuzzy Control", New York: Springer-Verlag, 1993.
- [128] G. K. I. Mann, , B. G. Hu and R. G. Gasine, "Analysis of direct action fuzzy PID controller structures", IEEE Trans. on Systems, Man, and Cybernetics – Part B: Cybernetics, 29(3), pp. 371-388, 1999.
- [129] P. Govender and V. B. Bajic, "Fuzzy-logic modifier f PID control for processes having long dead-time".

- [130] H. Nomura, I. Hayashi, and N. Wakami, "A self-tuning method of fuzzy control by decent method," in Proc. Int. Fuzzy Syst. Assoc., Brussels, Belgium, 1991, pp. 155–158
- [131] M. Yoshida, Y. Tsutsumi, and T. Ishida, "Gain tuning method for design of fuzzy control systems," in Proc. Int. Conf. Fuzzy Logic Neural Networks, Fukuoka, Japan, 1990, pp. 405–408.
- [132] S. Z. He, S. Tan, F. L. Xu, and P. Z. Wang, "Fuzzy self-tuning of PID controller," Fuzzy Sets Syst., vol. 56, pp. 37–46, 1993.
- [133] T. Iwasaki and A. Morita, "Auto-tuning controller with fuzzy identification", in Proc. Int. Conf. Fuzzy Logic Neural Networks, Fukuoka, Japan, 1990, pp. 401–404.
- [134] M. Maeda and S. Murakami, "A self-tuning fuzzy controller," FuzzySets Syst., vol. 51, pp. 29–40, 1992.
- [135] J. Lee, "On methods for improving performance of PI-type fuzzy logic controllers," IEEE Trans. Fuzzy Syst., vol. 1, pp. 298–301, Nov. 1993.
- [136] S. Hayashi, "Auto-tuning fuzzy PI controller," in Proc. Int. Fuzzy Syst.Assoc., Brussels, Belgium, 1991, pp. 41–44.
- [137] J.H. Holland, "Adaptation in Natural and Artificial System", University of Michigan Press, USA, 1975.
- [138] K.S. Nawaz Ripon, S. Kwong, K.F. Man, "A real-coding jumping gene genetic algorithm (RJGGA) for multiobjective optimization", Information Sciences 177 (2007) 632–654.
- [139] A. Soltoggio, "An enhanced GA to improve the search process reliability in tuning of control systems", in Proceedings of the 2005 Conference Genetic and Evolutionary Computation, GECCO'05, Washington, DC, USA, 2005, pp. 2165–217
- [140] Q.G. Chen, N. Wang, "The distribution population-based genetic algorithm for parameter optimization PID controller", Acta Automatica Sinica 31 (2005) 646–650.
- [141] B. Kristiansson, B. Lennartson, "Robust PI and PID controller including Smith Predictor Structure", American Control Conference (2001) Vol.3 2197-2202
- [142] A. Ingimundarson, T. H'agglund, "Performance Comparison between PID and dead time compensating controllers", Journals of Process Control, 12(8),887-895
- [143] P.O. Larsson, T. Hagglund, "Robustness margins separating process dynamic uncertainties", European Control Conference (ECC), Budapast, 543 - 548. IEEE
- [144] C.Grimholt, S. Skogestad, "Should we forget the Smith Predictor?", IFAC Papers Online 51-4 (2018) 769 - 774