ENHANCED SOFT COMPUTING TECHNIQUES FOR ROBUST CONTROLLER DESIGN

Ph.D. THESIS

by

PRASHANT JAICHANDRAO GAIDHANE

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE, INDIA (247667)
MARCH,2019

ENHANCED SOFT COMPUTING TECHNIQUES FOR ROBUST CONTROLLER DESIGN

A THESIS

Submitted in partial fulfilment of the requirements for the award of the degree

of

DOCTOR OF PHILOSPHY

in

ELECTRONICS AND COMMUNICATION ENGINEERING

by

PRASHANT JAICHANDRAO GAIDHANE

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE, INDIA (247667)
MARCH, 2019

INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **"ENHANCED SOFT COMPUTING TECHNIQUES FOR ROBUST CONTROLLER DESIGN"** in partial fulfilment of the requirements for the award of the Degree of Doctor of Philosophy and submitted in the Department of Electronics and Communication Engineering of the Indian Institute of Technology Roorkee, Roorkee is an authentic record of my own work carried out during a period from January, 2016 to March, 2019 under the supervision of Dr. P. M. Pradhan, Assistant Professor and Dr. M. J. Nigam, Professor (Retired), Department of Electronics and Communication Engineering, Indian Institute of Technology Roorkee, Roorkee, India.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other Institution.

(PRASHANT JAICHANDRAO GAIDHANE)

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

(P. M. Pradhan)
Supervisor

(M. J. Nigam)
Supervisor

The Ph.D. Viva-Voce Examination of **PRASHANT JAICHANDRAO GAIDHANE**, Research Scholar, has been held on ……..

Chairman, SRC **Signature of External Examiner**

This is to certify that the student has made all the corrections in the thesis.

Signature of Supervisors **Head of the Department**

Dated:

To

my beloved father

Late Shri. Jaichandrao Gomaji Gaidhane

Acknowledgements

At first, I intend to thank my supervisors Dr. Madhav J. Nigam and Dr. Pyari Mohan Pradhan for being the guiding light in providing the required inputs for my PhD work. I express my gratitude towards them for having confidence in me, sometimes more than I believe in myself. I feel grateful to share that Nigam sir provides me his personal computer and all necessary things with highest priority. The constructive opinions and suggestions about my work has been substantial in improving the quality of the work. I would like to sincerely thank them for providing me the constant support and motivation during the whole journey at IITR.

My gratitude towards my research committee members Prof. D. Ghosh, Prof. Durga Toshniwal, Prof. Meenakshi Rawat for assessing me at various stages of PhD and providing me their valuable opinions which have helped in improving the research work. I would like to thank my friends and labmates Dr. Anupam Kumar, Mr. Avadh Kishor, Dr. Meenakshi Awasthi, Dr. Vrince Vimal, Dr. Zaheer Khan, Prof. Sunil Jatti, Prof. Sandeep Shingade and other friends at IIT Roorkee for their valuable support during different stages of my research work.

I am very thankful to Principal Dr. R. P. Borkar and other colleagues at GCOE Jalgaon for their help and inspiration in this achievement. Over and above, I am always obliged to Mr. Sandeep Acharya (Editor, Loksatta, Mumbai) for his selfless support to get permission in time.

Words cannot express how grateful I am for having constant sources of love, concern, support and strength by my Aai and brother Annabhau. Most prominently, I would like to thank my sisters, brother-in-law, sister-in law, in-laws, and whole family for supporting me emotionally and spiritually throughout this journey. Specially, this journey could not be possible without support of my wife Bhavna and lovely daughter Tanvi, who equally suffer and enjoy these four years at IITR.

At the onset, I pay my sincere gratitude towards the Almighty for bestowing upon me with his blessings to pursue PhD at my dream institute, Indian Institute of Technology. I strongly believe that without the strength, courage and determination provided to me by the Almighty, I would have never been able to write this PhD Thesis. Finally, I would like to pay respect to my father late Jaichand G. Gaidhane Sir to enlighten the spark of knowledge in me which drives me to this stage and inspires to rise further.

Prashant Jaichandrao Gaidhane

List of Abbreviations

ABC	: Artificial Bee Colony Algorithm
AI	: Artificial Intelligence
CEC	: Congress on Evolutionary Computation
DOF	: Degree of Freedom
EA	: Evolutionary Algorithm
FE	: Function Evaluations
FLC	: Fuzzy Logic Controller
FLS	: Fuzzy Logic System
FOFPID	: Fractional Order fuzzy PID
FOPID	: Fractional Order PID
FOPTD	: First-order Plus Time Delay
FOU	: Footprint of Uncertainty
FPID	: Fuzzy PID
FP-PID	: Fuzzy Pre-Compensated PID
FS	: Fuzzy Set
GWO	: Grey Wolf Optimizer
GWO-ABC	: Grey Wolf Optimizer and Artificial Bee Colony based Hybrid Algorithm
IAE	: Integral of the Absolute Error
ISE	: Integral of the Square Error
IT2-FLC	: Interval Type-2 Fuzzy Logic Controller
IT2-FLS	: Interval Type-2 Fuzzy Logic System
IT2-FPID	: Interval Type-2 Fuzzy PID
IT2FP-PID	: Interval Type-2 Fuzzy Precompensated PID
IT2-FS	: Interval Type-2 Fuzzy Sets
IT2-MF	: Interval Type-2 Membership Functions
ITAE	: Integral of the Time Absolute Error
ITSE	: Integral of the Time Square Error
LMF	: Lower Membership Function
MA	: Metaheuristic Algorithms
MF	: Membership Function
MIMO	: Multi-Input Multi-Output

MLS	: Magnetic Levitation System
MOO	: Multi-Objective Optimization
MOEA	: Multi-Objective Evolutionary Algorithms
NSGA-II	: Fast and Elitist Non-Dominated Sorting Genetic Algorithm-II
OBL	: Opposition-based Learning
PID	: Proportional-Integral-Derivative
PSO	: Particle Swarm Optimization
SOO	: Single Objective Optimization
SISO	: Single-Input Single-Output
SOPTD	: Second-order Plus Time Delay
T1-FLC	: Type-1 Fuzzy Logic controller
T1-FLS	: Type-1 Fuzzy Logic System
T1FO-FPID	: Type-1 Fractional Order Fuzzy PID
T1FP-PID	: Type-1 Fuzzy Precompensated PID
T1-FS	: Type-1 Fuzzy Sets
T1-MF	: Type-1 Membership Functions
T2-FLC	: Type-2 Fuzzy Logic Controller
T2-FLS	: Type-2 Fuzzy Logic System
T2-FS	: Type-2 Fuzzy Sets
T2-MF	: Type-2 Membership Functions
TR	: Type Reducer
TSK	: Takagaki-Sugeno-Kang
UMF	: Upper Membership Function

List of Publications

International Peer-Reviewed Journals

1. **P. J. Gaidhane** and M. J. Nigam, "A hybrid grey wolf optimizer and artificial bee colony algorithm for enhancing the performance of complex systems", *Journal of Computational Science* , vol. 27, no. 7, pp. 284-302, Jul 2018. Elsevier, SCI, IF. - 1.925.

2. **P. J. Gaidhane**, M. J. Nigam, A. Kumar, and P. M. Pradhan , "Design of interval type-2 fuzzy precompensated PID controller applied to two-DOF robotic manipulator with variable payload", *ISA Transactions*, Dec 2018. Elsevier, SCI, IF. - 3.370 (In Press).

3. **P. J. Gaidhane** and M. J. Nigam, " A rational cooperative foraging based grey wolf optimizer and its application to optimize complex systems", *Applied Soft Computing*, Elsevier, (Under Review).

International Conferences

4. **P. J. Gaidhane**, A. Kumar, and M. J. Nigam, "Tuning of two-DOF-FOPID controller for magnetic levitation system: A multi-objective optimization approach", *in Proceedings of Computer Application in Electrical Engineering - Recent Advances (CERA)*, Oct 2017, pp. 479-484.

5. **P. J. Gaidhane** and M.J. Nigam, "Multi-Objective robust design and performance analysis of two-DOF-FOPID controller for magnetic levitation system," *in the Proceedings of 14th India Council International Conference (INDICON)*, Dec 2017, pp. 1-6.

6. A. Kumar, **P. J. Gaidhane**, and V. Kumar "A non-linear fractional order PID controller applied to redundant robot manipulator", *in Proceedings of Computer Application in Electrical Engineering - Recent Advances (CERA)*, Oct 2017, pp. 527-532.

7. A. Kumar, V. Kumar, and **P. J. Gaidhane**, "Optimal design of fuzzy fractional order $PI^{\lambda}D^{\mu}$ controller for redundant robot", *Procedia Computer Science* vol. 125, pp. 442-448, 2018.

Abstract

Last few decades have seen the rapid advances and diffusion of technologies in science, engineering, robotics, biomedical, economics and other fields. Diverse technologies are incorporated in industrial processes to increase the efficiency, which in turn alleviated the complexity of the systems. Highly specialized skills and expert knowledge are required to design, develop, operate, and control such complex systems. Along with this, the uncertainties caused by inaccurate modeling, external disturbances, and variations of working conditions influence the system performance. Eventually, design and development of robust control system with intelligent computational tools is essentially required to get desired performance in efficient manner. In recent years, the concept of soft computing based intelligent control has emerged as an efficient tool to enhance the existing non-linear, optimal, adaptive, and stochastic control methods. The intelligent control can be achieved through the involvement of various artificial intelligence (AI) and soft computing approaches utilized for closed-loop feedback control to improve system performance, reliability, and efficiency. The major soft computing techniques are fuzzy logic system, metaheuristic algorithms (MAs), chaos theory, neurocomputing, and probabilistic reasoning. Various soft computing techniques and their fusions are commonly used to enhance the intelligent control tools by incorporating human expert knowledge in computing processes.

The field of soft computing is always growing by contributions from the large community of researchers and provides an exceptional opportunity to advance its methodology and applications. Consequently, there is a great scope and motivation to ameliorate, design, hybridize, and apply these techniques. This fact motivates us to present some significant improvements and novel contributions to major components of soft computing like fuzzy logic system and MAs. The main aim of the overall work presented in this thesis is to enhance the soft computing techniques to improve the performance of the control system design. The complete work in this thesis is distinguished by four research objectives, given as (a) Design efficient optimization algorithm for enhancing the performance of complex systems, (b) Performance analysis of the proposed algorithm for optimization of different controller design problems, (c) Design of interval type-2 fuzzy precompensated PID (IT2FP-PID) controller applied to 2-link robotic manipulator with variable payload, and (d) Constrained multi-objective optimization (MOO) approach for robust controller design and performance analysis.

The tuning of controllers is considered as a high-dimensional, complex, multimodal numerical optimization problem, as many locally optimal solutions can be obtained for different combinations of the parameter values. Thus, it is always a challenging task for designers to get the optimal tuning parameters. MAs are extensively considered for solving such control system design problems to get the best performance and robust response. Some common deficiencies faced by the majority of population-based MAs are lack of exploration ability, slow and premature convergence behaviour, and stagnation to local optima. Considering the above-stated problems,

important features of two well established MAs, grey wolf optimizer (GWO) and artificial bee colony algorithm (ABC), are hybridized to develop an improved GWO-ABC algorithm as the first objective of the thesis. In GWO-ABC algorithm, information sharing property of employed bees in ABC is adopted with conventional GWO algorithm to comprehend the benefits of both the algorithms. A new population initialization strategy is introduced to get widespread range solutions. These strategies are incorporated to overcome the shortcomings of the conventional GWO algorithm by improving exploration capability, convergence rate, and reduce the chances of entrapment at local optima. The performance of the GWO-ABC algorithm is validated through extensive experimental analysis of 27 benchmark functions and comparisons with 5 other standard intelligent algorithms.

After successfully substantiating that GWO-ABC algorithm is an efficient algorithm for solving complex test functions, it is applied to control system design problem for linear and non-linear test bench process plants. Here, the GWO-ABC algorithm is applied to minimize the objective function so that optimal time-domain specifications could be achieved. All the design requirements like low overshoot, better rise time, faster settling time, minimum steady-state error, and performance index are evaluated and compared to other state-of-the-art algorithms. Further, the conventional GWO algorithm is improved by incorporating the communication signalling strategy used in cooperative foraging of wolves. The leadership hierarchy approach and communicating behaviours are merged to present improved cooperative foraging based GWO (CFGWO) algorithm. New acceleration coefficient is proposed to balance the exploration and exploitation behaviour throughout the iterations. The proposed algorithm is examined on a real-world optimization problem of controller designing for trajectory tracking problems of a 2-link robotic manipulator with payload at tip. The comparative graphs of trajectory tracking performance, the path traced by the end-effector, and X and Y coordinate versus time variations against their desired reference curves are plotted. Also, the plots of position errors and controller output for both the links are also presented.

As mentioned earlier, many of the real-world industrial processes are influenced by large amounts of uncertainties due to dynamic unstructured environments. The type-2 fuzzy logic controllers with type-2 fuzzy sets are highly recognized to deliver a satisfactory performance in the face of uncertainty and imprecision than their type-1 counterparts. In order to establish its applicability, an efficient IT2FP-PID controller is presented for trajectory tracking of a 2-link robotic manipulator with variable payload. The controller is comprised of unique features of interval type-2 fuzzy precompensated controller cascaded with a conventional PID controller. The fuzzy logic controller (FLC) based precompensator is incorporated to regulate the control signal to compensate the undershoots and overshoots in the system output when the system has unknown non-linearities. Tuning of control parameters and the antecedent MF structures emerged as a complex, high-dimensional, and constrained optimization problem. Hence, a systematic strategy for optimizing the controller parameters along with scaling factors and the antecedent MF parameters for minimization of performance metric integral time absolute error ($ITAE$) is

presented. The structures of MFs are also optimized to get maximum benefits of footprint of uncertainty (FOU) in type-2 FLC. Prominently, GWO-ABC algorithm is utilized for solving this high-dimensional constrained optimization problem. In order to witness effectiveness, the performance is compared with type-1 fuzzy precompensated PID (T1FP-PID), fuzzy PID (FPID), and conventional PID controllers. The efficacy of the proposed controller is also validated through exhaustive robustness analysis in presence of distinct non-linear dynamics such as (i) payload variations, (ii) model uncertainties, (iii) disturbance in signals, and (iv) random noise at feedback path. After experimental outcome, it is inferred that IT2FP-PID controller outperforms others and can be adopted as a viable alternative for controlling non-linear complex systems with higher uncertainties. As a whole, the work in this objective manifests that (a) additional tuning parameters provide extra degree of freedom (DOF) to get better performance in optimal controller design, (b) in case of IT2-FLC, the systematic strategy to optimize the shapes of MFs derive maximum benefit of FOU to handle uncertainty (c) the proposed IT2FP-PID controller revealed as viable alternative to control complex non-linear systems with high uncertainties, (d) GWO-ABC algorithm can efficiently solve the low- and high-dimensional constrained optimization problems.

The last part of the thesis is dedicated to investigate the applicability of MOO approach for tuning of controller parameters for multi-variable, constrained, and complex control systems. In the control system design, minimization of performance error indices for set-point and disturbance are considered as conflicting objective functions with various constraints. A simple design and parameter tuning strategy for 2-DOF fractional order PID (2-DOF-FOPID) controller is presented using MOO approach. A fast and elitist non-dominated sorting genetic algorithm (NSGA-II) with constraint handling methodology is utilized and the sensitivity function is used as a constraint. The major robustness investigations are carried for minimization of integrated absolute error (*IAE*) for both set-point tracking and external disturbance rejection. The comparative performance evaluation is assessed against equivalent counterparts and found that the proposed 2-DOF-FOPID controller performs with superior results.

Contents

Acknowledgements i
Abbreviations iii
List of Author's Publications v
Abstract vii
List of Figures xv
List of Tables xix

1 Introduction **1**
1.1 Basics of Control System Design 2
1.1.1 Structure of Control System 2
1.1.2 Challenges in Controller Design 3
1.2 Soft Computing Techniques in Control System Design 5
1.2.1 Fuzzy Logic based Control System 5
1.2.2 Metaheuristic Algorithms (MAs) 6
1.3 Motivation 7
1.4 Research Objectives and Thesis Contributions 8
1.5 Organization of the Thesis 10

2 Literature Survey and Preliminaries **11**
2.1 Enhanced Techniques in Controller Design 11
2.1.1 Evolution of Control Strategies 11
2.1.2 Type-1 Fuzzy Logic Systems 13
2.1.3 Type-2 Fuzzy Logic Systems 14
2.1.4 Fractional Order System 17
2.2 Techniques to Enhance MAs 19
2.2.1 Hybridization 20
2.2.2 Chaotic Mapping 21
2.2.3 Opposition-based Learning 21
2.3 Time-domain Performance Indices 22
2.4 Problems under Study 23
2.4.1 Robotic Manipulator 23

2.4.2 Linear and Non-linear Benchmark Plants. 25
2.4.3 Magnetic Levitation System. 25
2.5 Concluding Remarks 25

3 Novel Hybrid GWO-ABC Algorithm for Complex Systems 27
3.1 Introduction 27
3.2 Overview of Conventional GWO and ABC Algorithms 28
3.2.1 An Overview of GWO Algorithm 28
3.2.2 An Overview of ABC Algorithm 30
3.2.3 Chaotic Mapping and OBL Strategy. 31
3.3 Proposed Hybrid GWO-ABC Algorithm 31
3.3.1 Motivation for Hybridization 33
3.3.2 Structure of the Proposed GWO-ABC Algorithm 33
3.4 Simulation Results and Discussion 35
3.4.1 Performance Evaluation on Test Functions 35
3.4.2 Parameter Settings 37
3.4.3 Statistical Analysis (Wilcoxon Rank Sum Test) 37
3.4.4 Exploitation Analysis (Results for Unimodal Test Functions) 39
3.4.5 Exploration Analysis (Results on Multimodal Test Functions) 41
3.4.6 Convergence Analysis 41
3.4.7 Results for Composite Test Functions 42
3.5 Concluding Remarks 47

4 Optimization of Different Controller Design Problems 49
4.1 Introduction 49
4.2 Implementation of GWO-ABC on Test-bench Process Plants 50
4.2.1 FOPID Controller 50
4.2.2 Problem Definition 51
4.2.3 Performance Evaluation on Test-bench Process Plants 51
4.2.4 Simulation Results and Discussion 52
4.3 Proposed CFGWO Algorithm 58
4.3.1 Animal's Behaviour in Cooperative Foraging 58
4.3.2 Motivation from Cooperative Foraging 60
4.3.3 New Acceleration Coefficient 60
4.3.4 Structure of the Proposed CFGWO Algorithm 60
4.3.5 Performance Evaluation on Test Functions 64
4.4 Implementation of CFGWO for Controller Tuning Problem 66
4.4.1 FO-FPID Controller 66
4.4.2 Problem Definition 68
4.4.3 Simulation Results and Discussion 70

4.5 Concluding Remarks . . . 73

5 Design of Interval Type-2 Fuzzy Precompensated PID Controller 75
5.1 Introduction . . . 75
5.2 Design and Optimization of IT2FP-PID Controller . . . 76
5.2.1 Proposed IT2FP-PID Controller . . . 76
5.2.2 Proposed Optimization Strategy . . . 77
5.2.3 Implementation of IT2FP-PID Controller on Robotic Manipulator . . . 79
5.3 Simulation Results and Discussion . . . 83
5.3.1 Problem Definition . . . 84
5.3.2 Parameter Settings . . . 84
5.3.3 Results and Discussion . . . 84
5.4 Robustness Analysis . . . 90
5.4.1 Impact of Model Uncertainties . . . 90
5.4.2 Rejection of External Disturbances . . . 90
5.4.3 Suppression of Random Noise . . . 92
5.5 Concluding Remarks . . . 96

6 Constrained Multi-objective Optimization Approach for Control System Design 97
6.1 Introduction . . . 97
6.2 Multi-Objective Optimization . . . 98
6.3 Mathematical Model of Magnetic Levitation System . . . 99
6.4 Implementation of Multi-Objective Algorithm . . . 101
6.4.1 Non-dominated Sorting Genetic Algorithm-II . . . 103
6.4.2 Problem Definition . . . 103
6.5 Simulation Results and Discussion . . . 105
6.6 Concluding Remarks . . . 109

7 Conclusion and Future Scope 111
7.1 Conclusion . . . 111
7.2 Future Scope . . . 114

References 115

A Robotic Manipulator and Benchmark Functions 129
A.1 2-link Robotic Manipulator with Payload at Tip . . . 129
A.2 Benchmark Test Functions . . . 131

List of Figures

1.1 General structure of control system. 2

2.1 Structure of (a) T1-FLS, (b) T1-FS, (c) Membership functions used in T1-FLS. . 13
2.2 Structure of (a) IT2-FLS, (b) IT2-FS, (c) Membership functions used in IT2-FLS. 14
2.3 Illustrations of the general framework of consequent MF used in this study. . . . 17
2.4 FOPID controller region based on the values of fractional order terms (λ and μ). 18
2.5 Illustrations of point $\mathbf{X}$ and its corresponding opposite $\mathbf{X}^*$ according to OBL. . . 22
2.6 Model of 2-link robotic manipulator with payload at tip. 24

3.1 Flowchart of the proposed GWO-ABC algorithm. 32
3.2 Comparative illustrations of initial population by the proposed population initialization scheme against random distribution scheme. 35
3.3 3D plots of some benchmark functions. 36
3.4 Comparison of convergence curves of proposed GWO-ABC and other algorithms obtained for some of the unimodal and composite function. 45
3.5 Comparison of convergence curves of proposed GWO-ABC and other algorithms obtained for some of the multimodal and fixed-dimension multimodal functions. . 46

4.1 Design scheme of FOPID controller applied to system G(s). 51
4.2 Step response of first-order system with time delay. 53
4.3 Step response of non-linear system with time delay. 54
4.4 Step response of second-order stable linear system with time delay. 55
4.5 Step response of non-minimum phase system. 56
4.6 Comparative illustrations of variation in *ITAE* for optimized FOPID controllers applied to test-bench process plants. 57
4.7 Cooperative foraging and hunting behaviour of grey wolves by (a) Leadership based hunting approach, and (b) Cooperative signalling behaviour in the pack through (i) Vocalisation and (ii) Body language (Credit: National Park Service) [128]. 59
4.8 Behaviour of new acceleration coefficient and parameter a against iterations. . . . 61
4.9 Flowchart of the proposed CFGWO algorithm. 62

4.10 Comparison of convergence curves of proposed CFGWO and other algorithms obtained for some of the test functions. 65
4.11 Design scheme of FO-FPID controller applied to system $G(s)$. 67
4.12 Illustration of consequent MF used for FO-FPID controller. 68
4.13 Surface plot for rule base defined in Table 4.7. 69
4.14 Design scheme of FO-FPID controller applied to 2-link of robotic manipulator. . 69
4.15 Comparative illustrations of variation in $ITAE$ for 2-link robotic manipulator applied with FOPID controllers optimized by different algorithms. 71
4.16 Comparison of convergence curves of CFGWO and other algorithms obtained for FO-FPID controllers. 71
4.17 Various comparative illustrations demonstrating different performances of 2-link robotic manipulator with payload obtained by optimal FO-FPID controller. . . . 72

5.1 Design scheme of the proposed IT2FP-PID controller applied to system $G(s)$. . . 76
5.2 General structures of antecedent MFs for the inputs of (a) T1-FLC and (b) IT2-FLC applied to both the links. 78
5.3 Illustration of payload variations at the tip. 79
5.4 Constrained handling scheme proposed for (a) T1-FLC and (b) IT2-FLC. 80
5.5 Design scheme of FPID controller applied to 2-link robotic manipulator. 81
5.6 Design scheme of T1FP-PID and the proposed IT2FP-PID controllers applied to robotic manipulator. 82
5.7 Surface plot for rule base defined in Table 5.3. 83
5.8 Illustration of optimized antecedent MFs applied to T1FP-PID controller. 86
5.9 Illustration of optimized antecedent MFs applied to IT2FP-PID controller. 87
5.10 Comparative illustrations of different performances of 2-link robotic manipulator with variable payload. 88
5.11 Comparative illustrations of variation in $ITAE$ for 2-link robotic manipulator by different controllers. 89
5.12 Noise profile. 89
5.13 Comparative illustrations of different performances of 2-link robotic manipulator with disturbance $2\,sin\,(50t)$ in both the links. 94
5.14 Comparative illustrations of different performances of 2-link robotic manipulator with noise in both the links. 95

6.1 Schematic of multi-objective optimization procedure. 98
6.2 A multi-objective optimization design procedure for control system problems. . . 99
6.3 Model of magnetic levitation system. 100
6.4 Design scheme of 2-DOF FOPID controller applied to magnetic levitation system. 102
6.5 Illustration of maximum sensitivity function M_s, phase margin ϕ_m, and gain margin A_m. 104

6.6 Pareto-optimal front obtained by MOO for 1-DOF-FOPID controller applied to MLS. 106
6.7 Pareto-optimal front obtained by MOO for 2-DOF-FOPID controller applied to MLS. 106
6.8 Comparative illustration of step response. 107
6.9 Comparative illustration of step disturbance response. 107
6.10 Comparative illustration of response for variable input signal. 108
6.11 Comparative illustration of step response for disturbance $d = -0.1$ at $t = 1$ s. . . 108

List of Tables

3.1 Parameter settings of different algorithms used in this study. 37
3.2 *p*–values and *grades* obtained by statistical Wilcoxon rank-sum test with 1% significance level. 38
3.3 Comparison of results obtained for the unimodal benchmark functions. 39
3.4 Comparison of results obtained for the multimodal benchmark functions. 40
3.5 Comparison of results obtained for the fixed-dimension multimodal benchmark functions. 43
3.6 Comparison of results obtained for the composite benchmark functions 44

4.1 Optimal controller parameters and comparative results for the first-order system with time delay. 53
4.2 Optimal controller parameters and comparative results for the non-liner system with time delay. 54
4.3 Optimal controller parameters and comparative results for the second-order stable linear system with time delay. 55
4.4 Optimal controller parameters and comparative results for the non-minimum phase system. 56
4.5 Comparison of results obtained for some benchmark functions. 64
4.6 *p*–values and *grades* obtained by statistical Wilcoxon rank-sum test with 1% significance level. 65
4.7 Rule base for FO-FPID controller. 68
4.8 Optimal controller parameters and comparative results of the FO-FPID controller. 70

5.1 Initial upper and lower limits of antecedent MF parameters used in IT2-FLC. . . 81
5.2 Overall tuning parameters of PID, FPID, T1FP-PID, and IT2FP-PID controllers. 82
5.3 Rule base for all FLC used in this study. 83
5.4 Optimal controller parameters and comparative results of the IT2FP-PID controller and others. 85
5.5 Comparative *ITAE* values obtained by different controllers for variations in modelling parameters. 91
5.6 Comparative *ITAE* values obtained by different controllers for variable disturbance. 92

5.7 Comparative *ITAE* values obtained by different controllers in presence of random noise. 93

6.1 Optimal parameters for different controllers. 105
6.2 *IAE* values of controllers. 109

A.1 Description and values of different robot parameters used in this study. 129
A.2 Descriptions of the composite benchmark functions. 131
A.3 Descriptions of the unimodal and multimodal benchmark functions. 132
A.4 Descriptions of the fixed dimension multimodal benchmark functions. 133

Chapter 1

Introduction

Control system design has become an essential and fundamental domain of today's modern industrial automation system. Control requirements and designs are typically based on the configuration, specifications, identification of system under control, and the desired system goals. The control goals include faster and stable response that is able to retain acceptable performance levels under major unexpected uncertainties. However, most of the real-world systems are complex, non-linear, ill-structured, and have excessive parameter variations. As the precise mathematical models of such systems are not available, designing a proper controller for such systems is a challenging task and requires modern techniques, highly specialized skills, and expert knowledge. In previous years, classical control theory and conventional proportional-integral-derivative (PID) controller schemes were applied to control the systems. However, it has been observed in many studies that classical control strategies generally do not work well for high-order complex systems [1].

In last few decades, remarkable progress has been made in both the theory and application of all important areas of control systems. Recently, intelligent control has emerged as one of the efficient tools to deal with the increasing complexity of various processes in variety of engineering and other applications. Intelligent control can be achieved through proper understanding of system under control and involvement of various soft computing approaches. The principal constituents of soft computing are fuzzy logic, evolutionary optimization algorithms, chaos theory, neurocomputing, and probabilistic reasoning [2]. Soft computing constituents and their hybrid techniques are commonly used to incorporate human expert knowledge in computing processes for intelligent control. Knowing the potential of these strategies, many researchers proposed various model-based intelligent control strategies using different soft computing techniques. These design approaches involve various phases such as modeling, analysis, simulation, implementation and verification. Many of these conventional and model-based methods have found their way into practice and provide satisfactory and robust solutions to the spectrum of complex systems under various uncertainties. These techniques render a greater degree of autonomy to uncertain complex systems than the available usual control schemes. Thus, intelligent controllers play an important role in robotic systems, aerospace, chemical industry, and power plants.

This chapter further explains the general structure of the control system and associated challenges in its design. Various soft computing techniques used in control system design are discussed in Section 1.3. Motivation for overall work is presented in Section 1.4. The thesis contribution is briefed in Section 1.5 followed by organization of thesis in last section.

1.1 Basics of Control System Design

The performance of control system is evaluated by its ability to determine (a) process response reasonably close to desired values without strong demands on specifications, (b) proper regulation of external noise and disturbances, (c) realistic actuator signal that computes the control action, and (d) robustness against various uncertainties. These properties are established using a variety of control techniques that capture the essential dynamics of the system and permit the exploration of possible behaviors in the presence of uncertainty. The system performance can be influenced externally by noise in sensing and actuation subsystems, dynamic non-linearity, external disturbances, ambiguity and vagueness in systems structural and governing parameters. These factors affect the underlying system operation and intensify the complexity of control system design procedure. Hence, meticulous cyclic efforts of modeling, design, simulation, testing, and implementation of control system are required to obtain the optimal parameters for proper controller design [1].

1.1.1 Structure of Control System

The general block diagram of control system is represented in Fig. 1.1 which depicts the essential modules like system, controller, and tuning strategy. Along with this, the sources of uncertainties like disturbance, feedback noise, and model uncertainty are demonstrated. The main principle of working of this system depends upon the proper selection of controller parameters to generate the control law so that to get the desired performance. Here, the controller output is manipulated and applied to the system to produce the desired output response. In order to achieve this, the proper tuning of control parameters is a fundamental goal.

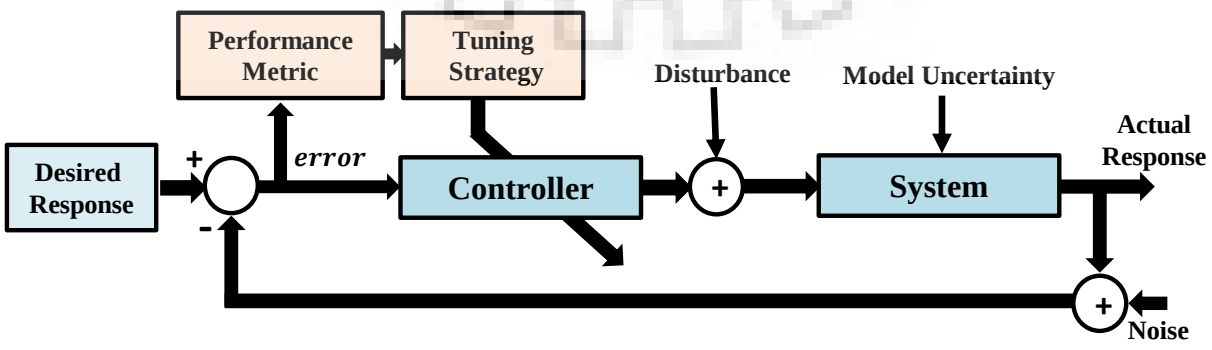


Fig. 1.1: General structure of control system.

The main aim of controller tuning is to search for optimal controller parameters to satisfy a set of specifications and design objectives. In order to derive a successful controller, characteristics like load disturbance rejection/attenuation, measurement noise immunity/attenuation, set point follow-up, robustness to model uncertainties are mainly considered [3]. Different time-domain and frequency-domain performance metrics are used as per requirement of desired specifications.

1.1.2 Challenges in Controller Design

The controller design procedure has certain issues and challenges other than external disturbance, model uncertainty, and noise which are shown in the Fig. 1.1. In the following paragraphs, we describe the major challenges that are considered in the controller design procedure.

Robustness : Last few decades have witnessed the rapid advances and diffusion of technologies in science, engineering, robotics, biomedical, economics and other fields [4]. Therefore, the complexity of such systems has increased and require highly specialized skills and knowledge to manufacture, operate, and control such complex systems. Along with other problems, uncertainties in overall system models arise frequently in number of different forms and affect decision-making [5]. The information deficiency and imperfection of knowledge - due to imprecise, contradictory, incomplete, vague, etc. or some other varying data - are considered as the main cause of the uncertainty [6]. Uncertainty is divided as a) Model uncertainty: due to the variations in system structure parameters, b) Measurement uncertainty: due to the error on observed quantities, and c) Linguistic uncertainty: due to vagueness in fuzzy rules modelling and information transfer. Plant uncertainties are also classified as structured (parametric uncertainties) and unstructured uncertainties. These uncertainties may severely degrade the performance and lead to system instability. In control theory, robust control is an approach to design controller that explicitly deals with such uncertainties. It is necessary to design robust controllers to handle such uncertain non-linear systems. Recently, many approaches including soft computing techniques are used to design robust controllers [7].

Formulation of Objective Function : The proper selection and formulation of objective function play an important role in controller design procedure. Generally, the control system performance is assessed by several time-domain and frequency-domain performance measures. Thus, the composition of overall objective function should be done in accordance with the required performance measures. Various time-domain performance indices, as defined in Section 2.3, are utilized, and the effectiveness of the different controllers are evaluated. Every performance index has certain specifications which define the response quality of the system.

One or more objective functions may be considered for particular control system design prob-

lem. The generalized equation of the objective function for optimization problem is given as

$$\text{Minimize / Maximize} \qquad F(x) = (f_1(x), \ldots, f_m(x), \ldots, f_M(x))^T \tag{1.1}$$

$$\text{Subject to} \qquad \left\{ \begin{array}{ll} g_j(x) \geq 0 & j = 1,2,...,J; \\ h_k(x) = 0, & k = 1,2,...,K; \\ x_i^L \leq x_i \leq x_i^U & i = 1,2,...,n. \end{array} \right\} \tag{1.2}$$

here $m = 1,2,\ldots,M$ indicates the number of objectives. A solution of this problem can be obtained from a decision vector $x = (x_1, x_2, \ldots, x_n)^T$ defined in the n dimensional decision space. Each decision variable x_i is constrained between lower bound x_i^L and as upper bound x_i^U. The $g_j(x)$ and $h_k(x)$ are inequalities and equality constraints, respectively.

The above equations (Eqs. (1.1) and (1.2)) define the optimization problem with equality, inequality, and side constraints. For $M = 1$, the problem is known as single objective optimization (SOO) problem. While, for $M = 2$ or 3, the problem is known as a multi-objective optimization (MOO) problem. Further, the problem with $M > 3$ are known as a many objective optimization problem. In few cases, the weighted sum of different objectives in MOO problem is considered to convert it to SOO problem. The objective functions for controller tuning problem are selected on the basis of specifications delivering set-point following, load disturbance rejection attenuation, robustness to model uncertainties, and measurement noise immunity attenuation.

Constraints : The control goals may have certain constraints in terms of limits of performance measures and bounds of controller parameters. The minimum - maximum limits on the parameters are considered as a side bounds and defined by lower bound x_i^L and as upper bound x_i^U in Eq. (1.2). In most of the systems, the maximum sensitivity function M_s is used for an effective robustness analysis, its acceptable value is in between 1.4 to 2 [3]. These limitations in control system design are formulated as constraints of the optimization problem. Several equality and inequality constraints are required to satisfy during the design procedure [8]. Though different constraint handling techniques have been suggested by researchers in literature, penalty function based constrained optimization is widely applied. In penalty function method, the penalty is assigned to variables according to the depth of violation of feasibility defined by constraints. In this way, penalized function is considered as new objective function to maintain feasibility of the solutions [9].

Tuning Methodology : As discussed earlier, the procedure for tuning of controllers is executed to obtain a optimal solution at the end of the optimization process. The process is executed to satisfy some design objectives that are required for enhancing controllers performance and capabilities. This is achieved through tuning of various controller parameters for minimum values of performance indices. As the conventional controllers are amended with several advance

strategies, the tunable parameters have been largely increased. Therefore, the controller design problems developed as high dimensional optimization problems to provide higher flexibility to the designer. The tuning of controllers is an iterative process which finds the optimized parameter values by converging the performance indices toward optimal values [10]. Though many famous classical tuning techniques are available [3], they are not found suitable for these advance control strategies with several parameters. The formulation of proper objective function from performance indices is very crucial in such design problems. The selection of an appropriate tuning strategy is also required to obtain optimal parameters in lower computational time [11]. It is important to note here that, the high level knowledge of system dynamics is always required by control system designer for tuning with either SOO or MOO algorithms.

1.2 Soft Computing Techniques in Control System Design

Prof. L. Zadeh introduced the concept of soft computing as the emergence of various computing paradigms to incorporate human decision-making behaviour [12]. Mainly, the emphasis is given to exploit the tolerance for imprecision and uncertainty to achieve tractability, robustness at low solutions cost. Distinct soft computing methods such as fuzzy logic system, evolutionary computation, neurocomputing, artificial neural network [13] and chaos theory, etc., are employed to produce powerful hybrid intelligent systems.

Over the past decades, developments in soft computing methods have attracted considerable research interest in control, automation, signal processing, system modeling, and other emerging fields. Although the field of soft computing is continuously enhanced by contributions from the large community of researchers, it always provides an exceptional opportunity to advance the methodology and applications. In the similar spirit, the overall work presented in this thesis provides some important advancements and novel contributions to a growing area of soft computing based control system design.

1.2.1 Fuzzy Logic based Control System

The field of control system is always developing and various control strategies and controller structures have been suggested in past few decades to enhance the performance of systems under control [14, 15]. Earlier studies reported the application of conventional proportional- integral-derivative (PID) controllers in industrial applications because of their elementary design, easy implementation, and low cost [16, 17]. On the other hand, recent studies in this field did not find PID controllers suitable for complicated systems with uncertainty and non-linearity [18].

The incorporation of fuzzy logic systems (FLS) in control theory has enhanced the flexibility of controller design and increased its applicability to control the complex, non-linear, ill-structured, and uncertain systems [19, 20]. The fuzzy logic controllers (FLC) (also called as type-1 fuzzy logic controllers (T1-FLC)) have received considerable interest and widely used

than their conventional counterparts for control applications. The T1- FLC have unique characteristics such as (a) incorporation of knowledge based on human expertize, (b) no exact dynamic model of system is required hence useful when precise mathematical formulation are infeasible, (c) low development and maintenance cost, and importantly, (d) the ability of general framework of T1-FLC to handle uncertainty, etc [6].

The working of conventional FLS is based on fuzzy logic sets (also known as type-1 fuzzy sets (T1-FS)) which are intended to represent the vagueness, ambiguity, and uncertainty in the information using linguistic variables [21]. A fuzzy controller can be designed to emulate human deductive thinking to infer conclusions from the past experience. To satisfy the control objective, the decision making process of the controller is designed on the basis of fuzzy rule base where both situation and action have suitable fuzzy representation. It suits the control problems which are complex and mathematically ill-defined [22]. In literature, a number of researchers have proposed various designs and strategies of T1-FLC and investigated them for numerous applications [23–25] including robotic control [11, 26].

The type-1 FLS are further upgraded to interval type-2 FLS (IT2-FLS) by incorporating footprint of uncertainty (FOU) in the membership functions to form interval type-2 fuzzy sets (referred as IT2-FS, hereafter). The concept of IT2-FLS is widely employed to design robust controllers and interval type-2 fuzzy logic controllers (IT2-FLC) are proposed [27, 28]. As a result, new variants of theoretical operations, membership grades, formulae for the type-2 relations are developed [29]. The IT2-FLC have emerged as an interesting generalization of T1-FLC with the enhanced uncertainty handling skills provided by incorporating the FOU. Several researchers have demonstrated IT2-FLC for various applications such as flexible robot manipulator, autonomous mobile robot, inverted pendulum system, aerospace applications, liquid level process, power system applications, image edge detection, and congestion control for video streaming across internet protocol (IP) network, etc. In last few years, a considerable amount of interest has been seen in research on the strategies to obtain the optimal right structures of the IT2-FS to make the best of FOU provided by them. These design problems have framed as a high-dimensional optimization problem with constraints.

1.2.2 Metaheuristic Algorithms (MAs)

Over the last decades, distinct natural selections, food foraging patterns, animals group movements, physical laws, and other natural paradigms have attracted the researchers from various disciplines. Having inspired by such beautiful natural phenomena, researchers have proposed various optimization algorithms, also termed as Metaheuristic Algorithms (MAs), to enrich the field of computational intelligence, soft computing and optimization at large [30]. The previous and on-going research is attempting to solve cumbersome optimization problems from basic research to a huge number of real-world applications in science, engineering, industry, business, economics [31, 32]. These algorithms are found to be more effective in finding the optimal so-

lution than deterministic algorithms. Therefore, recent optimization trends are approaching to apply MAs for solving real life optimization problems [33]. The prominent features of MAs are: (a) unlike deterministic method, they are derivative free and are easy to implement with basic calculation of fitness function, (b) applicable to the wide range of problems domain as no prior knowledge of problem as well as the domain of the problem is required, (c) contrary to deterministic methods, they employ multiple solutions which are refined over course of iterations. Also, they explore and exploit search space faster by evolving the population using search strategies. Finally, population based nature benefit MAs in escaping from the potential dangerous situations, such as local optima stagnation and premature convergence [28]. Thus, the MAs have a clear edge over deterministic algorithms [11].

1.3 Motivation

As discussed earlier, the field of soft computing and its constituents always provide an exceptional opportunity to enhance the performance of real-life problems with imprecision and uncertainty. Consequently, there is a great scope and motivation to ameliorate, design, hybridize, and apply various soft computing techniques to solve complex problems. The work presented in this thesis provides some important enhancements and novel contributions to growing area of soft computing based control system design. To elaborate further, specific motives for different research objectives are discussed below.

Some common deficiencies faced by the majority of population-based optimization algorithms are lack of exploration ability, slow and premature convergence behaviour, and stagnation to local optima [33–35]. Nevertheless, Wolpert and Macready [36] stated in No free lunch theorem that there is no single MA which is best suited to solve all type of optimization problems. With reference to this, researchers in this field found a scope as well as challenge to design an efficient and state-of-the-art optimization algorithm that can alleviate the deficiencies [37]. In point of fact, this theorem forms the basis of development and encourages researchers to propose new MAs, now and then [28, 38]. This fact challenges and motivates us to design an efficient and up-to-date optimization algorithm that can handle all these deficiencies [37]. In this line of research, several new strategies are incorporated in existing algorithms, and new modified versions of the existing algorithms are proposed [39–41]. The strategies like chaotic mapping and opposition-based learning (OBL) are used for the modifications in the algorithms.

The tuning of controllers is considered as a high-dimensional, complex, multimodal numerical optimization problem, having several sub-optimal solution sets [42, 43]. Consequently, it is always a challenging task for designers to get the optimal controller parameter values. MAs are always preferred for obtaining the optimal parameters to get the best performance and robust response. Along these lines, we get motivation to investigate the efficacy of proposed MAs for control system design problems to get the desired design requirements.

Though, IT2-FLC has developed as an efficient tool to solve the complex problems with high

uncertainty and non-linearity, optimizing the controller parameters and the antecedent MF structures emerged as a time consuming, difficult, high-dimensional, and constrained optimization problem [5]. Hence, a very few researchers have addressed this problem of optimizing the controller along with its FOU. Particularly, this issue rendered the inspiration and motivated us to propose the new controller and present the systematic optimization strategy using appropriate optimization algorithm.

In control system design, minimization of error index for set-point and disturbance are two different objective functions which are somehow in conflict with each other. The problem of dealing with both the objectives simultaneously is considered as multi-objective optimization problem. So to solve this MOO problem, we investigate the applicability of well known MOO technique called NSGA-II.

As discussed above, the robot manipulators are extremely non-linear, MIMO, highly coupled, and complex systems wherein the parameter uncertainties, external disturbance, and random noise adversely affect the performances of these systems. Essentially, this fact creates a rational motive behind using robotic manipulator as a complex system for performance analysis of proposed EA and controllers.

1.4 Research Objectives and Thesis Contributions

The aforementioned potential and prospect of soft computing techniques motivated us to work on various soft computing techniques to improve the performance of controller design. The main objective of this thesis is to improve the performance of the different soft computing techniques and use them to design high performance intelligent controllers. In this regard, we have enhanced few components of soft computing techniques such as evolutionary algorithm and type-2 fuzzy logic system. The major contributions of this work are elaborated in following research objectives.

- **Design Efficient Optimization Algorithm for Enhancing the Performance of Complex Systems.** Two well established SI based algorithms - Grey wolf optimizer (GWO) and artificial bee colony algorithm (ABC) - are hybridized to propose new improved GWO-ABC hybrid algorithm. In this work, information sharing property of employed bees in ABC is adopted with conventional GWO to comprehend benefits of both the algorithms. The elitism based population initialization scheme is presented incorporating chaotic mapping and OBL strategy to start with widespread range of solutions. These strategies have been incorporated to overcome the shortcomings of the conventional GWO by improving exploration capability, convergence rate and reduce the chances of entrapment at local optima. The performance of GWO-ABC is tested on a test bed of 27 synthesis benchmark functions of different properties and the result are compared with 5 other MAs. Various performance measures such as exploitation and exploration analysis, convergence rate analysis, ranking, and non-parametric Wilcoxon rank sum test are carried out.

- **Performance Analysis of the Proposed Algorithm for Optimization of Different Controller Design Problems.** Here, the efficacy of the proposed GWO-ABC algorithm is examined for controller design problem for variety of linear and non-linear test bench process plants, with and without time delay. All the design requirements like low overshoot, better rise time, faster settling time, minimum steady-state error, and performance index are evaluated and compared to other counterparts. Further, the communication signaling strategy used in cooperative foraging is incorporated in conventional GWO with continuing its leadership hierarchy approach to present improved CFGWO algorithm. Moreover, for testing the performance of the proposed CFGWO for real-world applications, the proposed algorithm is examined for optimal tuning of controller parameters for trajectory tracking problems of a 2-link robotic manipulator. The comparative graphs of trajectory tracking performance, the path traced by the end-effector, and X and Y coordinate versus time variations against their desired reference curves are illustrated. Also, plots of position errors and controller output for both the links are also presented and conclusions are drawn.

- **Design of Interval Type-2 Fuzzy Precompensated PID Controller.** A novel concept of an efficient interval type-2 fuzzy precompensated PID (IT2FP-PID) controller is presented for trajectory tracking of 2-link robotic manipulator with variable payload. Mainly, the controller is comprised with unique features of interval type-2 fuzzy precompensated controller cascaded with conventional PID controller. Along with this, the structure of antecedent MFs of IT2-FLC is optimized to acquire maximum benefit of FOU. A systematic strategy for optimizing the controller parameters along with scaling factors and the antecedent MF parameters for minimization of performance metric integral time absolute error (ITAE) is presented. The GWO-ABC algorithm is utilized and the results are compared to type-1 fuzzy precompensated PID (T1FP-PID), fuzzy PID (FPID), and conventional PID controllers. More significantly, exhaustive robustness analysis is carried out in presence of distinct non-linear dynamics such as (i) payload variations, (ii) model uncertainties, (iii) disturbance in signals, and (iv) random noise at feedback path.

- **Constrained Multi-Objective Optimization Approach for Robust Controller Design and Performance Analysis.** In this objective, we demonstrate the applicability of MOO methodology for tuning of controller parameters for multi-variable, constrained and complex control systems. We proposed a simple design and parameters tuning approach using MOO of two-degree of freedom fractional order proportional-integral-derivative (2-DOF-FOPID) controller applied to magnetic levitation system (MLS). The major robustness investigations are carried out by applying variable input and external disturbance. Minimization of performance index integrated absolute error (IAE) for both the set-point tracking and the external disturbance rejection is considered as two conflicting objective functions.

1.5 Organization of the Thesis

The rest of this thesis, subsequent to the introduction (in current chapter), is organized in the following chapters.

Chapter - 2 begins with the basics and literature review of different soft computing techniques and their applications in the field of control system design. Then comparative features of T1-FLS, IT2-FLS, fractional order system are elaborated. The mathematical concepts and basic fundamentals of the relevant methodologies of controller design, optimization algorithms, optimization criteria, and system models used in this thesis are also described.

Chapter - 3 presents the details of the new hybrid GWO-ABC optimization algorithm and its application to complex systems. Initially, the overview of original GWO and ABC algorithms is presented. Then, the proposed algorithm is systematically explained with flowchart, pseudo-codes, and algorithms. Algorithm for new population initialization scheme is also presented. The GWO-ABC is investigated on 27 synthesis benchmark unconstrained test functions of distinct characteristics and results are compared with other prominent algorithms for various analysis metrics. The performance is substantiated through obtained empirical results on exploration and exploitation analysis, convergence behaviour, rank, and statistical test results.

Chapter - 4 is organized with performance analysis of the proposed algorithms for optimization of different controller design problems. Initially, the GWO-ABC algorithm is applied on four test bench process plants to obtain the optimal time-domain specifications. Results with all the design requirements like low overshoot, better rise time, faster settling time, minimum steady-state error, and performance index are evaluated and compared to other counterparts. Further, improved CFGWO algorithm is presented and examined for optimal tuning of controller parameters for trajectory tracking problems of a 2-link robotic manipulator.

Chapter - 5 proposed the design procedure of novel interval type-2 fuzzy precompensated PID controller for trajectory tracking problem of 2-link robotic manipulator with the variable payload. In this, the strategy to tune the various controller parameters, scaling factors, and antecedent MF parameters of IT2FP-PID is presented. Later, the exhaustive robustness analysis in presence of distinct non-linear dynamics such as (i) payload variations, (ii) model uncertainties, (iii) disturbance in signals, and (iv) random noise at feedback path is reported.

Chapter - 6 establishes the multi-objective control system design problem and identifies the constraints on the basis of sensitivity function. Then, non-dominated sorting genetic algorithm (NSGA-II) is employed for solving this constrained multi-objective optimization problem. The comparative performance analysis of the proposed 2-DOF-FOPID controller against classical PID, FOPID, and 2-DOF-PID controllers is carried out and results are reported.

Chapter - 7 comprises conclusions of the entire work and discusses the recommended directions for future applications and studies.

Furthermore, appendices are also included to give detailed mathematical modelling of robotic manipulator and test function suite used for experimental studies.

Chapter 2

Literature Survey and Preliminaries

This chapter presents the review of the literature related to the soft computing techniques and different control methods followed by an overview of mathematical background of IT2-FLC and other methodologies studied in this thesis.

2.1 Enhanced Techniques in Controller Design

The control system design has been significantly enhanced by the advancements in various control theories and practices [44]. In this section, a comprehensive review of the developments in control strategies is presented. Furthermore, a detailed description of major techniques like FLC and fractional order calculus is also provided.

2.1.1 Evolution of Control Strategies

During earlier research work, the conventional proportional- integral- derivative (PID) controllers have been widely used in industrial applications because of their elementary design, easy implementation, and low cost [3]. Thereafter, several P/ PI/ PD/ PID controllers have been proposed in the literature and achieved a widespread acceptance in the control industry [45]. As the parameters of physical system deviate with time and operating conditions, appropriate tuning strategy is required to generate a moderate control signal [46]. Several classical tuning techniques like Ziegler-Nichols method, Tyreus-Luyben method, Cohen and Coon method, etc., have been developed and utilized to tune the controller parameters [3]. Recent studies in this field found that these classical PID controllers and tuning methodologies are inefficacious for complicated systems with uncertainty and non-linearity [47, 48]. Subsequently, various control strategies and controller structures such as fuzzy logic [16, 49], fractional order calculus [18, 50], sliding mode control [51], cascade control [52], feed forward control [53], neural network [54], etc. have been incorporated with modified PID structures to enhance the performance of system under control.

In this line of research, Er and Sun [55] proposed a fuzzy PI+D controller and optimize the control parameters by Genetic algorithm (GA) for the robotic system. Meza *et al.* [56] proposed

a self-tuning fuzzy PID semiglobal regulator based on Lyapunov theory. They investigated the proposed PID regulator on robotic manipulators and claimed that it is better than classical PID due to the utilization of variable gain matrices. Recently, Kumar and Kumar [23, 24] suggested IT2-FPID controllers for the 5-DOF robotic manipulator. They optimized the parameters of underlying controller using hybrid ABC-GA algorithm; and by extensive experimental study they claimed that the proposed controller has a clear edge over its conventional counterparts while dealing with trajectory tracking problems. Alavandar *et al.* [57] experimented the effectiveness of FP-PID controller for a flexible robot manipulator using bacterial foraging based optimization and reported the better performance in respect of transient and steady-state response along with robustness in the presence of varying load conditions. Further in this direction, various MAs namely GA [42], chaotic PSO [58], CSA optimization [59], Big Bang-Big Crunch optimization [60], BFO [61], cuckoo search [62] and Tabu Search [63] have been applied for tuning of the enhanced fuzzy controllers and finding the optimal parameters for diverse applications.

A considerable amount of interest in the research on the strategies to obtain the optimal right structures of the IT2-FS for making the best of FOU, has been seen in last few years [60, 62]. These design problems are formulated as a high-dimensional optimization problem with constraints.

Due to the complex nature of the problem without the a feasible mathematical model, MAs have emerged as an efficient tool for finding optimal solutions [64]. In literature, there have been made many serious attempts to optimize the IT2-FLC using MAs [24, 64, 65], over the past few decades. In [5], Oscar and Patricia reviewed the usefulness of IT2-FLC in intelligent control and discussed the viability of various MAs to obtain the appropriate parameters of controllers. Conceptually, the type-2 fuzzy set theory(referred to as IT2-FS, hereafter) was introduced by Zadeh as an extension of the concept of an ordinary type-1 fuzzy sets (T1-FS) [66]. Subsequently, research community contributed in terms of theoretic operations, properties of membership grades, and defines formulae for the composition of type-2 relations [23, 29]. Karnik and Mendel established general formula for the extended sup-star composition of type-2 relations and presented a complete type-2 FLS theory [67]. The T1-FLC is upgraded to IT2-FLC by incorporating the type-2 fuzzy sets with uncertainty about their MFs. The IT2-FLC has emerged as an interesting generalization of T1-FLC with the supplementary variables provided by incorporating the FOU in IT2-FS. Specifically, IT2-FLC is built with rules integrating antecedents or consequents with uncertainty therein [65, 68].

T1-FLCs act upon type-1 fuzzy sets (T1-FSs) which determine the uncertainty by number in the range of [0, 1], are also sufficient to handle low level of uncertainties. On the other hand, most of the control applications exhibit a higher degree of uncertainty as a consequence of disturbances in the system and noise in feedback path due to environmental changes during signal acquisition and transmission. Several studies have concluded that T1-FLCs is not well suited for controlling such problems and recommended IT2-FLC as viable alternative to provide superior performance [4, 5, 65]. The main challenge in IT2-FLC design is to tune its parameters

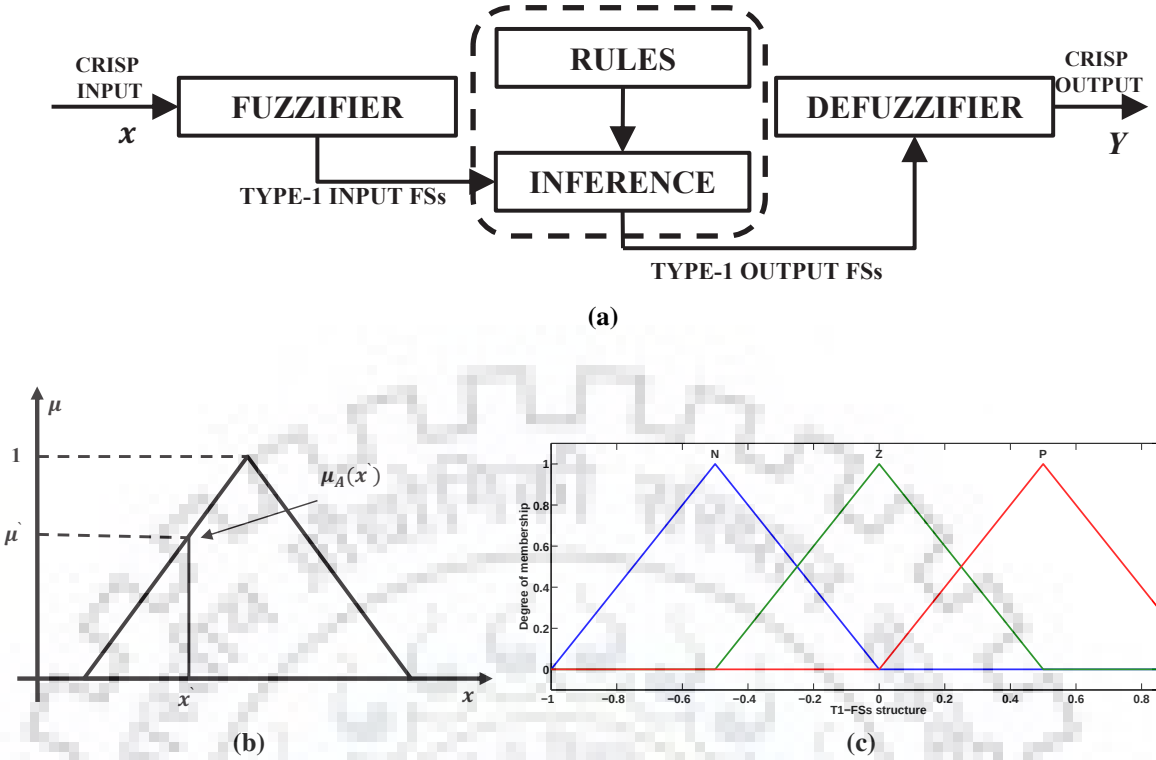


Fig. 2.1: Structure of (a) T1-FLS, (b) T1-FS, (c) Membership functions used in T1-FLS.

and MFs structure to suitable values to get optimal results [69]. It has been observed that, if the conventional PID control scheme is applied to the system with non-linearities, it will result in response with significant undershoots and overshoots. To overcome this behaviour, the fuzzy logic based precompensator PID (FP-PID) controller is proposed in [70,71] for the dc servomotor with uncertainties in variable load conditions. This advanced FLC based precompensation PID controller designed by adding fuzzy precompensator PD to traditional PID controllers. The FLC inference is designed to regulate the control signal which compensate the undershoots and overshoots in the system output when the system has unknown non-linearities. The significance of FLC for effectively improving the performance and robustness of conventional control methods is demonstrated and claimed.

Although, IT2-FLC has been developed as an efficient tool to solve the complex problems with high uncertainty and non-linearity, optimizing the controller parameters and the antecedent MF structures emerged as a time consuming difficult high-dimensional constrained optimization problem [5]. Thereby, a very few researchers have addressed this problem of optimizing the controller along with its FOU. In fact, this issue has rendered the inspiration to develop optimization strategies using appropriate optimization algorithm, to solve the problem.

2.1.2 Type-1 Fuzzy Logic Systems

The interval type-2 fuzzy logic system (IT2-FLS) is modified upgraded version of type-1 fuzzy logic system (T1-FLS). So, we first discuss the basic structure of T1-FLS whose schematic di-

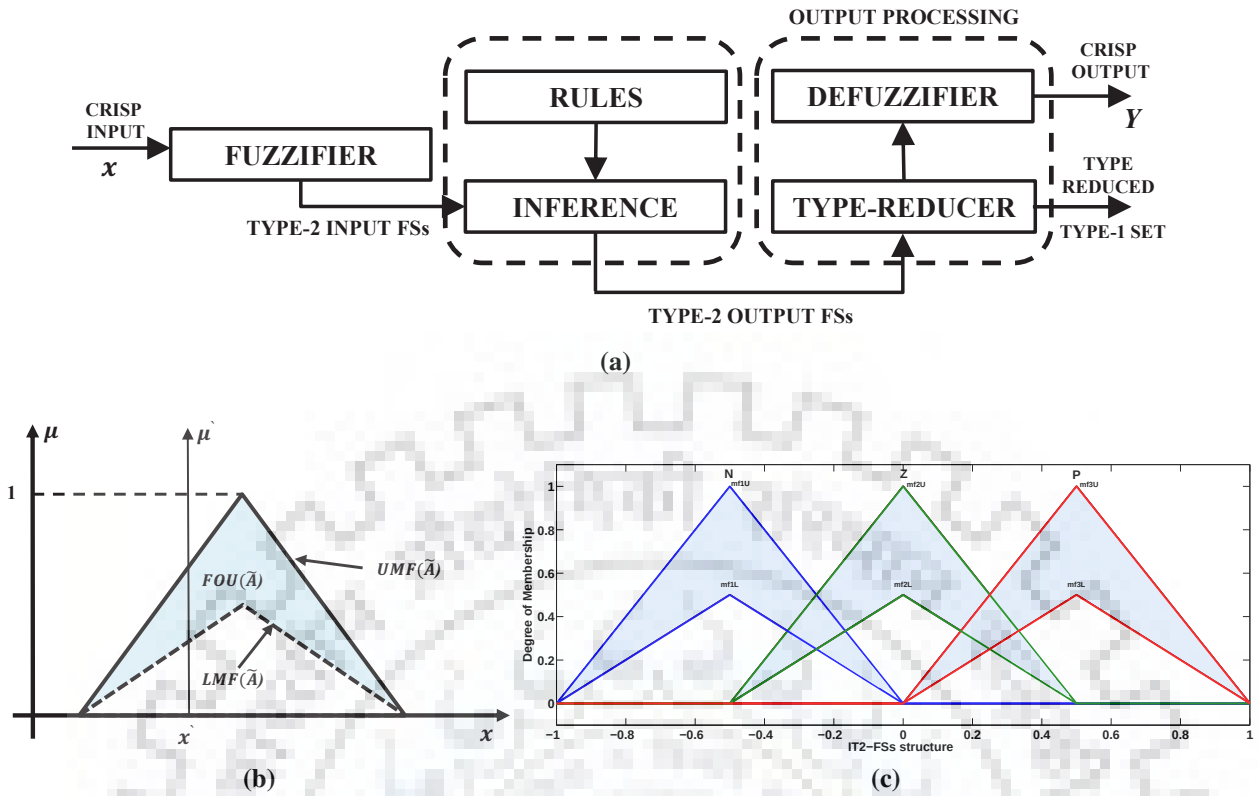


Fig. 2.2: Structure of (a) IT2-FLS, (b) IT2-FS, (c) Membership functions used in IT2-FLS.

agram is presented in Fig. 2.1 (a). As shown in Fig. 2.1 (a), T1-FLS consists of four units, fuzzifier, fuzzy rules, inference engine, and defuzzifier. The first unit fuzzifier is used to convert the input crisp signal to fuzzy sets (T1-FS) (as depicted in Fig. 2.1 (b)). Here, triangular MF is used which has single MF value (μ') at any point (x'). One or more MFs, as shown in Fig. 2.1 (c), are used to define the complete range of input variation. These inputs are further applied to inference mechanism which takes decision based on the fuzzy rules. The fuzzy rules are defined using expert knowledge to emulate human-like decision making in FLS. Generally, rule base is defined in the form of *If-then rules* and specified according to the design requirements. In most of the experiments here, Takagi-Sugeno fuzzy model is used where the nature of output singleton is as demonstrated in Fig. 2.3. The rule bases and their corresponding surface plots are described in further chapters. After fuzzy logic processing, the result is also obtained in the form of output fuzzy sets. Further, the defuzzifier is utilized to convert the final result into crisp output signal.

2.1.3 Type-2 Fuzzy Logic Systems

The design of an interval type-2 fuzzy logic system (IT2-FLS) is presented in Fig. 2.2 (a). The IT2-FLS is an extension of conventional T1-FLS, with additional unit of 'type reducer'. Therefore, it consists of five units - fuzzifier, fuzzy rules, inference engine, type-reducer and defuzzifier - to specify complete processing from 'Crisp Inputs: x ' to 'Crisp Outputs: $\boldsymbol{Y}$ '. The IT2-FLS can be viewed as a fuzzy operational mapping expressed quantitatively as $Y = f(x)$.

In IT2-FLS, the crisp inputs to fuzzifier may be uncertain (e.g., noisy signal) or certain (e.g., perfect signal) which are converted to IT2-FS as shown in Fig. 2.2 (b). The main feature of IT2-FS is that it consists of a finite region defined as *footprint of uncertainty* (FOU) to represent the uncertainty. Let us consider any IT2-FS, represented as $\tilde{A}$ and defined by type-2 MF $\mu_{\tilde{A}}(x,u)$, as shown in Fig. 2.2 (b).

The upper membership function (UMF) and lower membership function (LMF) of $\tilde{A}$ define the bounds of the FOU ($\tilde{A}$) in terms of two T1-MFs. In fuzzy theory, LMF and UMF defining FOU($\tilde{A}$) are symbolized as $\underline{\mu}_{\tilde{A}}(x), \forall x \in X$ and $\overline{\mu}_{\tilde{A}}(x), \forall x \in X$, respectively. Here the universe of discourse is represented as $X \in [0,1]$. Actually, the IT2-FS shown in Fig. 2.2 (b) is a vertical slice of the T2-FS where amplitude of secondary MF equal to unity and the shaded region represents the FOU.

Fuzzifier: To start with, the 'fuzzifier' maps a crisp point $x = (x_1, \ldots, x_m)^T \in X_1 \times X_2 \cdots \times X_r \equiv X$ into a type-2 fuzzy set $\tilde{A}_x$ in X. These signals then activate the inference engine and the rule base to produce output IT2-FSs. Further, these signals are converted to type reduced T1-FS using TR unit and defuzzifier is used to produce crisp output. There are many TR and defuzzification methods available in literature [29, 72, 73]. Karnik–Mendel (KM) algorithm is used in this work. The working principle and rule base of IT2-FLS is mostly equivalent to the T1-FLS with additional output processing TR unit.

IT2-FS : To understand IT2-FS reasoning mathematically [65, 74, 75], let us consider any T2-FS, represented as $\tilde{A}$ and defined by type-2 MF $\mu_{\tilde{A}}(x,u)$. Where x is primary variable.

We can state the above equation in another way as

$$\tilde{A} = \{((x,u), \mu_{\tilde{A}}(x,u)) \quad \forall x \in X, \forall u \in J_x \subseteq [0,1]\} \tag{2.1}$$

where $0 \leq \mu_{\tilde{A}}(x,u) \leq 1$, X is the primary domain, and J_x is the secondary domain. Similarly, $\tilde{A}$ can also be defined over all permissible range of x and μ as

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} \mu_{\tilde{A}}(x,u)/(x,u) \quad J_x \subseteq [0,1] \tag{2.2}$$

here '$\int\int$' represents the union operation [65]. Other terms are (i) primary variable: x in the domain of primary MF: X, (ii) secondary variable μ in the domain of secondary MF: J_x (also known as primary membership grade), and (iii) secondary membership grade: $\mu_{\tilde{A}}(x,u)/(x,u)$ (also known as amplitude of secondary MF).

According to definitions [65], with $\mu_{\tilde{A}}(x,u) = 1$ the T2-FS $\tilde{A}$ is converted to IT2-FS expressed as

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x \subseteq [0,1]} 1/(x,u) = \int_{x \in X} \left[\int_{u \in J_x \subseteq [0,1]} 1/u \right] /x \tag{2.3}$$

The IT2-FS are computationally simple form of T2-FS after reducing one dimension by fixing it to unity, i. e. the secondary membership grade $\mu_{\tilde{A}}(x,u) = 1$.

Rule base : The open source toolbox [72] provides intuitive execution of Takagi-Sugeno-

Kang (TSK) type IT2-FLS, whose generic rule structure for any $n \in N$ rule is defined by as :

$$R^n : \text{IF } x_1 \text{ is } \tilde{X}_1^n \text{ and } \ldots \; x_m \text{ is } \tilde{X}_m^n, \text{then } y \text{ is } \tilde{Y}^n, \tag{2.4}$$

here $\tilde{X}_j^n$ $(j = 1, \ldots, m)$ represents antecedent MFs and consequent MFs are expressed by $\tilde{Y}^n, n = 1, \ldots, N$. This implies the multi-input single-output (MISO) operation.

Output Processing : According to IT2-FLS shown in Fig. 2.3 (a), the output processing unit processes both TR and defuzzification and the overall output calculation procedures is explained in the steps given below [65, 72, 75]:

1. Initially, the fuzzifier determines IT2-FSs in terms of $[\underline{\mu}_{\tilde{X}_j^n}(x'_j), \overline{\mu}_{\tilde{X}_j^n}(x'_j)]$, where $j = 1, \ldots, m$, $n = 1, \ldots, N$, and $x' = (x'_1, \ldots, x'_j, \ldots, x'_m)$ is crisp input vector to IT2-FLS.

2. In agreement with rule base and inference mechanism, the firing interval $F^n(\boldsymbol{x'})$, $n \in N$ for the n^{th} rule is calculated as

$$F^n(\mathbf{x}') = \left[\underline{f}^n, \overline{f}^n\right], n \in [N] \tag{2.5}$$

where

$$\begin{aligned} \underline{f}^n &= (\underline{\mu}_{\tilde{X}_1^n}(x'_1) \times \cdots \times \underline{\mu}_{\tilde{X}_m^n}(x'_m)) \\ \overline{f}^n &= (\overline{\mu}_{\tilde{X}_1^n}(x'_1) \times \cdots \times \overline{\mu}_{\tilde{X}_m^n}(x'_m)) \end{aligned} \tag{2.6}$$

here '$\times$' denotes a product operation.

3. Next, a widely used center - of - sets (COS) TR method is performed to combine above firing intervals and the corresponding rule consequent as

$$Y_{cos}(\mathbf{x}') = \bigcup_{\substack{f^n \in F^n(\mathbf{x}') \\ y^n \in Y^n}} \frac{\sum_{n=1}^N y^n f^n}{\sum_{n=1}^N f^n} = [y_l, y_r] \tag{2.7}$$

$$y_l = \frac{\sum_{n=1}^L \underline{y}^n \overline{f}^n + \sum_{n=L+1}^N \underline{y}^n \underline{f}^n}{\sum_{n=1}^L \overline{f}^n + \sum_{n=L+1}^N \underline{f}^n} \tag{2.8}$$

$$y_r = \frac{\sum_{n=1}^R \overline{y}^n \underline{f}^n + \sum_{n=R+1}^N \overline{y}^n \overline{f}^n}{\sum_{n=1}^R \underline{f}^n + \sum_{n=R+1}^N \underline{f}^n} \tag{2.9}$$

here iterative KM method is used to obtain the switching points L and R [29].

4. Finally, TR output sets are converted to crisp value by mean operation calculated as

$$y = \frac{y_l + y_r}{2} \tag{2.10}$$

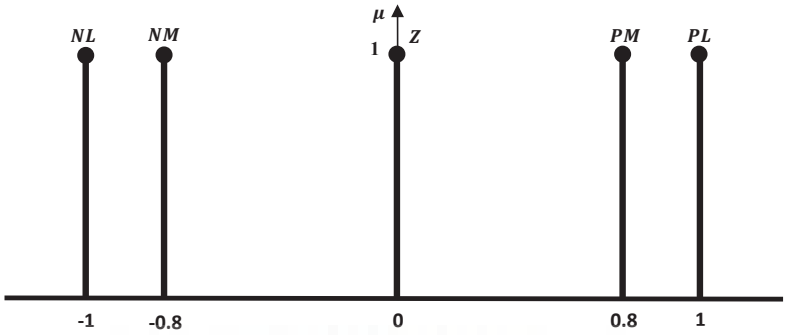


Fig. 2.3: Illustrations of the general framework of consequent MF used in this study.

2.1.4 Fractional Order System

The fractional order calculus was introduced much earlier and subsequently developed by researcher's community for various applications [24]. The theory of fractional calculus has been significantly contributed by various mathematicians like Liouville, Riemann, Weyl, Grünwald, and Letnikov [76]. Fraction order is considered as a generalized form of classical integer order integro-differentiation which avails accurate understanding of complex systems and provides an extra degree of freedom to designers [77]. Considerable amount of literature claimed that fractional order methods can precisely model the real objects and systems compared to classical integer methods [42, 61]. In last few decades, the non-integer orders in derivative and integrator operators have become an ongoing topic. Recently, the researches on fractional order modelling and fraction order controllers have been increased rapidly. Incorporation of fractional calculus in controller design has widely increased to take the benefits of additional flexibility to meet the controller's design specification. Thereby, the design of fractional order controllers and fractional orders processes has emerged as a new field for control system applications [22, 78]. Initially, the fraction calculus is integrated with PID to propose Fractional Order PID (FOPID) controller, which provides extra freedom to designers through two new tuning parameters, i.e. fractional orders of integro-differential terms (λ and μ). Various fractional order controllers are employed in this thesis to get benefits of this. The basic equation of FOPID controller is given in equation (4.1). The effect of fractional values λ and μ on FOPID controller can be seen in Fig. 2.4, where the shaded region represents the non-integer values defining FOPID controller. The integer values of λ and/or μ specify the P, PI, PD, or PID controllers.

Several studies have been reported that FOPID outperforms conventional PID controllers, in terms of robustness, over wide range of applications including robotics [79, 80]. Automatic voltage regulator (AVR) system is controlled by optimized FOPID controller [81]. Pan *et al.* [82] proposed the design of FOPID controller for AVR using chaotic MOO method to handle the conflicting objectives. Several other researchers have examined the implementation of the FOPID controllers for various plant models such as electric power and energy system [77], robot manipulator [26, 83] etc. Along with this, the fuzzy logic is incorporated with FOPID to use artificial intelligence in the design process. Also, the high dimensional optimization problems

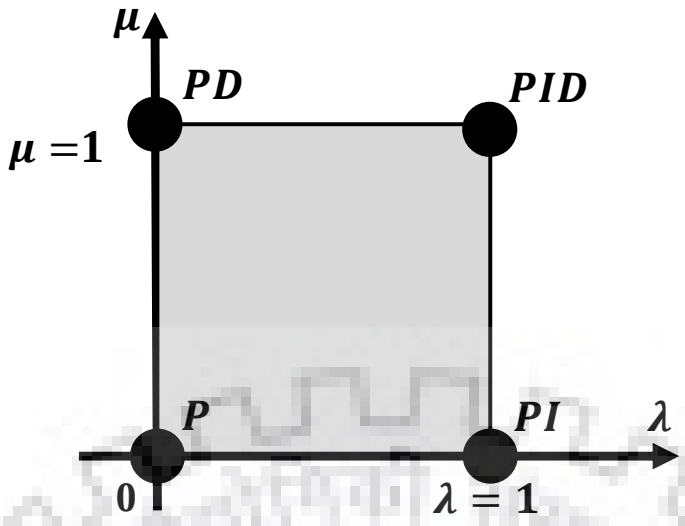


Fig. 2.4: FOPID controller region based on the values of fractional order terms (λ and μ).

are defined and the controller parameters have been optimally tuned using MAs. Das *et al.* [42] proposed fuzzy based FOPID and optimized the parameters with genetic algorithm. Optimal FOPID design procedure using tabu search and GWO is presented in [63] and [50], respectively. Chaotic PSO is proposed to control the hybrid power system with renewable energy using fuzzy and FO based controller [58]. In recent years, several types of FOPID controllers are suggested for controlling various industrial processes. In [20], the 2-DOF-FOPID controller is presented for robotic manipulator with payload. In this research work, Cuckoo search algorithm (CSA) is applied for parameter tuning and enhancing the robustness of controller towards trajectory tracking, parameter variation, and disturbance rejection.

Definitions of Fractional Order Calculus

The fundamental operator representing the fractional i.e., non-integer order of integro-differential terms is represented as ${}_aD_t^\alpha$, where $\alpha \in \mathbf{R}$ is the order of the differentiation or integration and a and t are the bounds of the operation. The basic fractional order integral and differential operators can be stated as below:

$$ {}_aD_t^\alpha = \begin{cases} \frac{d^\alpha}{dt^\alpha} & \alpha > 0 \\ 1 & \alpha = 0 \\ \int_a^t (d\tau)^{-\alpha} & \alpha < 0 \end{cases} \tag{2.11} $$

where constant a is related to initial conditions and α is fractional order operator. α is usually a real number ($\alpha \in \mathbf{R}$) but it can also be a complex number, where **R** stands for Real.

Cauchy's Definition

For the implementation of fractional order operators, multiple definitions are available in the literature such as Grünwaldv-Letnikov, Riemann-Liouville, Caputo definition, Cauchy integral

formula, and others. The definition of Cauchy integral formula used in this study is a general extension of the integer-order Cauchy formula and given as

$$D^{\alpha} f(t) = \frac{\Gamma(\alpha+1)}{2\pi j} \oint \frac{f(\tau)}{(t-\tau)^{1+\alpha}} d\tau \tag{2.12}$$

where circle on $\int$ is the smooth curve encircling the single-valued function $f(t)$, $\Gamma(.)$ is the Euler's Gamma function, and $0 < \alpha < 1$ is order of operator.

FO Approximation:

Among other competitive methods, Oustaloup's recursive approximation method [84] is applied in this work for implementation of fractional order operators (μ and λ). This method is based on a recursive distribution of poles and zeros in the form of an approximating transfer function.

$$s^{\lambda} = K_f \prod_{k=-N}^{k=N} \frac{s+\omega_{zr}}{s+\omega_{pr}} \tag{2.13}$$

here λ is the order of fractional integro-defferentiator and 2N + 1 represents the order of the approximation (filter). The zeros (ω_{zr}), poles (ω_{pr}), and the gain of filter (K_f) are represented by following equations.

$$\omega_{zr} = \omega_b \left(\frac{\omega_h}{\omega_b}\right)^{\frac{K+N+\frac{1}{2}(1-\lambda)}{2N+1}}, \ \omega_{pr} = \omega_b \left(\frac{\omega_h}{\omega_b}\right)^{\frac{K+N+\frac{1}{2}(1+\lambda)}{2N+1}}, \text{ and } K_f = \omega_h^{\lambda} \tag{2.14}$$

where $k \in [-N, N]$ and ω_b, ω_h are the ranges of frequency. This rational approximation method is preferred over other methods as it can easily exhibit the possibilities of implementing in higher-order digital and analog filters hardware design [22, 81]. The frequency range is taken as $\omega = (10^{-3}, 10^{3})$ rad/s, also, the 5th order Oustaloupas approximation with (N = 2) is applied for the fractional order operator design.

2.2 Techniques to Enhance MAs

The MAs are classified in three major categories: evolutionary, swarm intelligence (SI) based, and physical law based algorithms. Along with this categorization, some researchers have tried their hands on human-related techniques to propose new algorithms. Until last decade, evolutionary algorithms like genetic algorithm (GA) [85] and differential evolution (DE) [86] had received much attention in the literature. In recent years, there is rapid growth in the use of SI based optimizers like particle swarm optimization (PSO) [87, 88], grey wolf optimizer (GWO) [89], artificial bee colony (ABC) [90], ant lion optimizer ALO [91], cuckoo search CS [92–94], and firefly algorithm (FA) [95], krill herd (KH) algorithm [96], etc.. Basically, the catch behind the success of SI based algorithms is that the intelligence of swarm lies on the collective be-

haviour of swarms, mainly, (a) adoptive self-organization, and (b) decentralization using division of labour [34]. The swarm organization is the way of organizing the swarm in a particular manner without any external force or input. While, division of labour corresponds to the way of distributing the task among swarm candidates. Many other optimizers like central force optimization (CFO) [97], gravitational search algorithm (GSA) [98], and harmony search (HS) [99] are motivated by physical laws. Recently proposed teaching learning-based optimization (TLBO) [100] and League championship algorithm (LCA) [101] are motivated from human behaviour.

MAs have emerged as an important methodology among many researchers for solving complex problem in the field of computational intelligence. As noted in Section 1.2.2, the MAs outperform other classical techniques in many key features like collective learning process, self-adaptation, and robustness. Therefore, MAs are widely accepted for solving various complex practical applications in business, science, engineering, commerce, and other fields.

MAs are population based optimization algorithms and follow three steps in every iteration to guide their search process towards global optima. These steps are: (a) self-adoption - each member in population learns from its environment and advances toward optimal solution, (b) cooperation - members share information by collaborating with each other, and (c) competition - members undergo selection on the basis of their fitness to proceed to next iteration operation. Usually, these algorithms are stochastic and adopt different ways to realize these steps [34]. New techniques for executing these steps are always proposed and the field of MA is continuously developing by many researchers by incorporating various techniques. As one objective of this work is to design and propose a new MA, some enhancement techniques, which are incorporated in this work, are elaborated in this section.

2.2.1 Hybridization

As discussed above, several MAs are developed recently and their behavior is determined by trade-off between the exploitation and exploration relationship. It is reported in many empirical studies that most of these MAs are better in either of one behaviour and lacking in others. The current research trend is shifting towards the hybridization which is done by combining two or more algorithms. Therefore, the hybridized algorithm accrues the benefits of more than one algorithm. Some of the modifications are done by (a) incorporating two or more conventional algorithms [37], (b) enhancing the randomness in initialization, selection, and mutation procedures using chaotic theory [102–104], random walk [38], levy flight [105], or Cauchy operator, (c) adopting other concepts like OBL, orthogonal learning, different crossover, and mutation strategies [106, 107], and (d) applying fuzzy logic and neural network based adoptive parameters. Thus, hybridization of algorithms is done to design a new algorithm to handle several real-world problems involving complexity, noisy environment, imprecision, uncertainty, and vagueness. In sum, the advantages of hybridized algorithms over a canonical one are as follows.

- Better balance in exploration and exploitation properties,

- Fast convergence rate,
- Better avoidance of stagnation at local optima,
- well suited for solving noisy multimodal optimization problems.

2.2.2 Chaotic Mapping

The chaotic evolution function is an iterative non-linear dynamical equation evaluated to generate fractals exhibiting chaotic behaviour. Generally, in stochastic MA, random-based techniques are superseded by chaotic variables to avail benefits of randomness of chaos, non-repetition, and ergodicity to enhance speed of search strategies and avoid the stagnation at local optima. Wang *et al.* [104] investigated 12 different chaotic maps by combining them with cuckoo search algorithm to tune the step size of cuckoos and reported that combination of Sinusoidal map gives comparatively better performance. In literature [102, 103], authors incorporated GWO with various types of chaotic mechanisms and reported considerable improvements in results, however, no clear-cut rule for selection of any specific chaotic map is suggested. In this thesis, chaotic mapping is incorporated in initialization and communication strategy of search agents to improve the performance of proposed algorithms. The equation of logistic chaotic evolution function, used in this work, is given as

$$ch(k+1) = 4 * ch(k) * (1 - ch(k)) \tag{2.15}$$

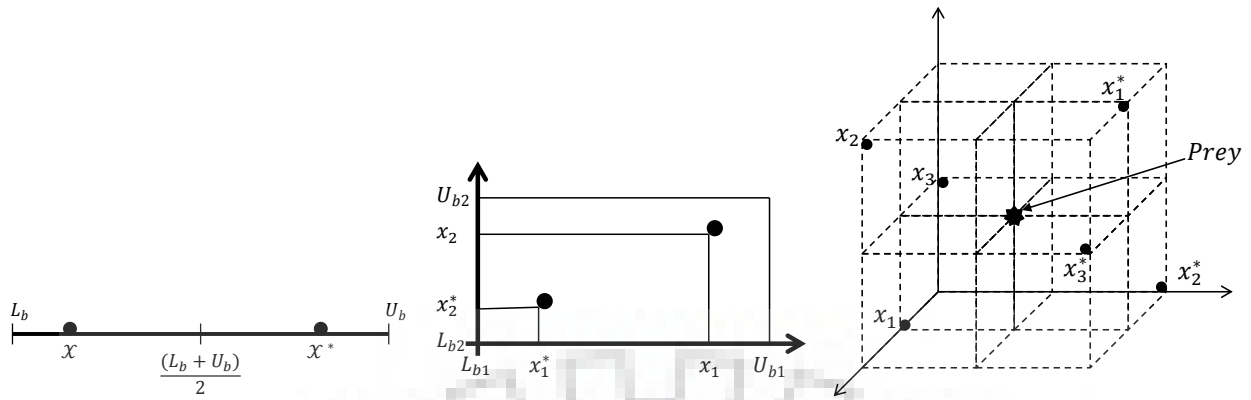
here k is number of iterations, fixed to 300, and initial value $ch(0) \in [0,1]$ is randomly selected.

2.2.3 Opposition-based Learning

In the realm of the metaheuristic optimization, the concept of opposition-based learning plays an important role for providing a better start to the underlying optimization algorithm. The prime concept behind OBL is to simultaneously consider an estimated solution and its corresponding opposite estimate to cover the larger portion of the search space in order to improve the diversity in the search regions [108]. Geometrically, as shown in Fig. 2.5, the solution X is transferred to geometrically opposite position X^* in given n-dimensional bounded search space. The movement of points after OBL in $1D$, $2D$, and $3D$ space is demonstrated in Fig. 2.5 (a), (b), and (c), respectively for clear understanding. The mathematical description of OBL is given below.

Consider a solution $X_i = (x_1, x_2, \ldots, x_j, \ldots, x_n)$, where $i \in N$ and N represents the size of population under consideration. Let for each decision variable $j \in [n]$ (n is the number of decision variable or dimension) and Ub_j and Lb_j denote the upper and lower bounds, respectively. Then, for each decision variable, its opposite candidate is generated as

$$x_j^* = Lb_j + Ub_j - x_j \tag{2.16}$$



(a) 1*D* **(b)** 2*D* **(c)** 3*D*

Fig. 2.5: Illustrations of point **X** and its corresponding opposite $\mathbf{X}^*$ according to OBL.

It is reported by several researchers that the probability of initialization of MA with fitter solutions, covering larger search space area, is enhanced by simultaneously probing original guess solutions along with their opposite counterparts [109]. In the proposed work OBL strategy is utilized in population initialization and to decide the opposite position for search agents in cooperative signalling.

2.3 Time-domain Performance Indices

In pursuance of implementing the optimization algorithms, the proper selection of objective function (Obj_fun) (also called cost function) is very crucial step. Mostly, error based time-domain performance measures are employed to achieve desired performance in controller designing. In literature, various indices like, IAE - integral absolute error, ISE - integral square error, and $ITSE$ - integral time square error have been employed in different control design studies. Mathematically, these indices are formulated as

$$IAE = \int_0^\infty |e(t)|dt$$

$$ISE = \int_0^\infty e(t)^2 dt$$

$$ITSE = \int_0^\infty te(t)^2 dt$$

$$ITAE = \int_0^\infty t|e(t)|dt \tag{2.17}$$

here t indicates time and $e(t) = y(t) - r(t)$ is a difference between current output $y(t)$ and reference desired output $r(t)$.

The whole work in this thesis utilized minimization of $ITAE$ - integral time absolute error for having the edge over other indices. Some important features of $ITAE$ are

- The absolute error minimizes the percent overshoot ($M_p\%$),
- The time multiplication term minimizes the oscillations in further response and effectively reduces settling time (t_s).

The minimized value of $ITAE$ signifies the negligible peak overshoots and smaller rise time (t_r). It also indicates the settling time (t_s) taken by the response to reach to nil steady-state error E_{ss}. The $ITAE$ based objective function is expressed as

$$Obj\_fun = ITAE = \int t|e(t)|dt \tag{2.18}$$

When there are multiple subsystems to be controlled under a complete control system design problem, the final objective function for the overall system with M subsystems is defined by the weighted sum of individual functions and given as

$$Obj\_fun = w_1 \times Obj\_fun_1 + w_2 \times Obj\_fun_2 + \cdots + w_M \times Obj\_fun_M$$

$$Obj\_fun = \sum_{j=1}^{M} w_j * Obj\_fun_j \tag{2.19}$$

here $w_1, w_2, \ldots, w_M$ are weights of M objective functions. The weights are selected by the designers as per requirements and set to unity in this thesis. The above equation is also used when the MOO problem is considered as a SOO problem. Furthermore, the objective functions for MOO are also considered as performance indices for different conditions. They are discussed in detail in Chapter 6.

2.4 Problems under Study

The enhanced soft computing techniques proposed in this work has been investigated against existing techniques for different systems under control. Here, a brief overview the variety of systems considered in this work is provided. Most of the systems are complex and non-linear, and their performance has been tested for distinct disturbances and noises.

2.4.1 Robotic Manipulator

In this era of automation, robotic manipulators are evolved by many researches with a desire to synthesize some aspects of human function by the use of mechanisms, instrumentation, and computers. After inception of robotics, the robotic manipulators are widely used as a promising device in the automated industrial applications in diverse engineering and science fields. These manipulators are incorporated to enhance flexibility, productivity, and accuracy considering reduced working cost and increased human working conditions. These systems are widely applied areas where repetitive and hazardous works are executed [110, 111].

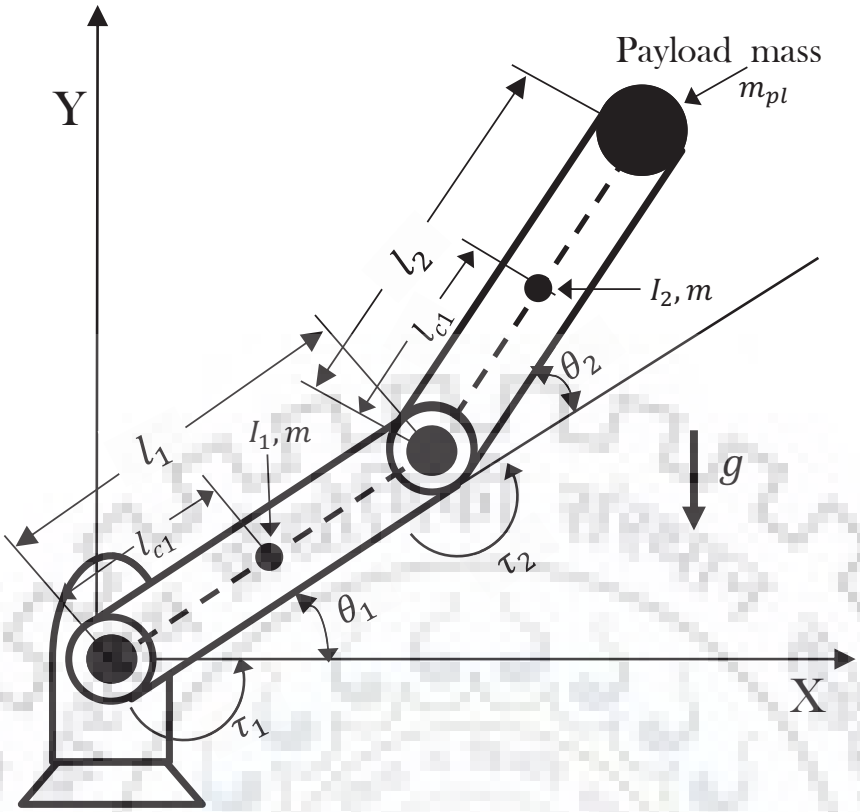


Fig. 2.6: Model of 2-link robotic manipulator with payload at tip.

The basic model of 2-link robotic manipulator is shown in Fig. 2.6. The details of mathematical modelling and definition of desired path trajectory are relegated to the appendix in Section A.1. The description of all the parameters is given in Table A.1. As the 2-link-joints provide 2-DOF angular motion, this model is also termed as 2-DOF robotic manipulator in some studies. In general, robotic manipulators are multi-input multi-output (MIMO), coupled, and highly complex non-linear systems. The motion of the manipulator and end-effector is manipulated by the efficient actuators connected to individual link-joints. In order to deal with overall complexities, these systems require an efficient and robust controller for accurate positioning of the end-effector which is a challenging task for control system designers.

From the literature survey, it is observed that conventional robot control methods require highly accurate mathematical modeling, analysis, and synthesis, yet these methods are not suitable for controlling robots in structured and unstructured uncertain environment. In this thesis, the different robotic models used for experimentation are as follows:

1. 2-link robotic manipulator with fixed payload.

2. 2-link robotic manipulator with variable payload.

2.4.2 Linear and Non-linear Benchmark Plants.

In conventional process control applications, controller designing for higher-order plants is considered as a difficult procedure. In literature, equivalent reduced-order models are used to represent higher-order plants. Some conventional controllers are applied to these systems using classical tuning rules [10, 78]. The model order reduction techniques are employed to higher-order plants for representing them by lower-order linear transfer functions. Some lower-order plants like first-order plus time delay (FOPTD) and second-order plus time delay (SOPTD) are used in [64] to validate the controller performance. In general, most of the practical processes under automatic control are non-linear higher-order systems and may have considerable dead time. For performance analysis, non-linear systems are approximated to linear systems with an approximate linear mathematical model of the non-linear system. In this way, the higher-order transfer functions are converted into generalized FOPTD and SOPTD systems using model order reduction techniques. In this study, some of the standard linear and non-linear benchmark plants are also used to verify the performance of the proposed algorithm.

2.4.3 Magnetic Levitation System.

The Magnetic levitation (MAGLEV) technology is applied in various fields like magnetic levitation vehicle, non-contact actuators, satellite launching, and precision engineering, etc. Being non-contact frictionless technology, maintenance cost of MAGLEV based systems is reduced and energy efficiency is increased. It is observed that, the behavior of MAGLEV is highly non-linear and unstable. Therefore, a robust controller design for MAGLEV system is very challenging task for control engineers. The controller designing for magnetic levitation system (MLS) provides platform for the research on MAGLEV technology [112]. In literature, various control schemes are suggested for MLS control, some are, optimized PID [113], fuzzy logic based digital controller [114], [115], adaptive robust output feedback control [116] and fuzzy PID [115]. The model, structural parameters, and mathematical modelling of the MLS is described in Chapter 6.

2.5 Concluding Remarks

This chapter describes the literature review and provides insight to the major techniques used in this complete work. A brief introduction on T1-FLS and IT2-FLS is presented with mathematical modelling of IT2-FS. In addition, the fractional order theory with definitions and Oustaloup's Recursive approximation is presented. The chaotic mapping and OBL methods are discussed with illustrations. Finally, the performance indices and system used in this study are discussed.

Chapter 3

Novel Hybrid GWO-ABC Algorithm for Complex Systems

This chapter presents the novel hybrid GWO-ABC algorithm and verifies its efficacy by several standard performance metrics. The original publication [11] is thoroughly described in different sections organized as follows. Basic introduction and motivation is discussed in Section 1. Following, Section 2 presents the brief overview of GWO and ABC algorithms. The architecture of the proposed GWO-ABC algorithm is described in Section 3. Subsequently, the GWO-ABC is substantiated using 27 test functions and the statistical and convergence curve analysis are discussed in section 4. Finally, the conclusions are drawn in Section 5.

3.1 Introduction

The group foraging and hunting behaviour of various species always provide the inspiration to develop various SI based algorithms. As discussed earlier, the catch behind the success of SI based algorithms is that each member of the swarm follows (a) self-organization, and (b) division of labour. Thus, swarm is organized in particular manner and distribute the task among swarm candidates. Recently, Mrijalili *et al.* [89] proposed SI based GWO algorithm widely utilized in several studies [117, 118]. The GWO algorithm is based on democratic social behaviour of the group of grey wolves demonstrated during chasing and hunting the prey. In last few years, GWO is mostly used by researchers for solving various engineering optimization problems [119–121]. At the same time, several attempts have been made to modify and hybridize GWO. Recently, in [105], authors incorporated levy flight- based pattern for hunting mechanism of wolves and greedy selection strategy. The algorithm is investigated on various test functions as well as real-world problems, and its performance is proved to be better than GWO. Similarly, in [38], authors

The work outlined in this chapter has been disseminated in the following publication:

- P. J. Gaidhane and M. J. Nigam, "A hybrid grey wolf optimizer and artificial bee colony algorithm for enhancing the performance of complex systems,"*Journal of Computational Science*, vol. 27, pp. 284-302, Jul 2018.

incorporated random walk mechanism in GWO to optimize the search ability of GWO. A new hybrid algorithm is introduced in [122] by incorporating GA in the GWO algorithm. It is used to minimize a simplified model of the energy function of the molecule. PM2.5 prediction model is combined with GWO [106] to improve the accuracy of the results. In [107], the basic GWO is hybridized with crossover and mutation operators to solve the economic dispatch problems.

As we discussed earlier, most of the population based algorithms suffer from the problem of either exploration or exploitation. GWO is also not isolated from some of these lacunas. Recently, Kishor and Singh [123] conducted a comprehensive study and reported some strengths and drawbacks of GWO over wide range of problems. In order to overcome the drawbacks of GWO, we proposed a new hybridized GWO called GWO-ABC. In this algorithm, the random initialization is replaced by a new initialization strategy, and to improve the exploration capability of GWO, information sharing methodology of ABC is adopted. In GWO-ABC algorithm, wolves adopt the information sharing strategy of bees to promote their exploration ability while keeping their original hunting strategy to retain exploitation ability. Moreover, a new method based on chaotic mapping and OBL is proposed to initialize the population. As discussed in Section 2.2.3, OBL is one of the initialization mechanism that is used to cover the larger portion of search space [108]. Similarly, in stochastic optimization field, random-based methodologies can be replaced by chaotic variables to avail benefits of chaos, non-repetition and ergodicity. Incorporating chaotic mapping in MAs improves their search performance and enhance the capability of jumping out of local optima. The aim for this new initialization method is to generate an initial population with already better individuals to set a solid ground for rest of the GWO-ABC algorithm to execute. The sole motivation behind incorporating changes in GWO is to help the algorithm to evade premature convergence and to steer the search towards the optimal region in faster manner. To assess the performance of the GWO-ABC, it is tested on a test bed of 27 synthesis benchmark functions of different properties and the result are compared with 5 existing state-of-the-art algorithms. From the analysis of the numerical results, it is apparent that the projected changes in the GWO ameliorate its overall performance and efficacy especially while dealing with noisy (problem with many sub-optima) problems.

3.2 Overview of Conventional GWO and ABC Algorithms

In this section we present a brief introduction of conventional GWO and ABC. For details of these algorithms, readers are encouraged to refer [89] and [90].

3.2.1 An Overview of GWO Algorithm

The GWO algorithm [89] mimics the democratic social behaviour of the group of grey wolves demonstrated during chasing and hunting the prey. Mostly, the grey wolves live in a pack of 5-12 members and follow strict dominant hierarchy based on leading qualities of wolves. The group is

Algorithm 1 Pseudo Code for GWO.

Input : Population of search agents N, dimension of solutions n, upper and lower bounds of solutions $[Ub_1, ..., Ub_n, Lb_1, \ldots, Lb_n]$, Max_iteration

Output : The best search agent $\overrightarrow{X_\alpha}$

Initialize the grey wolf population $\overrightarrow{X_i} = (x_1, x_2, \ldots, x_n)$ where $(i \in [N])$ and $x_j \in [Ub_j, Lb_j]$ $|\forall j \in [n]$
Initialize a, $\overrightarrow{A}$, $\overrightarrow{C}$, and $t = 1$.
Calculate the fitness of each search agent $f(X_i)$, where $(i \in [N])$
$\overrightarrow{X_\alpha}$= the best search agent
$\overrightarrow{X_\beta}$ = the second best search agent
$\overrightarrow{X_\delta}$ = the third best search agent
while ($t <$ Max_iterations) **do**
    for each search agent **do**
        Update the position of the current search agents by Eq. (3.5)
    end for
    Update a, $\overrightarrow{A}$, and $\overrightarrow{C}$
    Calculate the fitness of all search agents
    Update $\overrightarrow{X_\alpha}, \overrightarrow{X_\beta}$ and $\overrightarrow{X_\delta}$
    $t = t + 1$
end while
    return $\overrightarrow{X_\alpha}$

generally led by most prominent wolf, termed as alpha (α) wolf. Following, the second and third level of leading wolves are named as beta (β) and delta (δ) wolves in GWO. These second and third rank of subordinate wolves assist the alpha wolf in decision making for hunting the prey. All other following wolves are noted as omega (ω) wolves and they follow these high rank wolves for chasing and approaching the prey. The step-wise pseudo-code of original GWO [89] is presented in an Algorithm 1. Please note that here wolves are analogous to the potential solutions (solution agent) of the problem.

Mathematical Modelling of GWO : The mathematical model of social hierarchy of grey wolves and their strategy for encircling, hunting, and attacking of prey is described below :

Social hierarchy : Search is initialized with fixed number of wolves (solutions) randomly positioned in search space. The first three best solutions are obtained and named as alpha (α), beta (β) and delta (δ) wolves. The optimization procedure in GWO is mainly guided by these three wolves.

Encircling prey : The strategy of encircling of prey is adopted for the hunting. The mathematical model for this strategy is expressed for iteration t is as follows:

$$\overrightarrow{D} = |\overrightarrow{C} \cdot \overrightarrow{X_p}(t) - \overrightarrow{X}(t)| \tag{3.1}$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X_p}(t) - \overrightarrow{A} \cdot \overrightarrow{D} \tag{3.2}$$

where $\overrightarrow{A}$ and $\overrightarrow{C}$ are coefficient vectors, defined as $\overrightarrow{A} = 2\overrightarrow{a} \cdot \overrightarrow{r_1} - \overrightarrow{a}$ and $\overrightarrow{C} = 2 \cdot \overrightarrow{r_2}$.

Here the random vectors $r_1, r_2 \in [0,1]$ and $\overrightarrow{a} = a_1(1 - t/max\_iter)$ decreases linearly from a_1 to 0. The value of a_1 was set to 2 in original GWO. Also, here *max_iter* is the maximum iterations.

Hunting : The hunting process in GWO is led by α, β, and δ. Therefore, the positions of these three leading solutions are saved in the pack and the remaining omega wolves update their positions according to them. This position update approach can be modelled mathematically as follows:

$$\overrightarrow{D_\alpha} = |\overrightarrow{C_1} \cdot \overrightarrow{X_\alpha}(t) - \overrightarrow{X}|, \overrightarrow{D_\beta} = |\overrightarrow{C_2} \cdot \overrightarrow{X_\beta}(t) - \overrightarrow{X}|, \overrightarrow{D_\delta} = |\overrightarrow{C_3} \cdot \overrightarrow{X_\delta}(t) - \overrightarrow{X}| \tag{3.3}$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha}(t) - \overrightarrow{A_1} \cdot (\overrightarrow{D_\alpha}), \overrightarrow{X_2} = \overrightarrow{X_\beta}(t) - \overrightarrow{A_1} \cdot (\overrightarrow{D_\beta}), \overrightarrow{X_3} = \overrightarrow{X_\delta}(t) - \overrightarrow{A_1} \cdot (\overrightarrow{D_\delta}) \tag{3.4}$$

$$\overrightarrow{X}(t+1) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_2}}{3} \tag{3.5}$$

Attacking prey (Exploitation) : This phase of the GWO is controlled by parameter a, which gradually decreases in accordance with increasing iterations. Along with this, other two parameters $\overrightarrow{A}$ and $\overrightarrow{C}$ controls the search for the prey. $\overrightarrow{A}$ is adjusted to vary between $-2a$ to $2a$, and when $|A| < 1$ the wolves attack the prey.

Search for prey (exploration) : The exploration rate of GWO is controlled by parameter A and the search is more diverge if $|A| > 1$.

As discussed above, the exploration rate is controlled by parameters $\overrightarrow{A}$ and $\overrightarrow{C}$, which is reduced in later generations. This fact purveys lack of knowledge of candidate solutions from search space as limited information is shared among the solutions in the pack. Consequentially, many studies [103,105,123] have reported poor exploration capability as a major limitation of GWO. Therefore, this work aims to overcome this weakness by modifying the conventional GWO.

3.2.2 An Overview of ABC Algorithm

The ABC algorithm [90], proposed by Karaboga and Basturk, is inspired by foraging behaviour of honey bees' swarm and executed in three phases, namely, employed bee phase, onlooker bee phase, and scout bee phase. In ABC algorithm a food source position represents a single candidate solution and the amount of nectar in each food source is considered as fitness. The

number of employed bees and the onlooker bees are equal and half of the overall population size (colony size). An employed bee updates her current source position in her memory and shares the information about the new source with onlooker bee. Based on the information, an onlooker bee explores her new neighbourhood position. In ABC, the search strategy for both employed bees and onlooker bees is directed by updating a random element in solution vector with other solution vector according to the following Eq. (3.6).

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \tag{3.6}$$

where v_{ij} is the upgraded decision variable at j^{th} dimension of i^{th} solution. Here the information is exchanged by j^{th} decision variable of i^{th} solution with j^{th} decision variable of k^{th} neighbouring solution. The term ϕ_{ij} is a random number between $[-1, 1]$.

Although this update strategy of ABC provides better exploration, it lacks in utilizing the information of the best solution. It is observed that in the functioning of the ABC algorithm proceeds different from other population based algorithms like GWO and PSO as it does not take advantage of the best solutions to lead the search process. This may result in decline of the convergence rate of the algorithm. It has been seen that the best solution information plays an important role to improving the convergence performance.

3.2.3 Chaotic Mapping and OBL Strategy.

The chaotic mapping and OBL strategies are also employed in the initialization phase of the proposed GWO-ABC algorithm. The chaotic mapping is described in Section 2.2.2 and the OBL strategy is explained in Section 2.2.3 of the previous chapter 2.

3.3 Proposed Hybrid GWO-ABC Algorithm

In literature, many researchers have conducted empirical studies on the performance of GWO and modified the algorithm to get balance between exploitation and exploration. Kishor and Singh [123], reported an authentic observations over performance of conventional GWO for different populations on diverse set of standard test functions. In this work, they specified some strengths and weaknesses of the GWO algorithm. In case of multimodal problems with many local optima, the algorithm may converge prematurely into some locally optimal solution avoiding further exploration. Along with this, it is observed that the GWO lacks in information sharing in the pack as the ω wolves only acts as a followers to first three best solutions and do not play role of an important individual in the pack. These limitations encourage us to adopt some strategies to modify exploration ability of algorithm.

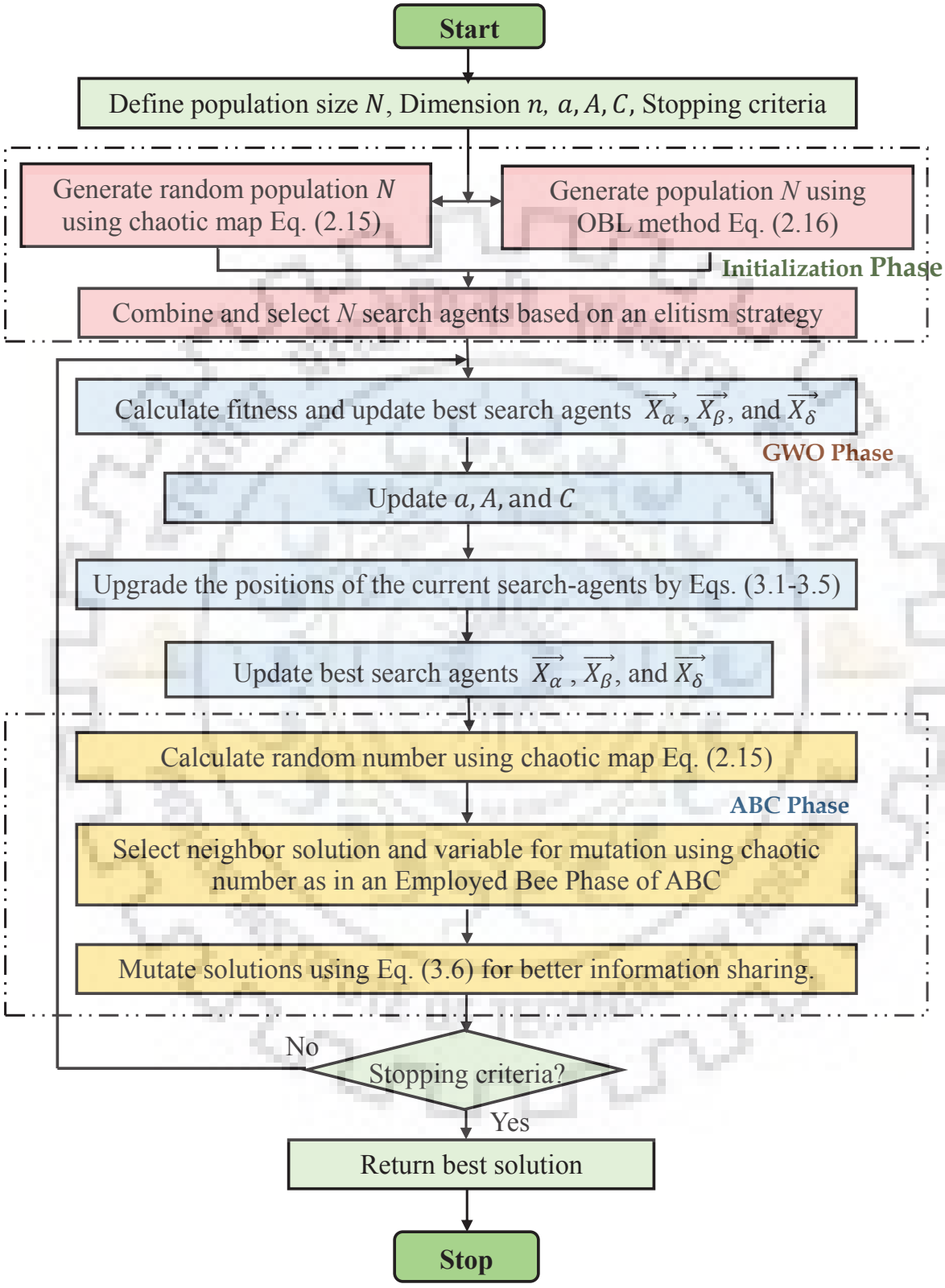


Fig. 3.1: Flowchart of the proposed GWO-ABC algorithm.

3.3.1 Motivation for Hybridization

Fundamentally, in conventional GWO, the adaptive values of parameters of $\overrightarrow{a}$ and $\overrightarrow{A}$ decide the transition between exploitation and exploration. Therefore, the candidate solutions' diverge from or converge towards the prey depends upon the values if $|\overrightarrow{A}| > 1$ or $|\overrightarrow{A}| < 1$, respectively. It is observed that, the convergence speed is reduced in the final iterations of the runs as it favours exploration in the early iterations and latter iterations are committed to exploitation only [123]. Further the primary observation in GWO is that, as it strictly follows dominant hierarchy based on leading wolves, hence, instead of sharing of information among all the individual candidate solutions, only three leading wolves (alpha, beta, and delta) decide and forward this information to all other individuals in the pack for further progress. In this manner, these best leading agents increase the convergence pressure and steer the search quickly in the vicinity of sub-optimal solution by loosing its diversity that leads to premature convergence. In other words, the GWO lacks diversity as it is directed by the three best leading search agents only. Primarily, it is reported in [123] that in case of unimodal test functions, initially the search advances rapidly towards the optimal solution but later relaxes because of the diversity problem. Consequentially, mostly in the case of multimodal problems, (where the search space possess many locally optimal solutions, prone to mislead the search) it gets stuck into vicinity of any such local sub-optima, losses diversity and faces up the problem of premature convergence.

Ultimately, this insight led us to suggest a modified and hybrid optimizer GWO-ABC, by introducing novel strategy which improves the information sharing among the pack of candidate solutions with each other, throughout all iterations.

3.3.2 Structure of the Proposed GWO-ABC Algorithm

The step-wise flowchart of the proposed GWO-ABC algorithm is presented in Fig. 3.1. As depicted in flowchart, all the steps in the GWO-ABC are same as the conventional GWO except that some extra strategies are included for population initialization and information sharing in the algorithm. To start the search process, initial parameters like population size N, dimension of solution space n, maximum allowed function evaluations (FE), which is also used as stopping criteria, are defined and other parameters like a, A,and C are calculated. After defining initial parameters, the proposed algorithm is executed in three phases, namely : Population initialization phase, GWO phase, and ABC phase as per their execution. The working principle of these three phases are explained below.

Population Initialization Phase : In the proposed algorithm, population initialization is implemented through chaotic mapping and OBL methodology to generate fitter initial candidate solutions from wider search space. The strategy adopted for population initialization is explained in Algorithm 2. As defined in steps 1 and 2, initial population $X \in |N|$ is obtained using random variable $ch(k)$ produced by Logistic chaotic map, within variable space defined by given bounds. The equation of logistic chaotic evolution function, used in this work, is as defined in Eq. (2.15).

Algorithm 2 Procedure for population initialization based on OBL and chaotic mapping.

Input : Population size of search agents N, dimension of solutions $j \in [n]$, upper and lower bounds of solutions $x_j \in [Ub_1, ..., Ub_n, Lb_1, \ldots, Lb_n]$, $k = 300$

Output : Initial population of size N

1: Initialize the population of size N using chaotic mapping in n dimensional space, $ch(k)$ is obtained using Eq. (2.15)

2: Thus, $X_i = (x_1, x_2, \ldots, x_n)$, where $i \in N$ and $x_j \in [Ub_j, Lb_j] \forall j \in [n]$ is defined by $x_j = Lb_j + ch(k) * (Ub_j - Lb_j)$

3: Obtain another set of population of size N using OBL

4: Such that, $X_i^* = (x_1^*, x_2^*, \ldots, x_n^*)$, where $i \in [N]$ and each new opposite solution is defined by $x_j^* = Lb_j + Ub_j - x_j \ \ \forall j \in [n]$

5: Combine both as $X = (X_i \cup X_i^*)$, $|X| = 2N$

6: Calculate fitness $f(X) = (f(X_i) \cup f(X_i^*))$ and sort accordingly

7: Select N fittest solutions.

Here also, the value of k is set to 300.

Further, another set of opposite population X^* is derived through OBL methodology using equations in step 4. The details of OBL is given in Section 2.2.3 in previous chapter. Thus, both the sets are combined as $X = (X_i \cup X_i^*) \in |2N|$ solutions and their fitness $f(X)$ is calculated. In next step, according to elitism principle, the fitness vector is sorted and first N fitter solutions are selected for further generations. Fig. 3.2 depicts the comparison between initial population generated by random distribution method used in GWO and the proposed population initialization scheme described in Algorithm 2. Here, the population size of $N = 100$ is generated for f_9 function with dimension $n = 3$ by both the methods. It can be observed that the initial population of candidate solutions, generated by the proposed scheme, is well distributed over the search space ensuing exploration capability. Further, the investigation of the statistical measure of diversity of the initial population is carried out for a test function f_1 with 10 dimension and 20 population size. The average of standard deviation (SD) for the values of each dimension is calculated and noted for different initialization schemes. The SD values for Chaotic initialization is 66.63906, Random initialization is 57.33381, Opposition Based Learning (OBL) based initialization is 67.99783, and Proposed Initialization Scheme is 73.84750. As the average value of SD for proposed population initialization scheme is highest among all initialization schemes. It shows that the diversity of the initial population generated by proposed scheme is more than other schemes and the initial seeds are widely spread.

GWO Phase : Once the initial population is generated, algorithm proceeds according to the conventional GWO and updates its parameters and current positions of search agents using Eqs. (3.1) to (3.5).

ABC phase : As discussed above, bees (both employed and onlooker) in ABC share infor-

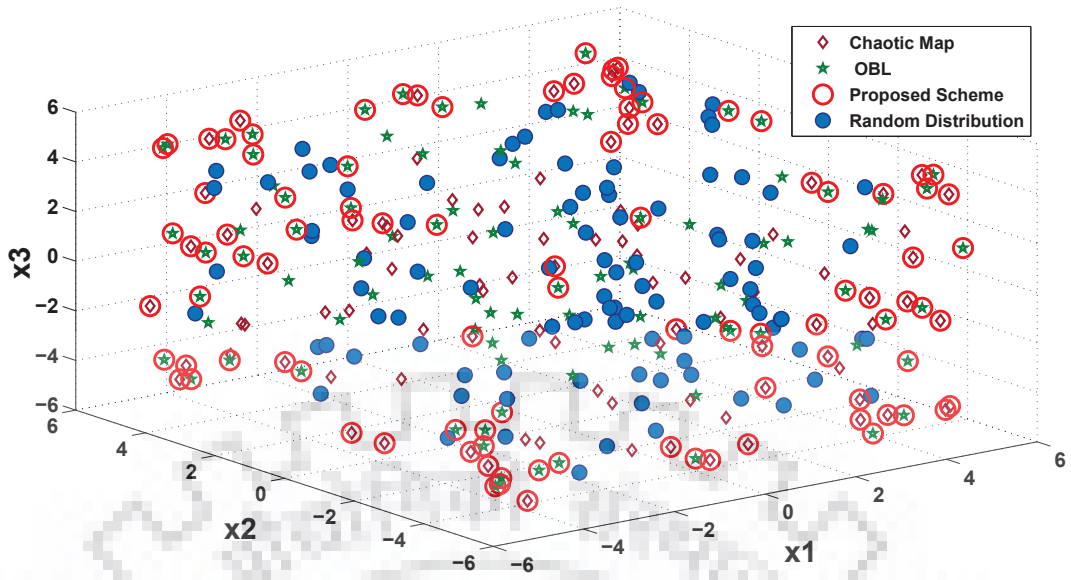


Fig. 3.2: Comparative illustrations of initial population by the proposed population initialization scheme against random distribution scheme.

mation among the candidate solutions in the pack and modify old solutions using Eq. (3.6). To elevate the randomness and non-repetition, logistic chaotic mapping defined in Eq. (2.15), is used instead of uniform random generator for defining ϕ in Eq. (3.6).

This gives better exploration opportunity by selecting arbitrary neighbouring solutions and location for information exchange.

The whole procedure, i.e., GWO Phase and ABC phase, is carried out for defined number of FE and then the best solution is returned as a result. By the search equation of ABC, i.e., Eq. (3.6) in GWO, the global search ability is ameliorated since every member of the pack gets opportunity to share information with other member. Thus, it helps to maintain necessary exploration, exploitation, and alleviates the problem of diversity and elude from premature convergence.

3.4 Simulation Results and Discussion

In this section, the performance of GWO-ABC is evaluated on various performance metrics, such as : ranking, statistical test, exploitation and exploration analysis, and convergence curve behaviour.

3.4.1 Performance Evaluation on Test Functions

The performance of the proposed GWO-ABC algorithm has been investigated on a test bed of 21 classical benchmark functions [89] and 6 composite functions (CEC 2014 [124]), extensively utilized by many researchers in their recent studies [33, 105, 125]. All these functions are minimization functions. This test bed is comprised of 7 unimodal (f_1 to f_7), 6 multimodal (f_8 to f_{13}), and 8 fixed-dimension multimodal benchmark functions (f_{14} to f_{21}), and a detailed description

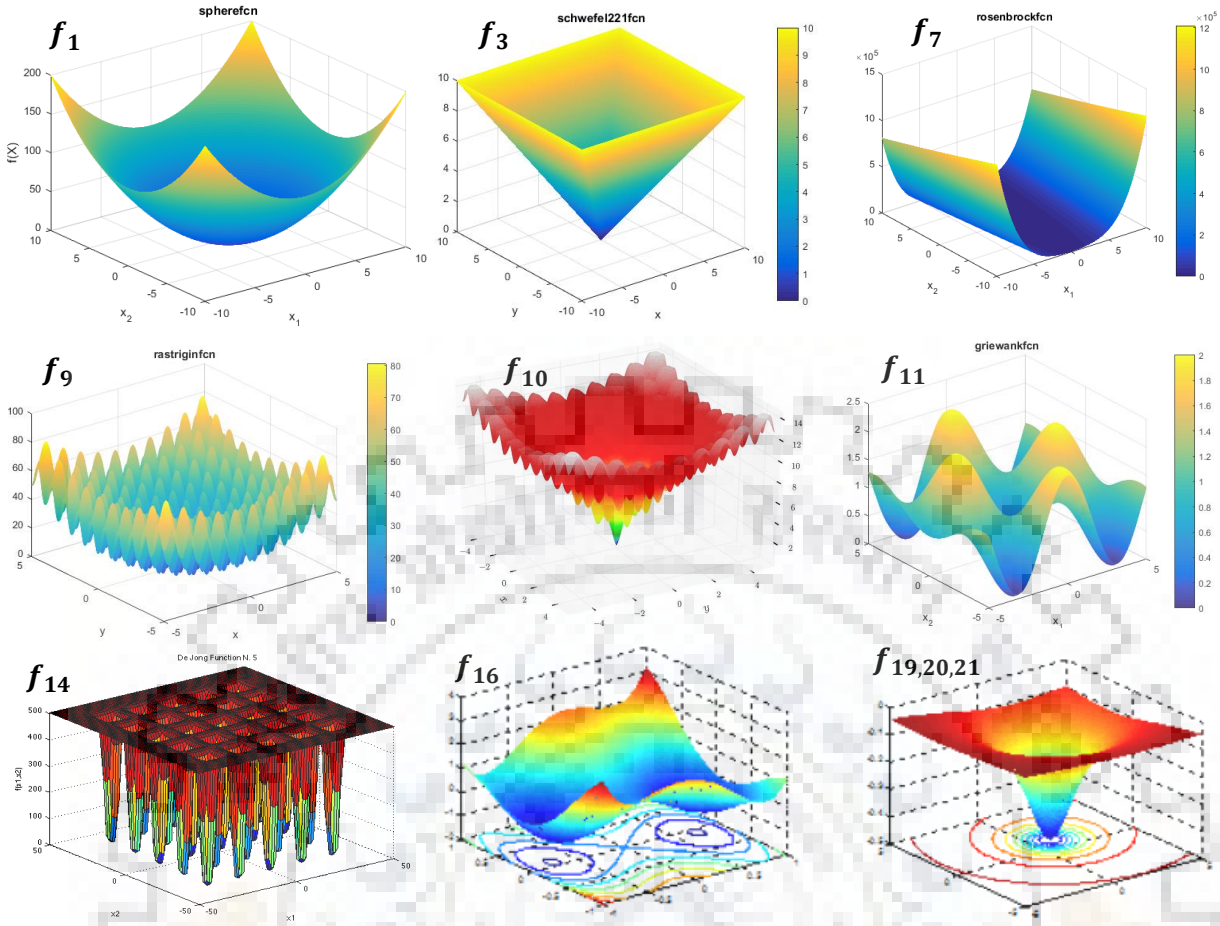


Fig. 3.3: 3D plots of some benchmark functions.

of these are given in appendix Tables A.3 and A.4. For further experimentation, 6 composite functions from CEC2014 test bed (f_{22} to f_{27}), as given in Table A.2 are also investigated. The 3D plots of some of the benchmark functions are demonstrated in Fig. 3.3. Here, we can observe that there are multiple suboptimal points and single optimal solution in the case of multimodal functions. Hence there are chances of stagnation at any suboptimal point.

For significant analysis, the average of best results (mean) and standard deviation (SD) of GWO-ABC and other algorithms for 100 independent runs are obtained and reported in Tables 3.3, 3.4, 3.5, and 3.6 for unimodal, multimodal, fixed-dimension multimodal, and composite test functions, respectively. The results of every function obtained by corresponding algorithms are sorted according to their mean values and ranks are assigned in accordance with the comparison of results. Further, the average rank of particular algorithm is calculated and overall rank is decided on the basis of average rank. Finally, Wilcoxon rank-sum test [126] with 1% significance level is also employed for performing the statistical analysis. The parameter settings used for experimental setup is explained below.

Table 3.1: Parameter settings of different algorithms used in this study.

Algorithm	Description of parameters	Population	Maximum FE
PSO [87]	$c_1, c_2 = 2, w_{max} = 0.9, w_{min} = 0.2$	30	30000
GWO [89]	$a_1 = 2$	30	30000
ABC [90]	$limit = (N \times n)/2$	30	30000
GSA [98]	$\alpha = 20,\ G_0 = 100$	30	30000
ALO [91]	–	30	30000
GWO-ABC	$a_1 = 2$	30	30000
for Composite functions		50	50000

3.4.2 Parameter Settings

The GWO-ABC algorithm is executed for 100 independent runs over test functions. For f_1 to f_{21}, the population of each algorithm is fixed to 30. The dimension size of unimodal and multimodal benchmark functions is fixed to 30. The maximum number of allowed FE are 30000, which acts as a stopping criteria for all the algorithms. For composite functions f_{22} to f_{27}, the population size is fixed to 50. For these functions, the allowed FE are 50000. The other significant parameters for all algorithms used for comparison are same as their original articles and are listed in Table 3.1.

3.4.3 Statistical Analysis (Wilcoxon Rank Sum Test)

For comprehensive study, statistical tests are entailed [126] to quantify the performance in terms of statistical significant difference between results obtained by different algorithms. The statistical analysis of GWO-ABC against original GWO and other algorithms is done with Wilcoxon rank-sum test [126] with 1% significance level. As considered by many researchers, the samples are taken from final objective function values obtained by every algorithm. It tests the null hypothesis based on p –values calculated using Wilcoxon rank-sum method. The following symbols are used to represent the grading (ranking) of algorithms.

1. $(+)$:- If p –value < 0.01 then statistical difference is very significant and the performance of GWO-ABC is superior to the performance of the other algorithms.

2. $(\approx)$:- If p –value ≈ 0.01 then statistical difference is negligible and the performance of GWO-ABC is similar to the performance of the other algorithms.

3. $(-)$:- If p –value > 0.01 then statistical difference is not significant and the performance of GWO-ABC is inferior to the performance of the other algorithms.

Table 3.2: *p*–values and *grades* obtained by statistical Wilcoxon rank-sum test with 1% significance level.

	GWO		PSO		ABC		GSA		ALO	
$f\downarrow$	*p*–value	*grade*	*p*–value	*grade*	*p*–value	*grade*	*p*–value	*grade*	*p*–value	*grade*
f_1	0.00970	$+$	0.00658	$+$	0.00451	$+$	$2.80E-05$	$+$	0.00026	$+$
f_2	0.00456	$+$	0.00269	$+$	$4.51E-06$	$+$	$1.41E-06$	$+$	0.00083	$+$
f_3	0.00498	$+$	$3.84E-05$	$+$	$4.33E-10$	$+$	$7.11E-07$	$+$	$1.86E-05$	$+$
f_4	0.01152	$\approx$	$3.45E-05$	$+$	$1.82E-08$	$+$	0.00710	$+$	$2.92E-05$	$+$
f_5	0.00369	$+$	0.12916	$-$	$6.50E-07$	$+$	0.01080	$\approx$	0.03920	$-$
f_6	0.27945	$-$	0.01344	$\approx$	0.00018	$+$	0.00019	$+$	0.00018	$+$
f_7	0.01103	$\approx$	$2.36E-05$	$+$	$3.04E-06$	$+$	$4.92E-19$	$+$	$1.06E-05$	$+$
f_8	0.00115	$+$	$7.40E-10$	$+$	$1.11E-09$	$+$	$2.32E-03$	$+$	0.16641	$-$
f_9	0.00361	$+$	$9.80E-07$	$+$	0.00053	$+$	$6.10E-07$	$+$	$1.49E-09$	$+$
f_{10}	0.00033	$+$	$3.02E-24$	$+$	$5.60E-06$	$+$	$6.47E-10$	$+$	0.00016	$+$
f_{11}	0.01051	$\approx$	0.00546	$+$	$2.50E-27$	$+$	$5.69E-07$	$+$	0.00232	$+$
f_{12}	0.01015	$\approx$	0.01127	$\approx$	0.00022	$+$	0.01171	$\approx$	$3.42E-06$	$+$
f_{13}	0.00301	$+$	$2.01E-06$	$+$	$1.14E-06$	$+$	$1.47E-06$	$+$	0.18640	$-$
f_{14}	0.00416	$+$	$3.21E-28$	$+$	0.17171	$-$	0.00331	$+$	0.10393	$-$
f_{15}	0.01136	$\approx$	$4.25E-08$	$+$	$2.40E-06$	$+$	$2.95E-07$	$+$	0.01188	$\approx$
f_{16}	0.00942	$+$	0.01312	$\approx$	0.13910	$-$	$3.12E-06$	$+$	$8.30E-14$	$+$
f_{17}	0.00015	$+$	$2.12E-08$	$+$	$1.11E-05$	$+$	$2.23E-10$	$+$	$3.58E-08$	$+$
f_{18}	0.00123	$+$	0.04025	$-$	0.00734	$+$	0.00402	$+$	$1.02E-03$	$+$
f_{19}	0.01012	$\approx$	0.00333	$+$	$7.99E-07$	$+$	0.00250	$+$	0.00096	$+$
f_{20}	0.00172	$+$	0.09231	$-$	0.00331	$+$	0.0001	$+$	0.04231	$-$
f_{21}	0.00017	$+$	0.16341	$-$	0.01456	$\approx$	0.00391	$+$	0.00012	$+$
f_{22}	0.00376	$+$	0.01014	$\approx$	0.00056	$+$	0.00331	$+$	0.05234	$-$
f_{23}	0.00331	$+$	0.01342	$\approx$	0.01155	$\approx$	0.00881	$+$	$9.56E-07$	$+$
f_{24}	0.00071	$+$	0.00016	$+$	$3.22E-05$	$+$	0.00224	$+$	0.00098	$+$
f_{25}	$5.56E-05$	$+$	0.02134	$-$	0.01248	$\approx$	$2.16E-04$	$+$	0.00012	$+$
f_{26}	0.01245	$\approx$	0.04546	$-$	0.01223	$\approx$	0.00934	$+$	$1.24E-07$	$+$
f_{27}	0.003845	$+$	0.076413	$-$	0.00206	$+$	0.032291	$-$	$2.77E-04$	$+$

The p –values and their grades are represented in Table 3.2. Also, the summary of this statistical analysis is reflected in Tables 3.3, 3.4, 3.5, and 3.6 in terms of above symbols and grades for each algorithm. The final statistical performance of each algorithm is presented in last row of every table. From this, it is easy to make the quantitative decision about the performance of the proposed GWO-ABC algorithm. The detailed statistical analysis of all the functions is explained in three ensuing sections.

Table 3.3: Comparison of results obtained for the unimodal benchmark functions.

f		GWO	PSO	ABC	GSA	ALO	GWO-ABC
f_1	Mean	$2.25E-57$	$1.06E-08$	$5.83E-11$	$1.13E-16$	$8.41E-06$	**$3.65E-81$**
	SD	$2.13E-56$	$4.75E-09$	$5.33E-11$	$5.02E-17$	$5.06E-06$	**$6.23E-78$**
	Rank(Grade)	$2\,(+)$	$5\,(+)$	$4\,(+)$	$3\,(+)$	$6\,(+)$	**1**
f_2	Mean	$8.35E-36$	$2.36E-04$	$2.14E-07$	$5.5E-08$	48.95	**$2.98E-46$**
	SD	$7.23E-35$	$2.36E-04$	$4.01E-06$	$1.66E-08$	52.76	**$2.20E-45$**
	Rank(Grade)	$2\,(+)$	$5\,(+)$	$4\,(+)$	$3\,(+)$	$6\,(+)$	**1**
f_3	Mean	$7.13E-15$	14.194	$1.25E+04$	$5.15E-02$	$1.18E+03$	**$1.24E-24$**
	SD	$1.44E-14$	6.224	$2.35E+03$	145.26	498.32	**$4.69E-23$**
	Rank(Grade)	$2\,(+)$	$4\,(+)$	$6\,(+)$	$3\,(+)$	$5\,(+)$	**1**
f_4	Mean	$1.23E-10$	1.5647	23.11456	1.04778	10.5229	**$1.37E-21$**
	SD	$2.15E-09$	0.2351	3.5689	1.09421	4.7058	**$1.66E-22$**
	Rank(Grade)	$2\,(\approx)$	$4\,(+)$	$6\,(+)$	$3\,(+)$	$5\,(+)$	**1**
f_5	Mean	26.0127	49.5689	3.98754	**$6.49E-07$**	83.1839	24.126
	SD	0.001998	65.8352	5.1153	**$8.69E-11$**	90.68	1.0042
	Rank(Grade)	$4\,(+)$	$5\,(-)$	$2\,(+)$	**$1\,(\approx)$**	$6\,(-)$	3
f_6	Mean	0.96456	**$1.57E-08$**	$3.12E-03$	$1.15E-06$	$7.99E-06$	0.34556
	SD	0.112456	**$2.15E-08$**	$1.54E-03$	$3.15E-17$	$5.02E-06$	0.012441
	Rank(Grade)	$6\,(-)$	**$1\,(\approx)$**	$4\,(+)$	$2\,(+)$	$3\,(+)$	5
f_7	Mean	0.00056	0.05466	0.23418	2.7883	0.08601	**$3.12E-5$**
	SD	0.0014	0.03124	0.01234	0.03406	0.03379	**$1.23E-4$**
	Rank(Grade)	$2\,(\approx)$	$3\,(+)$	$5\,(+)$	$6\,(+)$	$4\,(+)$	**1**
Average Rank		2.85	3.86	4.42	3.0	5.0	1.85
Overall Rank		2	4	5	3	6	**1**
Grades +/≈ /-		$4/2/1$	$5/1/1$	$7/0/0$	$6/1/0$	$6/0/1$	**$28/4/3$**

The "best results" are indicated by bold values.

3.4.4 Exploitation Analysis (Results for Unimodal Test Functions)

As given in Table A.3, the functions f_1 to f_7 are unimodal functions having only one optima i. e. global optima, therefore, these functions are used to test the exploitation capability of an algorithm. Hence, the overall exploitation behaviour of the proposed algorithm can be investigated

Table 3.4: Comparison of results obtained for the multimodal benchmark functions.

f		GWO	PSO	ABC	GSA	ALO	GWO-ABC
f_8	Mean	-6102.33	-4671.12	-7105.93	-4456.02	-5779.93	$\mathbf{-8146.67}$
	SD	855.2340	**234.1245**	162.2757	1652.223	1076.820	460.1560
	Rank(Grade)	$3(+)$	$5(+)$	$2(+)$	$6(+)$	$4(-)$	**1**
f_9	Mean	$1.12E-16$	49.07340	1.452810	25.57040	82.58130	**0**
	SD	$1.23E-12$	15.11290	0.584178	7.113900	11.50790	**0**
	Rank(Grade)	$2(+)$	$5(+)$	$3(+)$	$4(+)$	$6(+)$	**1**
f_{10}	Mean	$2.56E-12$	19.12590	$2.34E-04$	$8.04E-09$	1.647401	$\mathbf{1.16E-17}$
	SD	$1.62E-15$	0.068790	$8.23E-05$	$1.02E-09$	0.931450	$\mathbf{2.45E-15}$
	Rank(Grade)	$2(+)$	$6(+)$	$4(+)$	$3(+)$	$5(+)$	**1**
f_{11}	Mean	0.002145	0.012173	2966344	8.416200	0.009890	**0**
	SD	0.003567	0.025648	4347.167	2.322601	0.008870	**0**
	Rank(Grade)	$2(\approx)$	$4(+)$	$6(+)$	$5(+)$	$3(+)$	**1**
f_{12}	Mean	0.051321	0.042115	$\mathbf{7.14E-12}$	0.012440	11.55400	0.021096
	SD	0.012101	0.0110213	$\mathbf{1.56E-11}$	0.012147	3.634900	0.011345
	Rank(Grade)	$5(\approx)$	$4(\approx)$	$\mathbf{1}(+)$	$2(\approx)$	$6(+)$	3
f_{13}	Mean	0.645225	0.003336	$\mathbf{3.10E-09}$	0.003290	3.885000	0.124562
	SD	0.123454	0.004207	$\mathbf{3.22E-10}$	0.005300	12.19740	0.002340
	Rank(Grade)	$5(+)$	$3(+)$	$\mathbf{1}(+)$	$2(+)$	$6(-)$	4
Average Rank		3.16	4.5	2.83	3.67	5.0	1.83
Overall Rank		3	5	2	4	6	**1**
Grades +/≈/-		$4/2/0$	$5/1/0$	$6/0/0$	$5/1/0$	$4/0/2$	$\mathbf{24/4/2}$

The "best results" are indicated by bold values.

from the mean and SD values obtained by algorithms are reported in Table 3.3. It can be observed that, for f_1 to f_7, the proposed GWO-ABC outperforms the conventional GWO. Similarly, the results by GWO-ABC are superior to that of PSO, ABC, GSA, and ALO in the 5 out of 7 test functions (f_1, f_2, f_3, f_4, and f_7). Also, the low SD values indicate the stability and robustness of the proposed algorithm. Thus, It can be inferred that the adopted modifications in GWO-ABC have enriched the exploitation behaviour of conventional GWO.

The ranks are assigned in accordance with mean values obtained over each function. It can be clearly perceived from Table 3.3 that, GWO-ABC is ranked one in 5 out of 7 functions and therefore overall ranking is best (rank 1) among all algorithms. Thus, the overall rank of GWO-ABC indicates that the proposed algorithm outperforms conventional GWO, PSO, ABC, GSA, and ALO algorithms. According to the last row of Table 3.3, GWO-ABC is statistically superior to other algorithms in 28 cases (91%) and similar in 4 cases. By incorporating information sharing strategy in the pack features the advantage of recognizing newly explored candidate solutions in new area of search space. This provides the solutions from promising regions of the search space

faster and enhance the exploitation to nearby newly explored solutions. Eventually, it can be observed that the proposed modifications in conventional GWO have ameliorated the exploitation capability in dealing with unimodal function.

3.4.5 Exploration Analysis (Results on Multimodal Test Functions)

Multimodal test functions (f_8 to f_{21}) are opted to validate the exploration capability of an optimization algorithm, as these test functions have many optima which include many local optima and single global optima. The average (mean) and standard deviation (STD) results over 100 independent runs of GWO-ABC and other methods for multimodal (f_8 to f_{13}) and fixed dimension multimodal (f_{14} to f_{21}) test functions are noted in Tables 3.4 and 3.5, respectively. It can be observed from Tables 3.4 and 3.5 that GWO-ABC obtains the superior results compared to GWO, PSO, ABC, GSA, and ALO algorithms for f_9, f_{10}, f_{11}, f_{15}, f_{16}, f_{17}, f_{18}, f_{19}, f_{20}, and f_{21}. As per expectation, it is worth to notice that GWO-ABC outperforms conventional GWO in all multimodal (f_8 to f_{21}) functions. It can also be seen that the proposed GWO-ABC is ranked one in 11 out of 14 multimodal functions. Hence, the overall ranking of the GWO-ABC is superior to all other algorithms. Based on SD index, it can be claimed that GWO-ABC provides better accuracy in comparison to others.

In summary, it can be concluded that the exploration ability of conventional GWO is enhanced by introducing information sharing strategy of ABC. The chaotic random selection of neighbouring solutions intensifies the exploratory behaviour of GWO. In this manner, the new strategies applied in proposed algorithm have assisted GWO in maintaining a significant balance between local and global search inclinations. The enhanced exploration behaviour intensifies the capability of GWO to escape from local optima in case of multimodal test functions. Remarkably, the statistical test results, in last row of Tables 3.4 and 3.5, also indicate that the GWO-ABC is superior to other methods in approximately 88% of evaluations. This proves that there is significant statistical difference in the results obtained by GWO-ABC and other algorithms.

3.4.6 Convergence Analysis

The convergence behaviour of a proposed GWO-ABC algorithm is analyzed by convergence graph for some functions in Figs. 3.4 and 3.5. The mean objective function value profile (in logarithmic scale) over 100 independent runs for each function is plotted on Y-axis against increasing number of FEs on X-axis. For comparative analysis, the convergence graph of proposed algorithm is exhibited with GWO, PSO, ABC, GSA, and ALO.

Fig. 3.4 shows the convergence graphs of algorithms for five unimodal functions (f_1, f_2, f_3, f_4, and f_7). It can be clearly seen that the proposed GWO-ABC shows better convergence behaviour from the initial iterations itself and improves further with increase in iterations. This behaviour is attributed to the fact that information sharing strategy adopted in GWO-ANBC escalates the convergence speed.

Likewise, the convergence graphs of three multimodal functions (f_9, f_{10}, and f_{11}) along with three fixed dimensional multimodal functions (f_{15}, f_{18}, and f_{21}) are demonstrated in Fig. 3.5. The multimodal functions have many local optimal points, hence there are more chances of stagnation at any local optimal point. Therefore, the algorithm should possess a strategy to explore search space efficiently to avoid entrapment in these local optimal regions. As discussed above, the proposed algorithm adopts the strategies to enhance the search ability and achieve global optimal point with better convergence rate. Therefore, in most of the functions we can realize the accelerated trends in convergence curves of GWO-ABC compared to conventional GWO and other algorithms.

3.4.7 Results for Composite Test Functions

The investigation results on Composite test functions for 10 dimension are demonstrated in Table 3.6. These functions are taken from the CEC'14 benchmark test suits [124] to evince the efficacy of the proposed GWO-ABC. Among these, for function f_{30}, the GWO-ABC shows poor result than PSO, but it outperform GWO and other algorithms. The lower SD values show that GWO-ABC has smooth distribution than the conventional GWO. Also, for Most of the functions like f_{22} to f_{29}, the proposed algorithm indicates efficient improvements in results than other algorithms. The challenging task in case of optimizing composite functions is to overcome stagnation at local optima, the information sharing strategy adopted in GWO-ABC efficiently handle this by balancing the exploitation and exploration. The overall ranking (1^{st}) and statistical simulation grade manifest the performance of GWO-ABC. The performance of GWO-ABC, in 25 out of 30 (83%) functions, is better than or commensurable to the performance of other algorithms. To observe the convergence behaviour of GWO-ABC, we investigate f_{24} i. e. $CF3$ and plot the convergence curves in Fig. 3.4 (f). It is clearly visible that the convergence behaviour of GWO-ABC is much faster than all other algorithms.

Thus, in most of the functions, it is clearly visible that GWO-ABC outperforms GWO, PSO, ABC, GSA, and ALO algorithms in terms of in convergence speed, which reflects the impact of adoption of search strategy of bees in GWO.

Table 3.5: Comparison of results obtained for the fixed-dimension multimodal benchmark functions.

f		GWO	PSO	ABC	GSA	ALO	GWO-ABC
f_{14}	Mean	5.458330	22.09483	0.998004	3.024601	1.295820	**0.998001**
	SD	5.422380	0.001200	**1.30E − 16**	1.819220	0.669810	0.836571
	Rank(Grade)	4 (+)	5 (+)	1 (−)	3 (+)	2 (−)	**1**
f_{15}	Mean	0.000390	0.000791	0.000690	0.002040	0.002780	**0.000301**
	SD	$5.15E-07$	$8.16E-05$	0.000217	0.000445	0.006180	**2.44E − 08**
	Rank(Grade)	2 (≈)	4 (+)	3 (+)	5 (+)	6 (≈)	**1**
f_{16}	Mean	−1.03162	−1.03162	−1.03162	−1.03161	−1.01160	**−1.03160**
	SD	$1.12E-10$	$1.23E-15$	0	0	$8.31E-14$	**0**
	Rank(Grade)	1 (+)	1 (≈)	1 (−)	1 (+)	1 (+)	**1**
f_{17}	Mean	0.397839	0.397012	0.397901	0.397012	0.397901	**0.397801**
	SD	$1.45E-05$	0	0.004500	$1.45E-05$	0	**0**
	Rank(Grade)	2 (+)	3, (+)	2 (+)	3, (+)	2 (+)	**1**
f_{18}	Mean	3.539001	3.000000	3.002356	3.000000	3.000100	**3.000000**
	SD	39.83322	**4.11E − 17**	0.006215	$4.12E-15$	0.040200	$8.29E-08$
	Rank(Grade)	3 (+)	1 (−)	2 (+)	1 (+)	1 (+)	**1**
f_{19}	Mean	−9.647	−6.75649	−10.15317	−9.7119	−6.11400	**−10.1510**
	SD	0.023145	3.321304	**0.000100**	3.824420	2.949000	0.000146
	Rank(Grade)	3 (≈)	4 (+)	2 (+)	3 (+)	5 (+)	**1**
f_{20}	Mean	−9.76150	−9.222455	−10.40317	−8.243	−8.53567	**−10.4028**
	SD	2.554682	1.256470	**0.002309**	0.001230	3.029300	0.008945
	Rank(Grade)	3 (+)	2 (−)	1 (+)	4 (+)	5 (−)	**1**
f_{21}	Mean	−9.998451	−9.756483	−9.965840	−10.5360	−5.69750	**−10.5311**
	SD	1.542310	3.221450	0.037870	0.000070	2.623400	**0.000259**
	Rank(Grade)	3 (+)	4, (−)	3 (≈)	2, (+)	5 (+)	**1**
Average Rank		2.62	3.0	1.87	2.75	3.37	1
Overall Rank		3	5	2	4	6	**1**
Grades +/≈/-		6/2/0	4/1/3	5/1/2	8/0/0	5/1/2	**28/5/7**

The "best results" are indicated by bold values.

Table 3.6: Comparison of results obtained for the composite benchmark functions

f		GWO	PSO	ABC	GSA	ALO	GWO-ABC
f_{22}	Mean	2692.011	2690.181	2690.936	4456.021	2689.012	**2681.925**
CF1	SD	22.06500	27.56001	162.2750	1652.223	243.5560	**7.766001**
	Rank(Grade)	$4\,(+)$	$3\,(\approx)$	$3\,(+)$	$5\,(+)$	$2\,(-)$	**1**
f_{23}	Mean	2796.813	2760.140	2768.956	2896.321	2789.889	**2710.135**
CF2	SD	24.90010	0.073710	0.584178	7.113900	**0.010900**	29.96000
	Rank(Grade)	$5\,(+)$	$2\,(\approx)$	$3\,(\approx)$	$6\,(+)$	$4\,(+)$	**1**
f_{24}	Mean	2990.400	2974.200	2981.920	3059.108	3018.976	**2959.108**
CF3	SD	161.3200	140.8010	144.2610	189.6650	144.2350	**73.94400**
	Rank(Grade)	$4\,(+)$	$2\,(+)$	$3\,(+)$	$6\,(+)$	$5\,(+)$	**1**
f_{25}	Mean	3265.809	3360.890	3456.670	3390.213	3296.768	**3257.192**
CF4	SD	69.22000	79.39000	**64.09600**	72.32200	292.1050	78.85100
	Rank(Grade)	$2\,(+)$	$4\,(-)$	$6\,(\approx)$	$5\,(+)$	$3\,(+)$	**1**
f_{26}	Mean	$4.78E+05$	$2.54E+05$	$9.30E+05$	$7.63E+06$	$8.03E+08$	**3675.897**
CF5	SD	$8.63E+05$	$9.39E+05$	$1.70E+06$	$6.23E+06$	$1.52E+06$	**1067.211**
	Rank(Grade)	$3\,(\approx)$	$2\,(-)$	$4\,(\approx)$	$5\,(+)$	$6\,(+)$	1
f_{27}	Mean	3957.947	**3899.831**	3969.550	4356.178	4428.688	3950.060
CF6	SD	329.5320	399.9910	443.9710	564.2350	1433.256	**244.4200**
	Rank(Grade)	$3\,(+)$	**1**$\,(-)$	$4\,(+)$	$5\,(-)$	$6(+)$	2
Average Rank		3.5	2.33	3.83	5.33	4.33	1.16
Overall Rank		3	2	4	6	5	**1**
Grades +/≈/-		$5/1/0$	$1/2/3$	$3/3/0$	$5/0/1$	$5/0/1$	**19/6/5**

The "best results" are indicated by bold values.

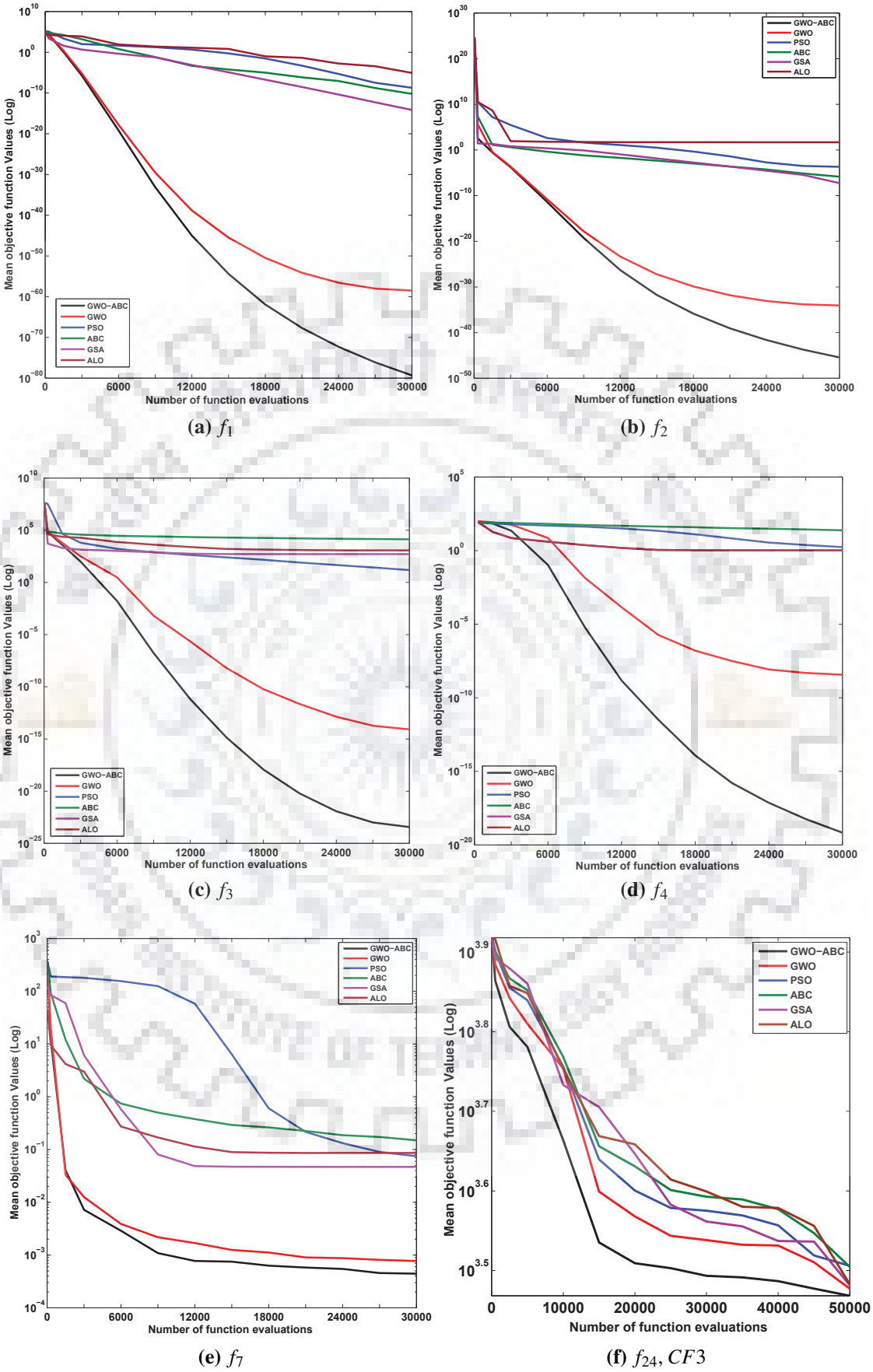
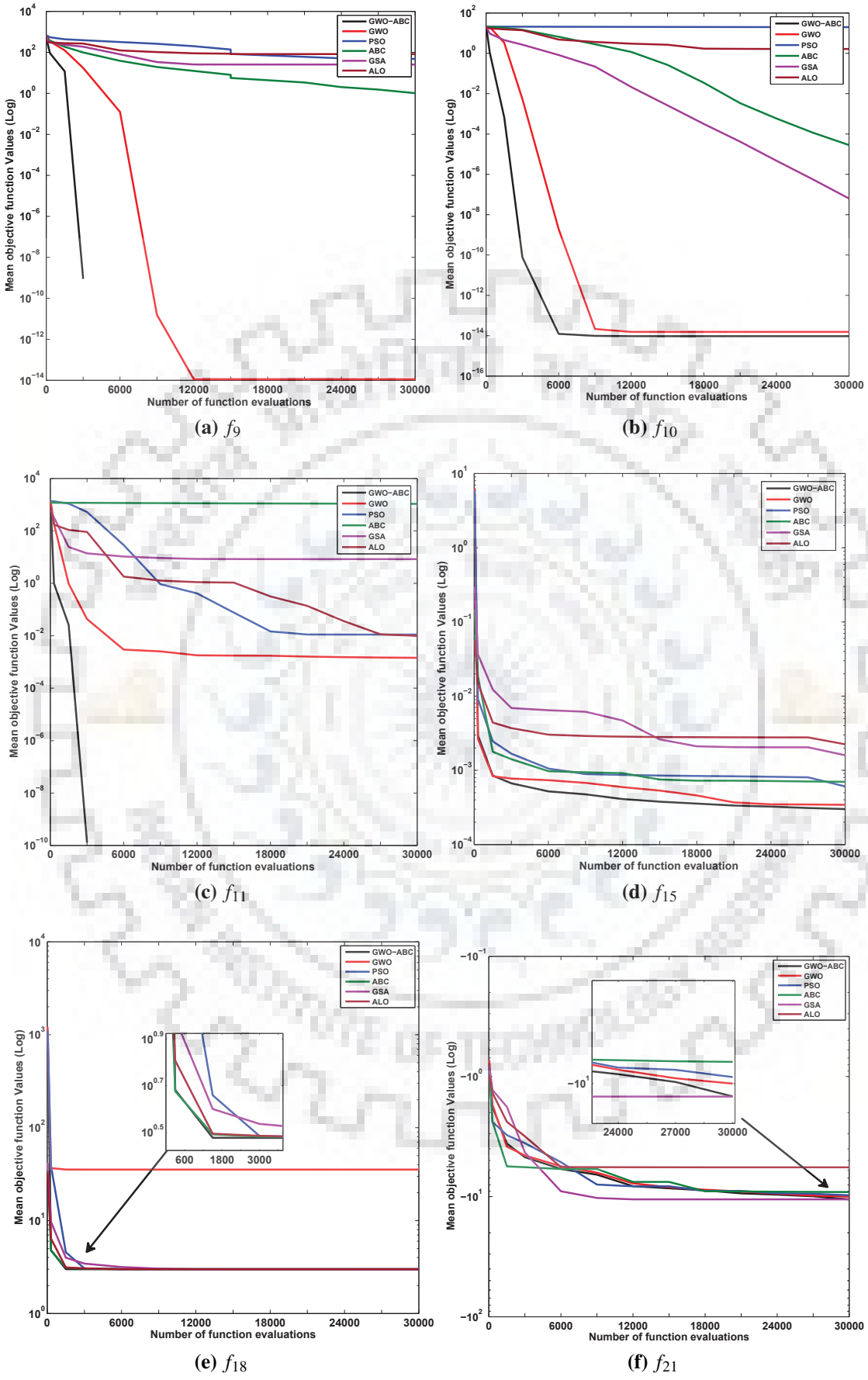


Fig. 3.4: Comparison of convergence curves of proposed GWO-ABC and other algorithms obtained for some of the unimodal and composite function.



(a) f_9 **(b)** f_{10} **(c)** f_{11} **(d)** f_{15} **(e)** f_{18} **(f)** f_{21}

Fig. 3.5: Comparison of convergence curves of proposed GWO-ABC and other algorithms obtained for some of the multimodal and fixed-dimension multimodal functions.

3.5 Concluding Remarks

The accomplishment of proposed GWO-ABC has been substantiated through exploitation and exploration capability analysis, convergence rate analysis, and non-parametric Wilcoxon rank-sum test. The statistical analysis and evolution convergence curves exhibit the outstanding performance of proposed algorithm to other counterparts within competitive computational complexity. Finally, from results and discussions in previous section, following decisive comments can be concluded as prime attributes of GWO-ABC.

- The GWO-ABC can still manifest the prime features of original GWO with improved exploitation and exploration tendencies.
- The elitism based population initialization, incorporating chaotic mapping and OBL strategy, generates well diverse (i.e. guess and its opposite guess) fitter starting candidate solutions.
- Hybridizing with employed bee's upgrading strategy from ABC elevates the information sharing ability among the members of the pack. It intensifies exploration tendency and boost the participation of all candidate solutions in the pack without affecting the leadership hierarchy approach of conventional GWO.
- As no extra FE are required in the proposed GWO-ABC algorithm, the performance is enhanced in similar computational time as in conventional GWO.

For the future work, the proposed modified algorithm can be validated for interdisciplinary and multi domain complex engineering and science optimization problems.

Chapter 4

Optimization of Different Controller Design Problems

This chapter is aimed to test the performance of proposed algorithms for optimization of different controller design problems. Initially, Section 4.1 briefs the qualification of MAs for control design problems. In Section 4.2, the proposed GWO-ABC algorithm is applied on four test bench process plants to obtain the optimal time-domain specifications. Further, improved CFGWO algorithm is presented in Section 4.3 and examined for optimal tuning of controller parameters for trajectory tracking problems of a 2-link robotic manipulator in Section 4.4. Finally, the conclusions are drawn.

4.1 Introduction

The tuning of controllers is considered as a high-dimensional, complex, multimodal numerical optimization problem, as many locally optimal solutions can be obtained for different combinations of the parameter values [42, 43]. Thus, it is always a challenging task for designers to get the global optimal value. Recently, MAs are widely used for obtaining the optimal tuning parameters of controllers to get the best performance and robust response [31, 32]. In view of this, this chapter is dedicated to reveal the efficacy of MAs over control system design problems for various linear and non-linear test bench process plants with and without time delay, for unit step response. Initially, the GWO-ABC algorithm, proposed in previous chapter, is applied to activate the optimal time-domain specifications. All the design requirements like low overshoot, better rise time, faster settling time, minimum steady-state error, and performance index are evaluated and compared to other counterparts.

The work outlined in this chapter has been disseminated in the following publications:

- P. J. Gaidhane and M. J. Nigam, "A hybrid grey wolf optimizer and artificial bee colony algorithm for enhancing the performance of complex systems,"*Journal of Computational Science*, vol. 27, pp. 284-302, Jul 2018.
- P. J. Gaidhane and M. J. Nigam, " A rational cooperative foraging based grey wolf optimizer and its application to optimize complex systems", *Applied Soft Computing*, 2018. (Under review)

Further, the communication signalling strategy used in cooperative foraging is incorporated in conventional GWO with continuing its leadership hierarchy approach to present improved CFGWO algorithm. Moreover, for testing the performance of the proposed CFGWO for real world applications, the proposed CFGWO algorithm is examined for optimal tuning of controller parameters for trajectory tracking problems of a 2-link robotic manipulator. The comparative graphs of trajectory tracking performance, the path traced by the end-effector, and X and Y coordinate versus time variations against their desired reference curves are illustrated. Also, plots of position errors and controller output for both the links are presented.

4.2 Implementation of GWO-ABC on Test-bench Process Plants

Several studies have been reported that FOPID outperforms conventional PID controllers in terms of robustness over wide range of applications including robotics in recent years [79, 80]. The fractional operators are recommended for the availability of additional freedom to the designers. On the other hand, every added parameter increases the complexity of tuning methodology used for designing. The suitable optimization algorithms with high convergence rate and good exploration behaviour are utilized in such problems. The inherent complexity of control design problems led us to select the proposed GWO-ABC algorithm for tuning the control parameters. Here, optimization procedure is performed to achieve the optimal time-domain specifications by minimizing underlying objective function. FOPID control parameters are tuned to get minimum overshoot M_p, lower rise time t_r, faster settling time t_s. and null steady state error E_{ss}.

4.2.1 FOPID Controller

The FOPID controller used in this study is demonstrated in Fig. 4.1. In this design, the fractional order integro-differential operators are incorporated with classical PID controllers to enhance the dynamic response and to reduce the steady-state error. The definition and approximation method for the fractional order are detailed in Section 2.1.4. The FOPID transfer function used in this experimentation is represented as

$$\tau(s) = (K_P + K_I/s^{\lambda} + K_D s^{\mu})e(s) \tag{4.1}$$

where K_P, K_I, and K_D represent the proportional gain, integral and derivative gain constants, respectively. e is the error between desired and actual response of the system. The output of the controller τ is applied to the system under control as actuator signal. The terms λ and μ are the fractional orders of integral and derivative terms, respectively. The designing of an optimal FOPID controller is determined through finding the optimal values of the vector $K = (K_P, K_I, K_D, \lambda, \mu)$ for minimizing the performance index. In this study, the minimization is performed by the proposed GWO-ABC algorithm and the results are compared with five other

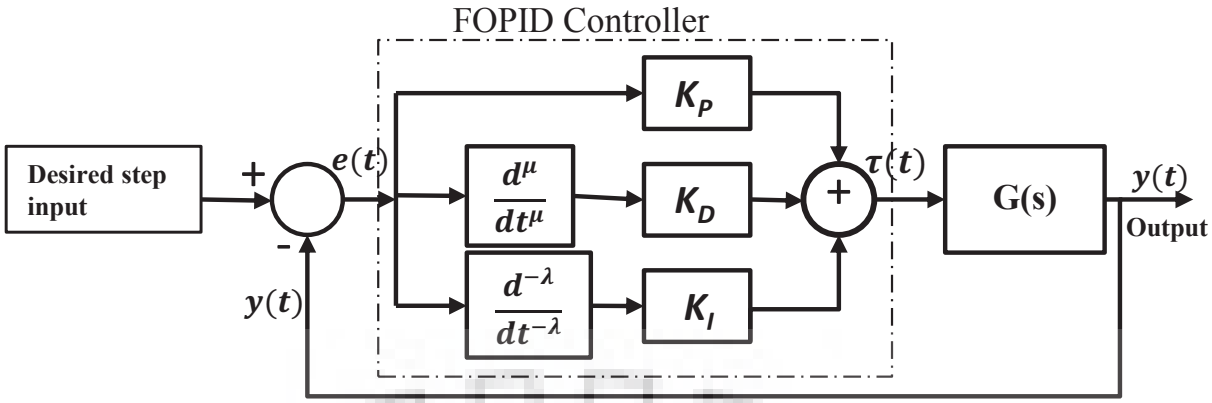


Fig. 4.1: Design scheme of FOPID controller applied to system G(s).

standard optimization algorithms GWO, PSO, ABC, GSA, and ALO.

4.2.2 Problem Definition

In this section, the performance of GWO-ABC is evaluated by determining the optimal parameters of FOPID controller for performance index $ITAE$. As reported in Section 2.3, there are other performance indices like IAE, ISE, $ITSE$ are also used by different studies. $ITAE$ is preferred in this study for its effective performance characteristics. The time multiplication term in error index $ITAE$ minimizes the chances of oscillations at later stages, therefore, the settling time (t_s) of the closed loop system is effectively reduced and the absolute value of error minimizes the percent overshoot ($M_p\%$). In case of adding square terms in indices like $ITSE$, oscillations damns faster but sudden change in set point may cause larger controller output. Thus, the rationale behind selecting $ITAE$, as a objective function in this study, is that it reduces percent overshoot ($M_p\%$) and rise time (t_r) and ensures faster settling time (t_s) with null steady-state error E_{ss}. It has also been mentioned in various studies that minimum value of $ITAE$ objective function provides better system response. Mathematically, the objective function based on $ITAE$ as given in (2.17) is

$$J_o = ITAE = \int t|e(t)|dt$$

where t is time and $e(t)$ is an error between actual and desired response.

4.2.3 Performance Evaluation on Test-bench Process Plants

It has been observed that, in conventional control applications, processes with higher-order models are approximated using first or second order systems with time delay [26]. Therefore, in order to verify the efficacy of the proposed algorithm for tuning of controllers comparative experiments are carried out over variety of complex test-bench process plants which are widely referred in the literature [3, 7, 26].

(a) First-order system with time delay [3] : The system in following generalized form is

considered.

$$G_a(s) = \frac{K}{Ts+1}e^{-Ls} \tag{4.2}$$

where K is process gain, T is time constant, and L is dead time. For the simulation in this work, these values are taken as; $K = 1$, $T = 1$, and $L = 0.5$.

(b) Non-linear system with time delay [7] :

$$G_b(s) = \frac{d^2y}{dt^2} + \frac{dy}{dt} + 0.25y^2 = u(t-L) \tag{4.3}$$

here time delay L is assigned as 0.5.

(c) Second-order stable linear system with time delay [26] : This second order system is considered with system-gain $K = 1$, the natural frequency $\omega_n = 0.45$, damping factor $\zeta = 1.125$ and L is the time delay, fixed as $L = 0.2$.

$$G_c(s) = \frac{K}{s^2 + 2\zeta\omega_n s + \omega_n^2}e^{-Ls} \tag{4.4}$$

(d) Non-minimum phase system [7] :

$$G_d(s) = \frac{1-0.5s}{(s+1)^3} \tag{4.5}$$

As shown in Fig. 4.1, the FOPID controller is applied to each of the above systems and controller parameters are tuned for step response.

4.2.4 Simulation Results and Discussion

The performance of the proposed GWO-ABC is compared with the conventional GWO, PSO, ABC, GSA, and ALO algorithms. All the parameters required to implement these algorithms are given in Table 3.1. For all the experiments, the population size is set as 30 and maximum number of function evaluations (for stopping criteria) is fixed to 3000. In order to attain the optimal parameters for FOPID controller, best of the results obtained from 20 runs are presented here. The suitable ranges assumed for different control parameters are as follows: $K_p \in [0,10]$, $K_I \in [0,10]$, $K_D \in [0,10]$, $\lambda \in [0,2]$, and $\mu \in [0,2]$. The step responses of all above systems with optimally tuned FOPID controllers are shown in Figs. 4.2, 4.3, 4.4, and 4.5. The comparative results of optimal controller parameters and resulted time-domain specifications for all the above systems $G_a(s)$, $G_b(s)$, $G_c(s)$, and $G_d(s)$ are summarized in Tables 4.1, 4.2, 4.3, and 4.4, respectively. From figures and tables, it can be observed that proposed GWO-ABC tuned FOPID controllers outperform the controllers tuned by GWO, PSO, ABC, GSA, and ALO algorithms in terms of minimum percent overshoot ($M_p\%$), lower rise time (t_r), and achieve null steady-state error (E_{ss}) in faster settling time (t_s), in most of the cases. On the other hand, although in case of $G_a(s)$

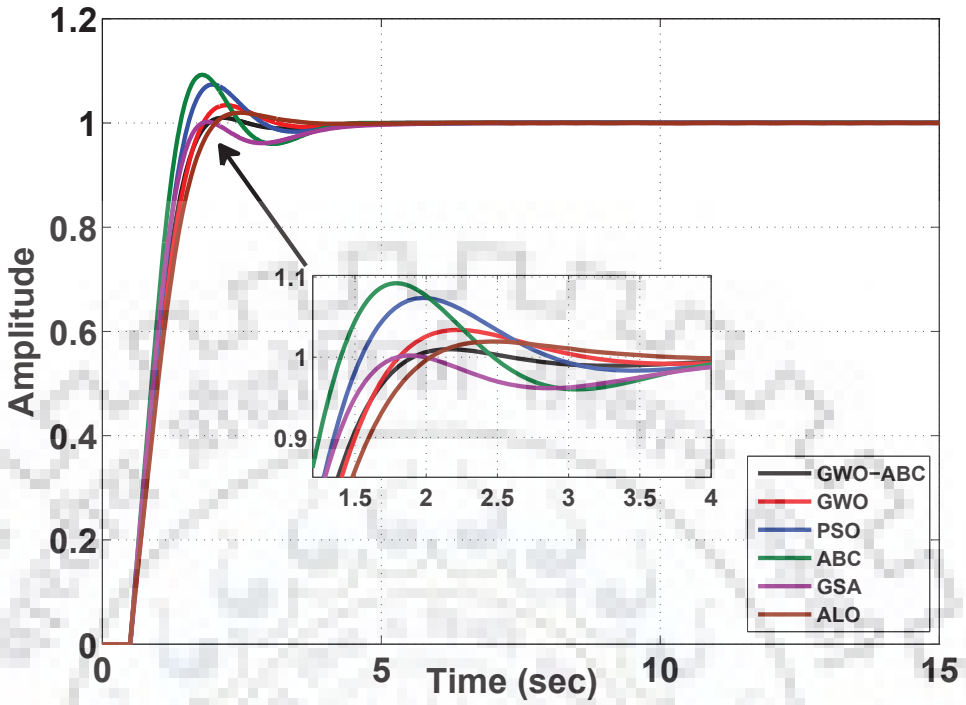


Fig. 4.2: Step response of first-order system with time delay.

Table 4.1: Optimal controller parameters and comparative results for the first-order system with time delay.

	Optimal Parameters					Time-domain Specifications				
Algorithm↓	K_P	K_D	K_I	μ	λ	$M_p(\%)$	$t_r(s)$	$t_s(s)$	E_{ss}	J_0
GWO	1.0011	0.0064	1.0012	0.0011	0.999	3.4823	0.9055	2.6321	0.0001	0.6843
PSO	1.1870	0.0012	1.0990	0.0208	1.0020	7.6044	0.7708	2.6301	0.0001	0.8930
ABC	1.3473	0.0001	1.1176	0.0031	1.0078	9.2012	**0.6730**	3.7010	0.0002	1.0207
GSA	2.1456	0.0450	1.9695	0.0841	0.8896	0.9020	0.7012	5.1241	0.0002	1.0127
ALO	3.0012	0.0664	2.1450	0.0641	0.2456	2.5691	1.0152	6.3210	0.0003	1.1021
GWO-ABC	1.0170	0.0902	1.0011	0.2977	0.9983	**0.9012**	0.8957	**1.7540**	**0.0000**	**0.4566**

The "best results" are indicated by bold values.

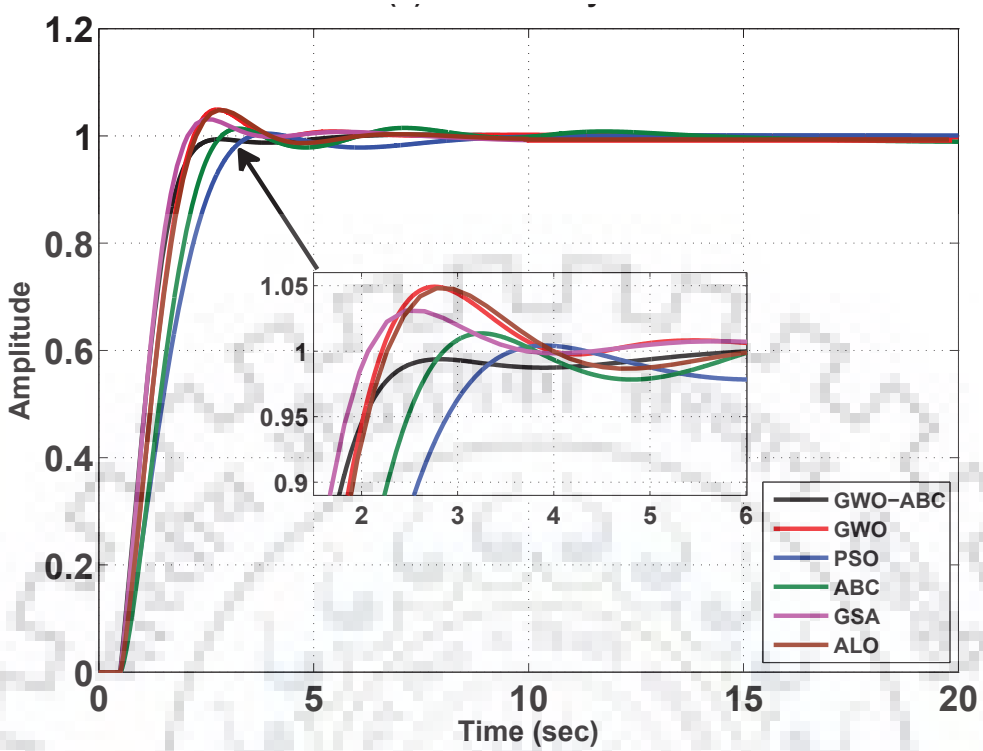


Fig. 4.3: Step response of non-linear system with time delay.

Table 4.2: Optimal controller parameters and comparative results for the non-liner system with time delay.

	Optimal Parameters					Time-domain Specifications				
Algorithm ↓	K_P	K_D	K_I	μ	λ	$M_p(\%)$	$t_r(s)$	$t_s(s)$	E_{ss}	J_0
GWO	0.4225	0.9701	0.2669	0.7270	0.9653	4.9212	1.1978	3.4711	0.0001	2.6998
PSO	0.8467	1.0011	0.0313	0.9488	1.0000	0.5031	1.8536	6.7212	0.0001	2.4523
ABC	0.4755	0.8874	0.2918	0.7790	0.8703	1.4011	1.5131	5.0911	0.0004	1.2146
GSA	0.6689	1.256	1.0254	0.5681	0.6541	2.5342	1.0815	5.3412	0.0003	2.8345
ALO	1.0123	2.1451	0.8654	0.6741	0.7546	4.8322	1.1893	6.0420	0.0004	2.2151
GWO-ABC	0.8140	0.9423	0.3224	0.9639	0.9965	**0.1001**	**1.0814**	**2.2991**	**0.0000**	**1.0280**

The "best results" are indicated by bold values.

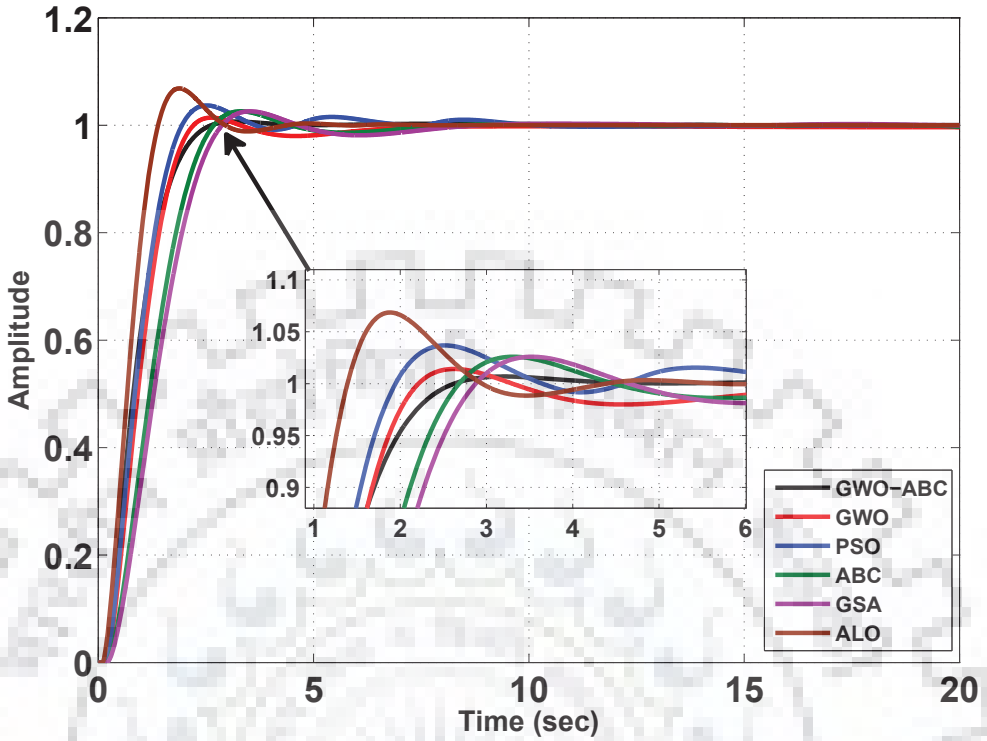


Fig. 4.4: Step response of second-order stable linear system with time delay.

Table 4.3: Optimal controller parameters and comparative results for the second-order stable linear system with time delay.

	Optimal Parameters					Time-domain Specifications				
Algorithm ↓	K_P	K_D	K_I	μ	λ	$M_p(\%)$	$t_r(s)$	$t_s(s)$	E_{ss}	J_0
GWO	0.8608	0.8798	0.2168	0.7187	0.9301	1.4023	1.2921	4.657	0.0004	2.4599
PSO	0.9722	0.9628	0.2809	0.7610	1.000	3.7051	1.1709	3.1210	0.0001	1.1348
ABC	0.1413	0.9730	0.2533	0.3988	0.8798	2.6180	1.6029	3.7361	0.0003	2.7291
GSA	3.4233	0.8879	1.0024	0.4501	0.8643	2.6210	18712	4.2311	0.0003	2.8134
ALO	1.0273	0.8475	2.0156	0.0315	0.5618	8.5301	**0.7802**	3.9050	0.0004	3.1024
GWO-ABC	0.9879	0.9762	0.3163	0.8759	0.9888	**0.7002**	1.3019	**2.2491**	**0.0001**	**1.0776**

The "best results" are indicated by bold values.

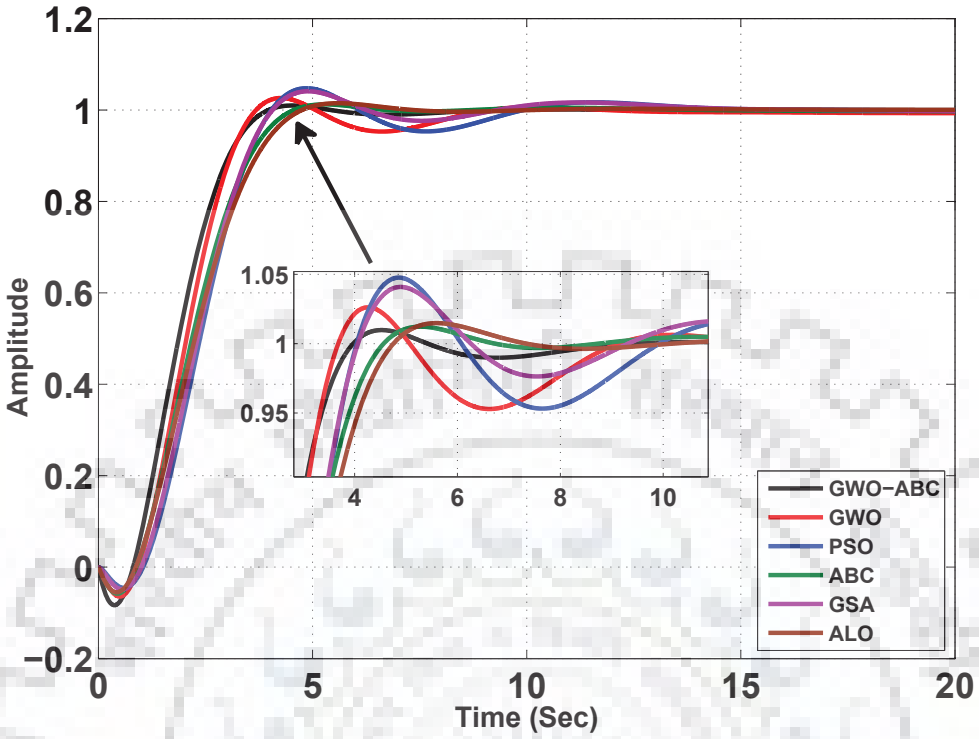


Fig. 4.5: Step response of non-minimum phase system.

Table 4.4: Optimal controller parameters and comparative results for the non-minimum phase system.

	Optimal Parameters					Time-domain Specifications				
Algorithm ↓	K_P	K_D	K_I	μ	λ	$M_p(\%)$	$t_r(s)$	$t_s(s)$	E_{ss}	J_0
GWO	0.9926	0.7540	0.4591	0.8537	0.9896	2.6042	2.0320	8.1221	0.0005	2.834
PSO	0.6580	0.7624	0.4620	0.4978	0.9842	4.8011	2.1461	9.1450	0.0002	4.8718
ABC	0.9797	0.6950	0.4467	0.8560	0.9952	1.3034	2.3440	4.2041	0.0001	3.0214
GSA	0.8964	0.4567	0.5896	0.7731	0.9812	4.1243	2.0123	4.3541	0.0003	4.0311
ALO	07698	0.8891	0.1176	0.6921	0.8978	1.5961	3.7602	13.230	0.0004	3.5019
GWO-ABC	0.9998	0.9988	0.5439	0.7910	0.9617	**0.9023**	**1.9919**	**3.645**	**0.0000**	**2.1724**

The "best results" are indicated by bold values.

and $G_c(s)$, the rise time of GWO-ABC tuned controller is marginally higher than that of some other algorithms, all other specifications like $M_p\%$, t_s, and E_{ss} by GWO-ABC tuned controller are superior in case of all. The last column of each table presents the values of performance index $J_o = ITAE$, which is also depicted in Fig. 4.6 for better comparative analysis. From all illustrations and tables, it is manifested that the proposed GWO-ABC algorithm can be a better option for multimodal controller tuning problems of complex systems.

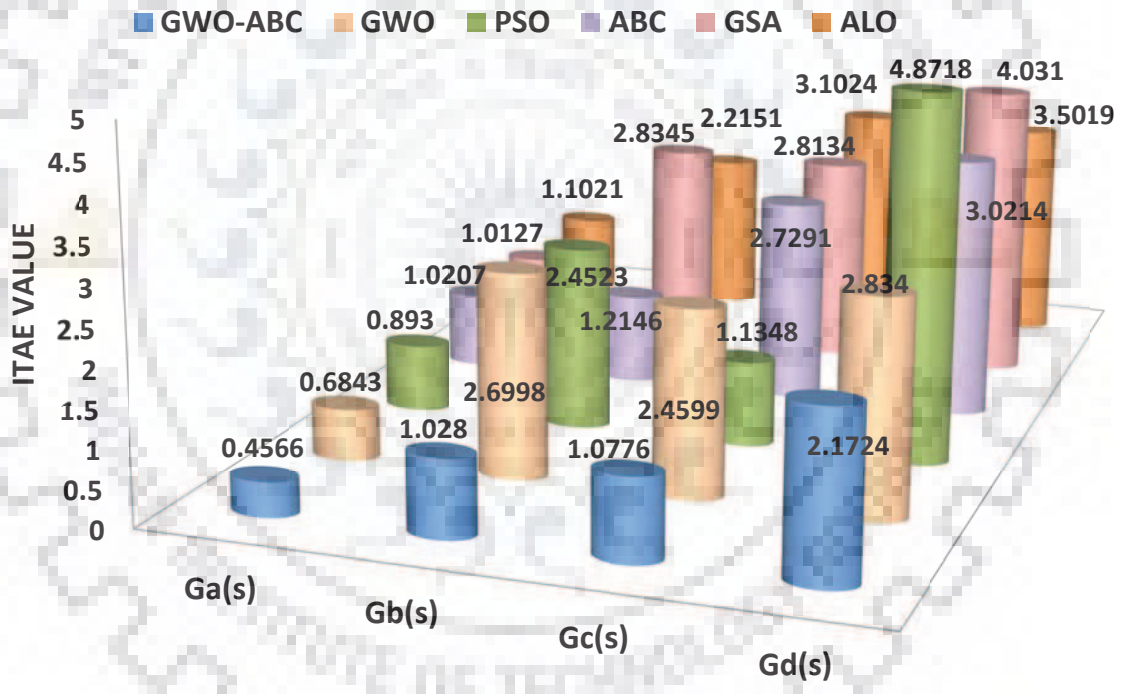


Fig. 4.6: Comparative illustrations of variation in *ITAE* for optimized FOPID controllers applied to test-bench process plants.

4.3 Proposed CFGWO Algorithm

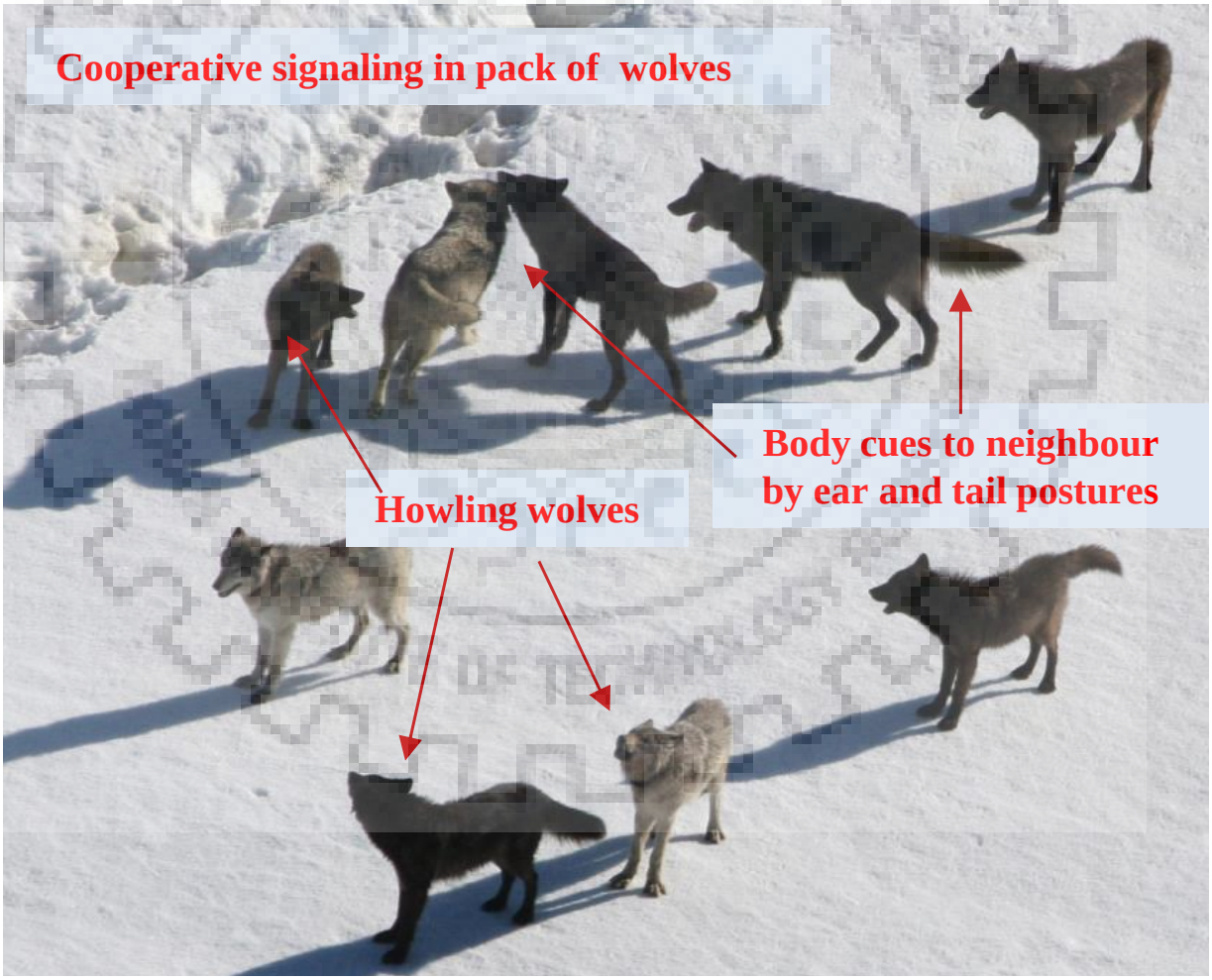
In this section, new cooperative foraging based GWO (CFGWO) is proposed by interpreting the motivations from the cooperative forging behaviour of animals (including grey wolves). The mathematical amendments are proposed in original GWO to incorporate cooperative foraging behaviour. Further, new acceleration coefficient is introduced and systematic structure of proposed CFGWO is presented.

4.3.1 Animal's Behaviour in Cooperative Foraging

Colin J. Torney, *et al.*, in [127] reported that the animals in cooperative foraging are able to acquire more knowledge about their environments than if they were to forage alone. This consequently enhances their ability to read environmental cues, and hence ameliorate their search efficiency. Authors also notified that sharing of information among conspecifics reduces the risk associated with unsuccessful foraging attempts (stagnation at local sub-optima) when conditions are unpredictable (such as stochastic MA). Similarly, in review paper [128], Ida Bailey *et al.*, provide evidences of different levels of hunt organisation and cooperation in carnivorans and stated that cooperative foraging associated with more advanced communication skills may improve hunt coordination than would otherwise be possible in some contexts. In [128], authors also specified in context of cooperative hunting that cooperation can simply means two or more individuals increasing their fitness (success as predator and, therefore, their chance of survival) by acting together (communicate). Thus, effective and honest communication in these situations would clearly enhance foraging efficiency of pack since it provides an additional level of reliable information to individuals [127]. Many studies on animal behaviour, including [129] by J. Ruch *et al.*, suggested the division of labour among the pack as main feature of cooperative foraging. It involves participation of each team member to adopt specialized roles to perform a subtask.

According to [128, 130], with in family of wolves communication helps in maintaining social stability. Specially, wolves communicate through (a) range of vocalization or (b) body language. Fig. 4.7 (b) illustrates these communication signalling among the members of the pack of wolves. These communications are used for gathering, hunting, and mourning for a lost pack mate or announcing territorial or mating intentions. It is observed that, during foraging and hunting, high level of communication strategy and information sharing is required in the pack until good prey is obtained. During chase, wolves may howl to motivate others to join a hunt and perform body movements like raising ears, and tail tips to give visual cues and signals to neighbours. But, once the global optimum region (around biggest prey) is obtained, whole pack is busy in exploiting it [127].

(a)



(b)

Fig. 4.7: Cooperative foraging and hunting behaviour of grey wolves by (a) Leadership based hunting approach, and (b) Cooperative signalling behaviour in the pack through (i) Vocalisation and (ii) Body language (Credit: National Park Service) [128].

4.3.2 Motivation from Cooperative Foraging

As discussed earlier, conventional GWO algorithm is the only population based algorithm which is based on leadership hierarchy where remaining low rank members of pack upgrade their positions based on the moves of three leading wolves, as depicted in Fig. 4.7 (a). In their original research paper [89], S.Mirjalili *et al.*, defined these remaining low ranking wolves as scapegoat and consider them as not so important individuals in the pack. Therefore, it would be plausible to claim that this algorithm follows dominance structure and poorly benefited by cooperative foraging strategy. Also, as discussed above, the adaptive values of parameters of $\overrightarrow{a}$ and $\overrightarrow{A}$ decide the transition by favouring exploration at the first half of the iterations and exploitation at the later half of the iterations. Mostly, these features make GWO incompetent in handling multimodal and complex problems in efficient manner.

Ultimately, these insights persuade us to model communication signalling of wolves to transfer information amongst individuals in the pack to aid all-inclusive cooperative foraging. The limitations also encourage us to introduce new acceleration coefficient to modify exploration ability of algorithm throughout iterations.

4.3.3 New Acceleration Coefficient

New acceleration coefficient ' $\varepsilon \in [1, 0.1]$' is introduced in the algorithm to control the rate of communication for information sharing between the search agents. The coefficient is decreasing with respect to increasing iterations as defined by following equation.

$$\varepsilon(t) = 1 - t \times \frac{0.9}{max\_iter} \tag{4.6}$$

here 't' represents current iteration value and 'max_iter' is the maximum number of iterations. The variation of ε for $max\_iter = 1000$ is illustrated in Fig. 4.8. It can be observed from figure that the algorithm starts with higher value of ε i.e., 1, however, with lapse of iterations it decreases linearly.

It is intended to initialize the algorithm with better communication signalling and reduce it in later iterations. This behaviour is supported by the fact that in later iterations search agents find better prey and converge to optimal region around it. Therefore, most of the search agents have similar information of prey and there is no point of sharing it among each-other.

4.3.4 Structure of the Proposed CFGWO Algorithm

In this section, the details of newly proposed CFGWO algorithm is presented. The step-wise flowchart of CFGWO algorithm is demonstrated in Fig. 4.9. The idea of cooperative foraging using learning, communication signalling, and cognitive processing are framed and implemented in this proposed algorithm. As delineated in flowchart, the steps in the CFGWO algorithm are

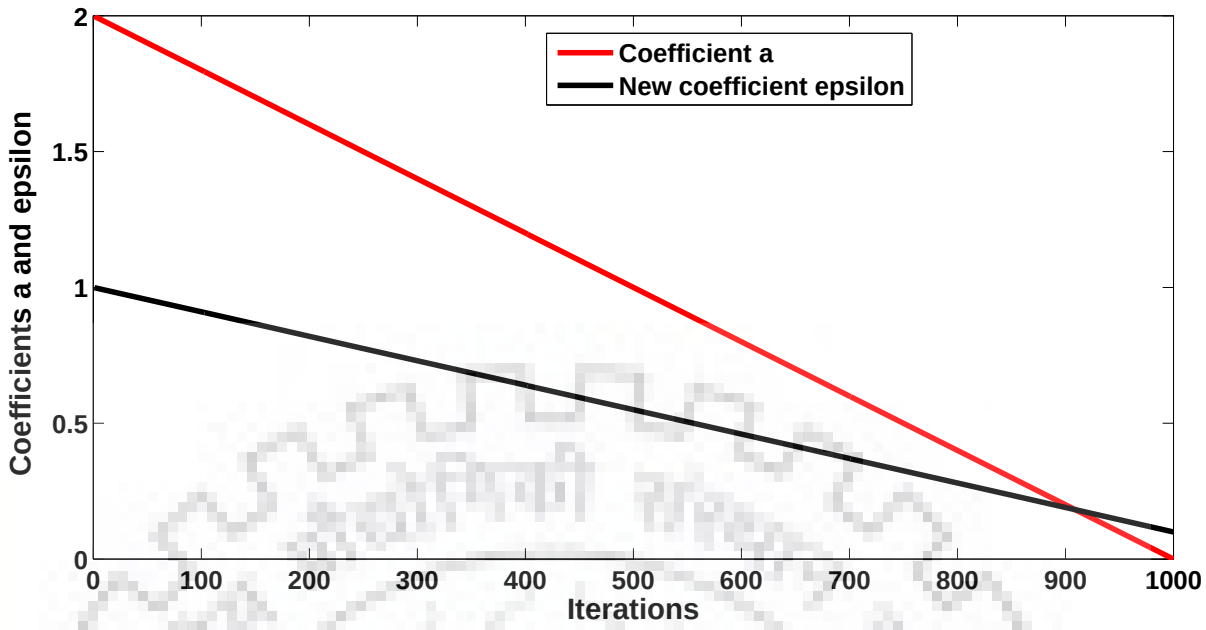


Fig. 4.8: Behaviour of new acceleration coefficient and parameter a against iterations.

similar to conventional GWO algorithm with upgraded initialization scheme and additional communication signalling strategy. For better reasoning, following description of CFGWO is divided in three main phases (i) initialization, (ii) learning by leadership hierarchy based GWO, and (iii) communication signalling to exchange the information among each-other.

Initially, swarm size N, search space with n dimension, maximum iterations max_iter are fixed and other parameters a,A, and C are evaluated.

Chaotic and OBL based population initialization: In [11], authors proposed an elitism based swarm initialization scheme by incorporating chaotic mapping and OBL strategy. In this work, we adopted the same scheme to initiate with widespread range of solutions. The step-wise detail of the scheme is given in Algorithm 2. Initially, set of solution candidates $X \in N$ is obtained through chaotic mapping, the equation of logistic chaotic map is given in Eq. (2.15), later, OBL methodology is used to generate X^* as given in step 4 in Algorithm 2. Both the sets are merged as $X = (X_i \cup X_i^*) \in 2N$ and the fitness is evaluated. Finally, all candidate solutions are sorted according to their fitness values and first N solutions are selected as initial population for further generations. The prime goal of this scheme is to get benefits of both chaotic mapping and OBL methodology, to get widespread solutions to initiate with fitter population.

Learning by leadership hierarchy: After generating initial population, algorithm proceeds according to the conventional GWO. In this phase, all individual search agents learn about the position of prey from three leading wolves and upgrade themselves. It follows the Eqs. (3.1) to (3.5), to update its parameters and current positions of search agents based on leadership hierarchy. Consequently, every search agent undergoes through learning about the prey position by following three leading wolves.

Communication signalling: The most striking aspect of the proposed CFGWO is coopera-

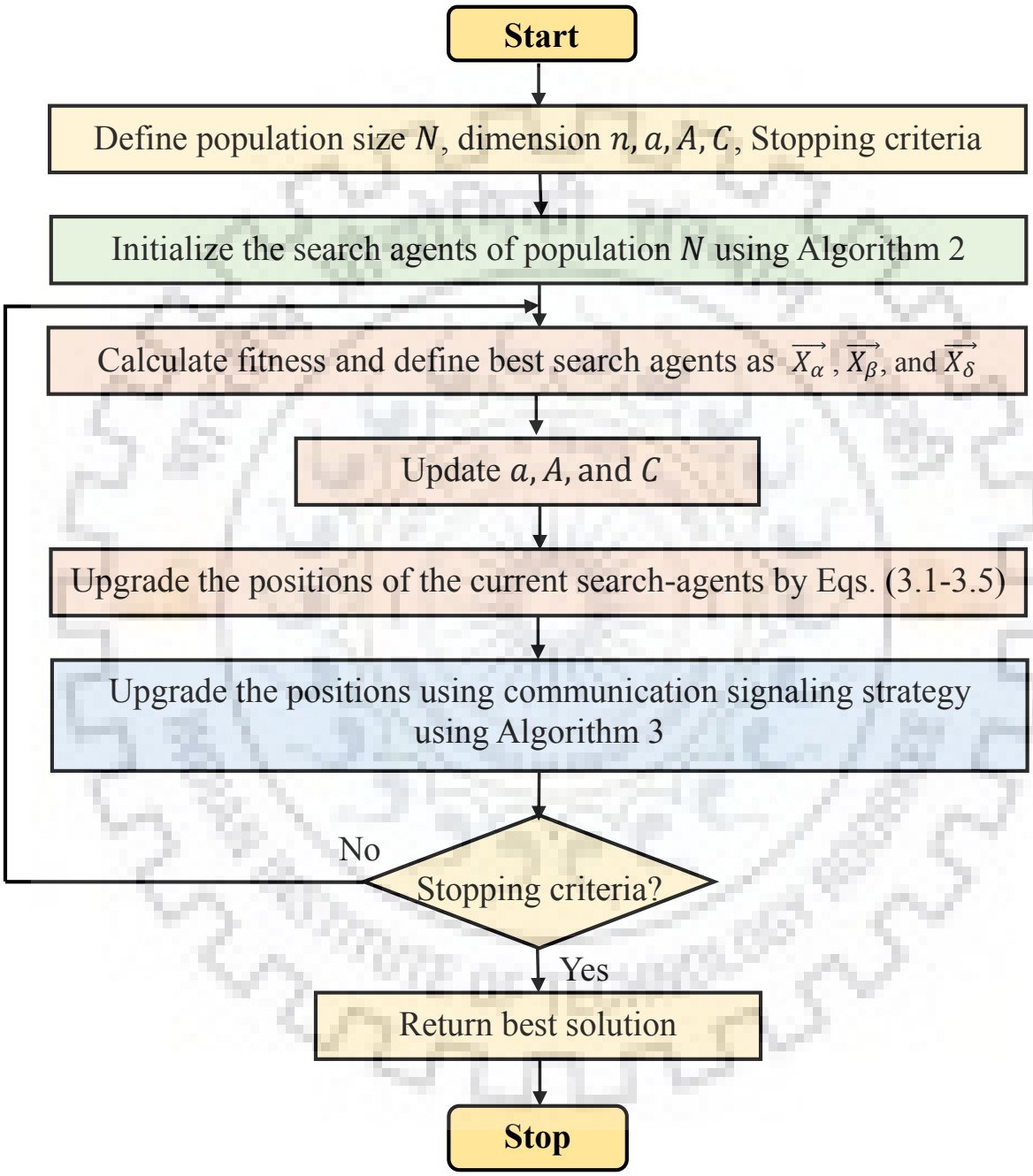


Fig. 4.9: Flowchart of the proposed CFGWO algorithm.

Algorithm 3 Procedure for communication signalling (information transfer) strategy among search agents.

Input : Updated search agents from GWO $X \in N$, dimension n, bounds $[Ub_1,\ldots,Ub_n,Lb_1,\ldots,Lb_n]$, current iteration t

Output : Advance search agents after information transfer $X = \hat{X}$

1: **for** each search agent $X_i = (x_{i1},x_{i2},\ldots,x_{il},\ldots,x_{im},\ldots x_{in})$, where $\forall i \in [N]$ **do**

2: Randomly select two numbers $l,m \in [n]$, where $l \neq m$

3: Randomly select neighbour of i, $k \in [N]$, where $k \neq i$

4: Obtain ch using Eq. (2.15) and $\varepsilon(t)$ using Eq. (4.6) (random number and new acceleration coefficient)

5: (i) $\hat{x}_{il} = ch \times x_{il} + \varepsilon(t) \times (x_{il} - x_{kl}) \times (ch - 0.5) \times 2$ (Information transfer with neighbour)

6: (ii) $\hat{x}_{im} = ch \times x_{im} + \varepsilon(t) \times (x_{im} - (Ub_m + Lb_m - x_{im}) \times (ch - 0.5) \times 2$ (Information transfer using OBL)

7: Update $\hat{X}_i = (x_{i1},x_{i2},\ldots,\hat{x}_{il},\ldots,\hat{x}_{im},\ldots x_{in})$

8: **end for**

tive foraging by communication through signalling and information transfer among the search agents. The detailed procedure for cooperative signalling is presented in Algorithm 3. Initially, two random separate positions $l,m \in [n]$ and neighbouring search agent $k \in [N]$ are selected for information exchange. As formulated in Eq. (i) in step 5, the variable at l is evaluated with respect to another randomly selected neighbouring search agents variable at k. Then, the OBL based move is incorporated in Eq. (ii) in step 6, to upgrade the variable at position m for exploring other side of search space. This formulation will enhance the exploration ability which is estimated by coefficient ε in proportionate manner to balance with exploitation.

These two communication schemes are amended to model the distinct signalling behaviours of wolves [130,131]. Correspondingly, (a) visual body signals to neighbour wolves is represented by Eq. (i) in step 5, and (b) vocalisation signalling, like howls of wolves, for communication with other members present elsewhere is emulated by Eq. (ii) in step 6 of Algorithm 3. The gradually decreasing acceleration coefficient ε is presented to vary this communication signalling from higher to lower value. The importance of ε is elaborated in detail in previous Subsection 4.3.3. Moreover, logistic chaotic mapping is employed to elevate the non-repetition, arbitrary selection, and randomness. The entire procedure of CFGWO is repeated until stopping criteria (max_iter) is met. Finally the best solution of last iteration $\overrightarrow{X_\alpha}$ is returned as optimal solution. Thus, cooperative signalling ameliorates global search ability of all individuals in the pack by bestowing them opportunity to share information with other members.

4.3.5 Performance Evaluation on Test Functions

The performance of the proposed CFGWO algorithm has been investigated on some of the test functions from appendix Tables A.3, A.4 and A.2. As discussed in previous chapter, different performance evaluations are carried out on the basis of Rank based analysis, statistical analysis, convergence curves, exploration and exploitation analysis. The results are compared with GWO, PSO, ABC, GSA, and recently proposed GWO-ABC [11] and LGWO [105] algorithms. The parameter settings for this analysis is similar to Table 3.1 and the grade notations defined in Section 3.4.3 for Statistical Analysis (Wilcoxon Rank Sum Test) are used in this investigation. In this study, 2 unimodal (f_1 and f_2), 2 multimodal (f_8 and f_9), and 2 fixed-dimension multimodal benchmark functions (f_{14} and f_{15}), and 2 composite functions from CEC2014 test bed (f_{24} and f_{25}) are investigated. The statistical analysis is given in Table 4.6. The convergence curves of the some of the functions are demonstrated in Fig. 4.10 which show the fast convergence trends by CFGWO than other algorithms. From the the overall ranking and results in Table 4.5, we can conclude that the cooperative foraging behaviour in Grey wolf improves the performance of the CFGWO than other algorithms. According to the last row of Table 4.5, CFGWO is statistically superior to other algorithms in 36 cases (75%) and similar in 9 cases (19%).

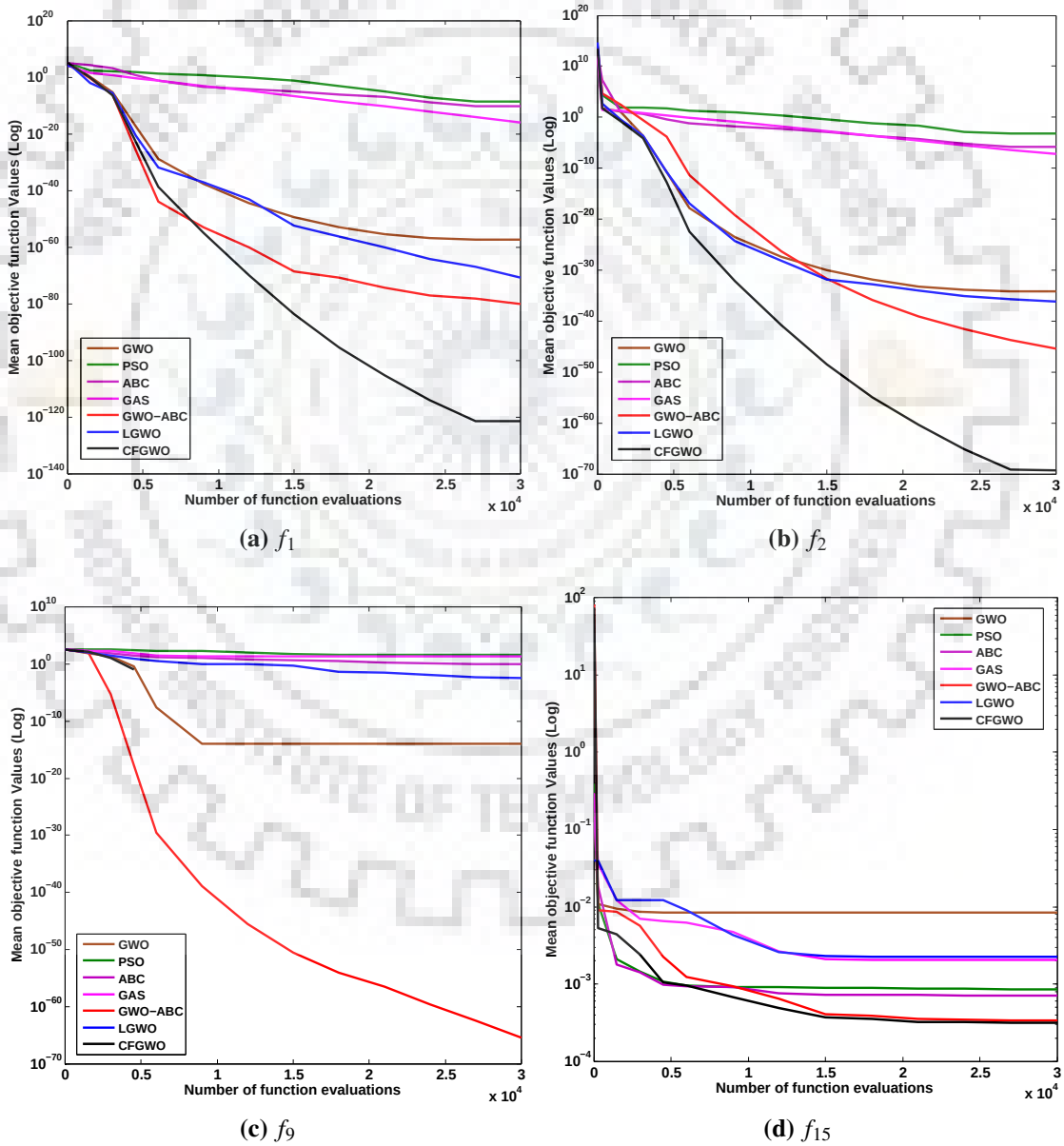
Table 4.5: Comparison of results obtained for some benchmark functions.

F ↓	Algorithm ↦	GWO	PSO	ABC	GSA	GWO-ABC	LGWO	CFGWO
f_1	Mean	$5.14E-58$	$2.47E-09$	$5.93E-11$	$1.13E-16$	$2.63E-82$	$2.11E-71$	**$2.34E-122$**
	SD	$5.60E-58$	$5.08E-09$	$5.40E-11$	$5.02E-17$	$1.46E-76$	$2.11E-19$	**$3.39E-122$**
	Rank(Test)	$4(+)$	$7(-)$	$6(+)$	$5(+)$	$2(+)$	$3(+)$	**1**
f_2	Mean	$6.82E-35$	$2.06E-04$	$1.42E-06$	$5.49E-08$	$3.56E-45$	$6.73E-37$	**$5.41E-70$**
	SD	$6.94E-35$	$2.02E-04$	$5.01E-07$	$1.69E-08$	$3.06E-05$	$1.23E-03$	**$5.60E-70$**
	Rank (Test)	$4(+)$	$7,(+)$	$6(+)$	$5(+)$	$2(+)$	$3(+)$	**1**
f_8	Mean	-6488.69	-6363.61	-7105.93	-5123.23	-8155.67	-6658.25	**-8750.71**
	SD ‘	615.5570	1403.804	162.2757	125.6531	345.6641	212.3540	122.1860
	Rank (Test)	$5(+)$	$6(+)$	$3(-)$	$7(+)$	$2(+)$	$4(\approx)$	**1**
f_9	Mean	$1.13E-14$	43.77850	1.022845	25.60100	$1.13E-69$	0.004501	**0**
	SD	$2.39E-14$	9.520100	0.684078	7.112000	$5.54E-16$	19.56420	**0**
	Rank (Test)	$3(-)$	$7(+)$	$5(+)$	$6(+)$	$2(+)$	$4(+)$	**1**
f_{14}	Mean	4.911902	4.053910	0.998100	3.024600	0.990100	1.120342	**1.002100**
	SD	4.588900	2.641901	**2.43E-15**	1.819201	0.044640	0.001210	0.011110
	Rank (Test)	$6(+)$	$5(+)$	$2(-)$	$4(+)$	$2(+)$	$3(+)$	**1**
f_{15}	Mean	0.00833	0.000858	0.0007	0.002043	0.00032	$2.36E-06$	**0.000301**
	SD	0.01035	0.000157	0.000127	0.000443	0.000127	$6.06E-05$	**1.76E-05**
	Rank (Test)	$7(+)$	$5(+)$	$4(+)$	$6(+)$	$2(\approx)$	$3(-)$	**1**
f_{24}	Mean	2623.651	2616.012	2648.276	2601.034	2541.300	2539.465	**2513.956**
	SD	0.986542	0.011205	0.005124	0.775424	0.062214	0.754860	**0.001564**
	Rank (Test)	$6(+)$	$5,(+)$	$7(-)$	$4(+)$	$3,(-)$	$2(-)$	**1**
f_{25}	Mean	2661.842	2693.554	2703.221	2710.341	2641.300	2626.125	**2621.022**
	SD	0.004570	12.88545	0.061452	1.325410	0.070012	0.001332	**0.000995**
	Rank (Test)	$4(+)$	$5,(\approx)$	$6(-)$	$7(+)$	$3,(+)$	$2(+)$	**1**
	Average Rank	4.875	5.875	4.874	5.625	2.250	3.000	**1**
	Overall Rank	4	7	4	6	2	3	**1**
	+/-/ ≈	7/1/0	6/1/1	4/4/0	8/0/0	6/1/1	5/2/1	**36/9/4**

The “best results” are indicated by bold values.

Table 4.6: *p*–values and *grades* obtained by statistical Wilcoxon rank-sum test with 1% significance level.

$F\downarrow$	GWO	PSO	ABC	GSA	GWO-ABC	LGWO
f_1	$4.51E-03\,(+)$	$6.58E-02\,(-)$	$9.60E-03\,(+)$	$1.92E-03\,(+)$	$2.81E-07\,(+)$	$1.82E-06\,(+)$
f_2	$3.21E-04\,(+)$	$1.20E-05\,(+)$	$1.42E-05\,(+)$	$7.92E-03\,(+)$	$3.34E-04\,(+)$	$1.24E-04\,(+)$
f_8	$2.55E-08\,(+)$	$5.36E-04\,(+)$	$8.44E-01\,(-)$	$5.64E-03\,(+)$	$7.93E-04\,(+)$	$1.10E-02\,(\approx)$
f_9	$6.89E-01\,(-)$	$2.45E-03\,(+)$	$1.47E-10\,(+)$	$4.66E-06\,(+)$	$6.44E-05\,(+)$	$6.52E-03\,(+)$
f_{14}	$8.32E-06\,(+)$	$6.12E-03\,(+)$	$9.21E-02\,(-)$	$1.04E-03\,(+)$	$2.32E-04\,(+)$	$6.14E-04\,(+)$
f_{15}	$2.21E-03\,(+)$	$6.32E-04\,(+)$	$4.81E-07\,(+)$	$2.01E-03\,(+)$	$1.03E-02\,(\approx)$	$6.28E-02\,(-)$
f_{24}	$6.28E-05\,(+)$	$6.54E-04\,(+)$	$5.23E-02\,(-)$	$2.10E-03\,(+)$	$8.64E-01\,(-)$	$9.22E-01\,(-)$
f_{25}	$6.44E-04\,(+)$	$1.09E-02\,(\approx)$	$9.33E-02\,(-)$	$1.02E-03\,(+)$	$5.67E-04\,(+)$	$9.11E-05\,(+)$



(a) f_1 **(b)** f_2

(c) f_9 **(d)** f_{15}

Fig. 4.10: Comparison of convergence curves of proposed CFGWO and other algorithms obtained for some of the test functions.

4.4 Implementation of CFGWO for Controller Tuning Problem

As discussed in Chapter 2, robotic manipulators are multi input multi output (MIMO), coupled, and highly complex non-linear system, hence their controller designing is crucial and challenging task. This problem can be considered as a multimodal complex problem with many local optima having nearly equivalent values. In literature [20, 23, 24], several works have been proposed to manifest the efficiency of MA to solve these complex tuning problems for precise trajectory tracking by robotic arms. With the aim to prevail coherent approach, this section investigates the efficacy of CFGWO for tuning of controller parameters. This experimentation gives us opportunity to validate CFGWO as efficient optimization tool for intelligent controller design. The fractional order fuzzy logic PID (FO-FPID) controller is explained in the following section.

4.4.1 FO-FPID Controller

As discussed earlier, many studies [26] had revealed the limitations of PID to handle complex systems. This fact always encourages researchers to amend additional tuning parameters in controller design to get extra DOF. Many researchers improve the decision making by using fuzzy logic based techniques. Along with this, the fractional parameters are integrated with fuzzy-PID controller to inherit the advantages of both the methodologies to improve the dynamic response and to eliminate the steady-state error. In this section, the structure and the mathematical modelling of FO-FPID controller is discussed, followed by brief introduction of FO-approximation and FLC implementation. The basic mathematical modelling of FOPID controller, shown in Fig. 4.1 in Section 4.2, is further explained as

$$u(t) = K_P e(t) + K_I \frac{d^{-\lambda}}{dt^{-\lambda}} e(t) + K_D \frac{d^{\mu}}{dt^{\mu}} e(t), \quad 0 \leq \mu, \lambda \leq 2 \tag{4.7}$$

Here the K_P, K_I, K_D are proportional, integral and derivative gains, respectively. The error is represented as $e(t)$ and λ and μ are integral and derivative fractional orders.

The above FOPID controller is integrated with two-input type-1 FLS, as shown in Fig. 2.1 (a), to improve the decision making using fuzzy inference mechanism. The new structure of FO-FPID controller structure is demonstrated in Fig. 4.11. Basically, it is designed as parallel structures of FO-FPD and FO-FPI. The introduction of FLC in controller has significantly enhanced the performance and applicability of controllers, as they can be designed with proper rules and membership functions (MF) without knowledge of exact dynamic model of system. Hence, their application, with human expert knowledge, has increased over conventional controller to control complex and non-linear systems.

As shown in Fig. 4.11, the two inputs to the FLC are the scaled values of error $e(t)$ and the its fractional derivative $\frac{d^{\mu}e(t)}{dt^{\mu}}$ and output is $U(t)$. The G_E and G_{CE} are respective input scaling factors. Though, FLC is non-linear controller, for simplicity, its linear approximation is

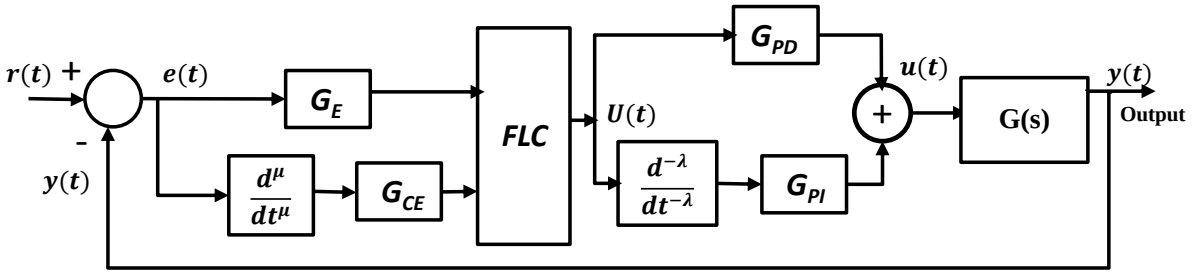


Fig. 4.11: Design scheme of FO-FPID controller applied to system $G(s)$.

considered as

$$U(t) = G_E e(t) + G_{CE} \frac{d^{\mu} e(t)}{dt^{\mu}} \tag{4.8}$$

ultimately, the overall control law $u(t)$ is formulated as

$$u(t) = G_{PD} U(t) + G_{PI} \frac{d^{-\lambda} U(t)}{dt^{-\lambda}} \tag{4.9}$$

later, putting the values of $U(t)$ from Eq. (4.8) and simplifying, we get

$$u(t) = [G_{PD} G_E] e(t) + [G_{PD} G_{CE}] \frac{d^{\mu} e(t)}{dt^{\mu}} + [G_{PI} G_E] \frac{d^{-\lambda} e(t)}{dt^{-\lambda}}$$

$$+ [G_{PI} G_{CE}] \frac{d^{\mu-\lambda} e(t)}{dt^{\mu-\lambda}} \tag{4.10}$$

Comparing above Eq. (4.10) with basic equation of FOPID controller defined in Eq. (4.7), the controller parameters in terms of scaling factors are equivalent in the following way.

$$\begin{aligned}
&\text{for } \mu < \lambda && K_p = G_{PD} G_E,\ K_D = G_{PD} G_{CE},\ K_I = G_{PI} G_E + G_{PI} G_{CE} \\
&\text{for } \mu = \lambda && K_p = G_{PD} G_E + G_{PI} G_{CE},\ K_D = G_{PD} G_{CE},\ K_I = G_{PI} G_E \\
&\text{for } \mu > \lambda && K_p = G_{PD} G_E,\ K_D = G_{PD} G_{CE} + G_{PI} G_{CE},\ K_I = G_{PI} G_E
\end{aligned}$$

It can be perceived that the scaling factors of FO-FPID controller are analogous to terms of the FOPID controller expressed in Eq. (4.7). The parameters G_E, G_{PI}, G_{PD}, and G_{CE} along with μ and λ are tuned for generating proper control law $u(t)$, to get desired output $y(t)$ as seen in Fig. 4.11.

Design of FLC

The fuzzy logic theory is assimilated in FO-FPID controller to adopt the designer's knowledge and provide flexibility via inference mechanism. The basic blocks in FLC are fuzzifier, inference engine based on rule base, and defuzifier as shown in Fig. 2.1 (a). The pre-processing and post-processing blocks are involved for scaling the input and output variables. The input variables are normalized to map the universe of discourse in $[-1, 1]$ range. Further, the output variable is

Table 4.7: Rule base for FO-FPID controller.

$\mathbf{E} \Downarrow / \frac{\mathbf{d}^{\mu}\mathbf{E}}{\mathbf{dt}^{\mu}} \Rightarrow$	**NB**	**NM**	**Z**	**PM**	**PB**
NB	NB	NB	NB	NM	Z
NM	NB	NB	NM	Z	PM
Z	NB	NM	Z	PM	PB
PM	NM	Z	PM	PB	PB
PB	Z	PM	PB	PB	PB

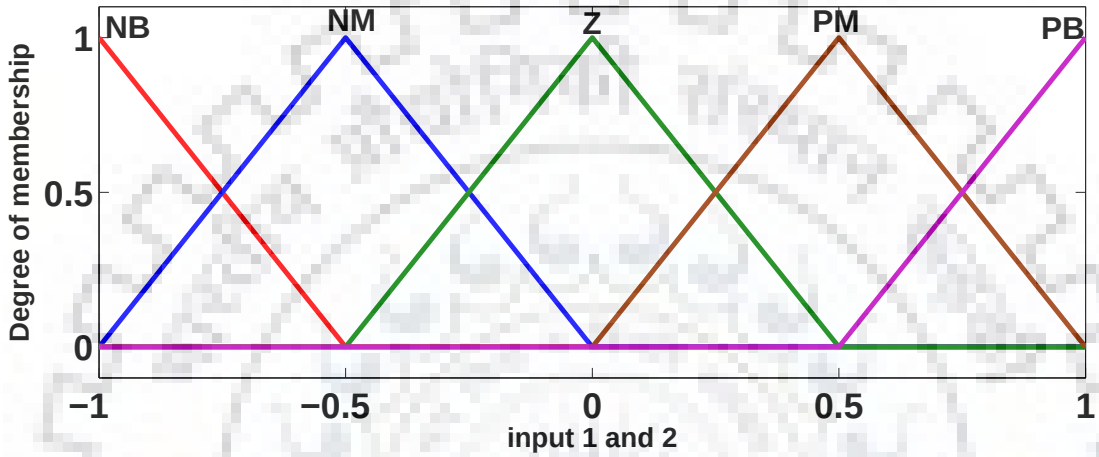


Fig. 4.12: Illustration of consequent MF used for FO-FPID controller.

multiplied with denormalization factor to convert it to actual range.

As shown in Fig. 4.11, their are two inputs: error $e_i(t)$ and its fractional derivative $\frac{d^{\mu_i} e_i(t)}{dt^{\mu_i}}$, where i represents the links of robotic arm. Each input is characterised by five triangular MFs represented as: 'Positive Big (PB)', 'Positive Medium (PM)', 'Zero (Z)', 'Negative Medium (PM)', and 'Negative Big (NB)'. Generally, triangular MFs are preferred for their ease of implementation in hardware as depicted in Fig. 4.12, [24]. The output MFs are also defined with similar number of crisp singletons as: NB = -1, NM = -0.8, Z = 0, PM = 0.8, and PB= 1, [24] as shown in Fig. 2.3. Rule base is the core part of FLC and defined on the basis of expertise knowledge, process dynamics, and nature of the response. The 5×5 rule base is designed and implemented in this work as represented in Table 4.7 and their non-linear surface plot is depicted in Fig. 4.13. For the presented work, Sugeno inference mechanism is used and all design and simulations are carried out in Fuzzy Logic Toolbox in MATLAB. The inference output is defuzzified to crisp value through the center of gravity procedure.

4.4.2 Problem Definition

In this work, the problem is framed to analyze the performance of CFGWO for real-world trajectory tracking problem. Fig. 4.14 represents the schematic diagram of FO-FPID controller for individual links of 2-link robotic manipulator. The main objective is to tune the controller param-

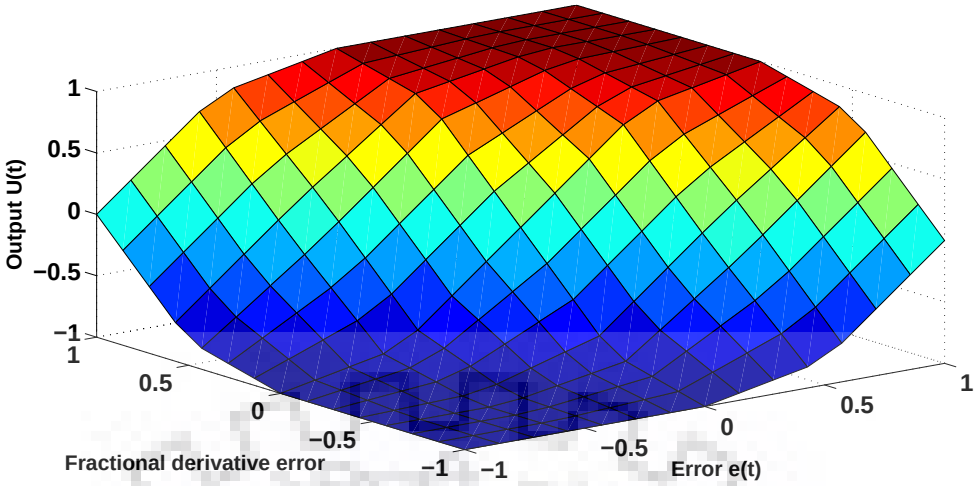


Fig. 4.13: Surface plot for rule base defined in Table 4.7.

eters to manipulate the torque applied to link joints, so that, the objective function is minimized. In order to apply optimization algorithms, the selection of proper objective function plays vital role to succeed. Mostly, in controller designing time-domain performance indices are manipulated to achieve desired performance. Hence, to determine the optimal parameters of FO-FPID controller, minimization of index *ITAE* is opted as the objective function. For simulation, the trajectory is tracked for 4 sec with the sampling time of 1 ms.

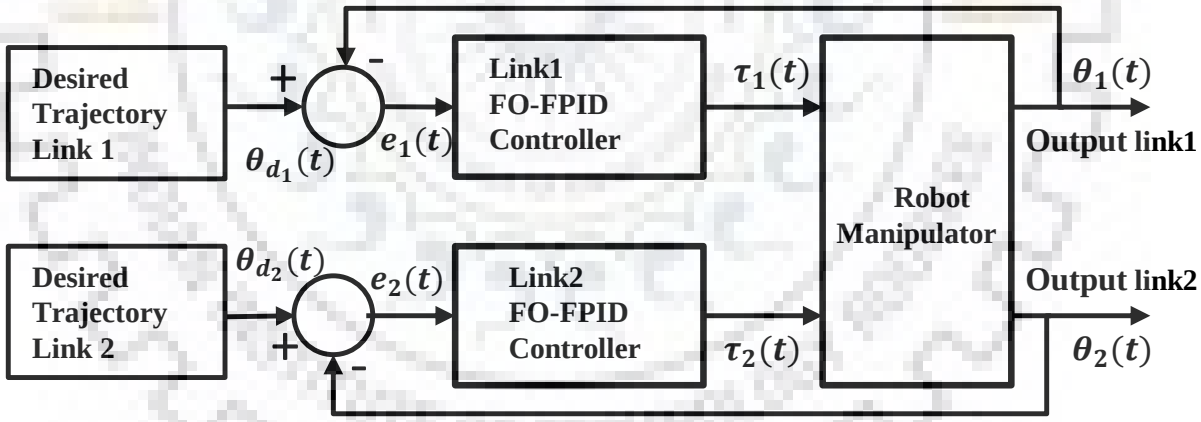


Fig. 4.14: Design scheme of FO-FPID controller applied to 2-link of robotic manipulator.

The main control objective is defined in terms of weighted sum of *ITAE* of individual links as defined in Eq. (4.13). The e_{l1} and e_{l2} in Eqs. (4.11) and (4.12) represent the errors between desired and actual trajectories of individual links.

$$J_{l1} = ITAE_1 = \int t|e_{l1}(t)|dt \tag{4.11}$$

$$J_{l2} = ITAE_2 = \int t|e_{l2}(t)|dt \tag{4.12}$$

$$J_0 = w_1 \times J_{l1} + w_2 \times J_{l2} \tag{4.13}$$

Table 4.8: Optimal controller parameters and comparative results of the FO-FPID controller.

Algorithm ⇒	GWO		LGWO		GWO-ABC		CFGWO	
Parameter ⇓	Link1	Link2	Link1	Link2	Link1	Link2	Link1	Link2
G_E	383.281	0.05712	211.812	374.355	186.000	458.803	494.242	498.391
G_{CE}	0.62053	77.0216	43.2122	29.7172	144.651	220.146	0.01250	0.36991
G_{PD}	300.000	265.380	151.968	110.403	39.7851	37.8610	499.303	470.916
G_{PI}	0.23442	82.6271	6.87710	90.2482	15.7800	499.031	120.472	88.1880
μ	1.24810	0.16530	0.42930	0.09620	0.19070	0.39890	0.35130	0.98980
λ	0.16821	1.28410	0.75621	0.79330	0.70671	1.39771	0.95521	1.43810
ITAE	$1.99E-03$	$2.19E-03$	$1.32E-03$	$9.71E-03$	$3.32E-03$	$5.90E-3$	**5.25E−05**	**9.79E−05**
SD	0.02123	0.03512	3.22145	0.06614	0.01056	$8.44E-03$	**1.28E−04**	**6.84E−07**

The "best results" are indicated by bold values.

where w_1 and w_2 are the weights assigned to J_{l1} and J_{l2}, respectively. In this work, both w_1 and w_2 are equal to 1.

4.4.3 Simulation Results and Discussion

As, problem definition demands to control trajectory of two-link manipulator with payload, two separate controllers are applied for each link as shown in Fig. 4.14. Accordingly, there are twelve tuning parameters, given as, G_{E1}, G_{CE1}, G_{PD1}, G_{PI1}, λ_1, and μ_1 for Link1 and G_{E2}, G_{CE2}, G_{PD2}, G_{PI2}, λ_2, and μ_2 for Link2. The experimental settings for the CFGWO and other optimizaers are as follows: The swarm population size is fixed to 30 for 100 iterations and results are noted from best of 20 independent runs. Multi-dimensional search space is defined to get optimal solutions with the wide range (L_b, U_b) of these parameters, specified as, $G_{Ei} \in [0,500]$, $G_{CEi} \in [0,500]$, $G_{PDi} \in [0,500]$, $G_{PIi} \in [0,500]$, $\lambda_i \in [0,2]$, and $\mu_i, \in [0,2]$, where $i = 1,2$. The performance of the proposed CFGWO performance is compared with conventional GWO, and its recent upgraded versions GWO-ABC and LGWO. Here, it is important to note that other algorithms namely:- PSO, ABC, and GSA are not considered for comparative analysis because of their lower overall ranks in GWO-ABC in Chapter 3. The parameter settings used in these evaluations are similar to values defined in Table 3.1.

The optimal parameters resulted after stopping criteria are reported in Table 4.8. The *ITAE* values by CFGWO for Link1 and Link2 are 5.25×10^{-05} and 9.79×10^{-05}, respectively. This signifies the exact tracking of desired trajectory by robotic arm. To substantiate the excellence, *ITAE* values for both the links resulted from rest of the algorithms are also represented in Fig. 4.15. It is interesting to observe that the *ITAE* values by CFGWO are much smaller than rest of the optimizers. Fig. 4.16 depicts the convergence behaviour of all four algorithms The plots clearly infer the ability of CFGWO to reach the optimum solution in faster time. Ultimately, the findings suggest that the FO-FPID controller parameter tuned by the proposed CFGWO algorithm are exceptionally optimized than other counterparts. The SD values obtained over results from 20 independent runs indicate the repeatability and robustness of CFGWO than GWO and others.

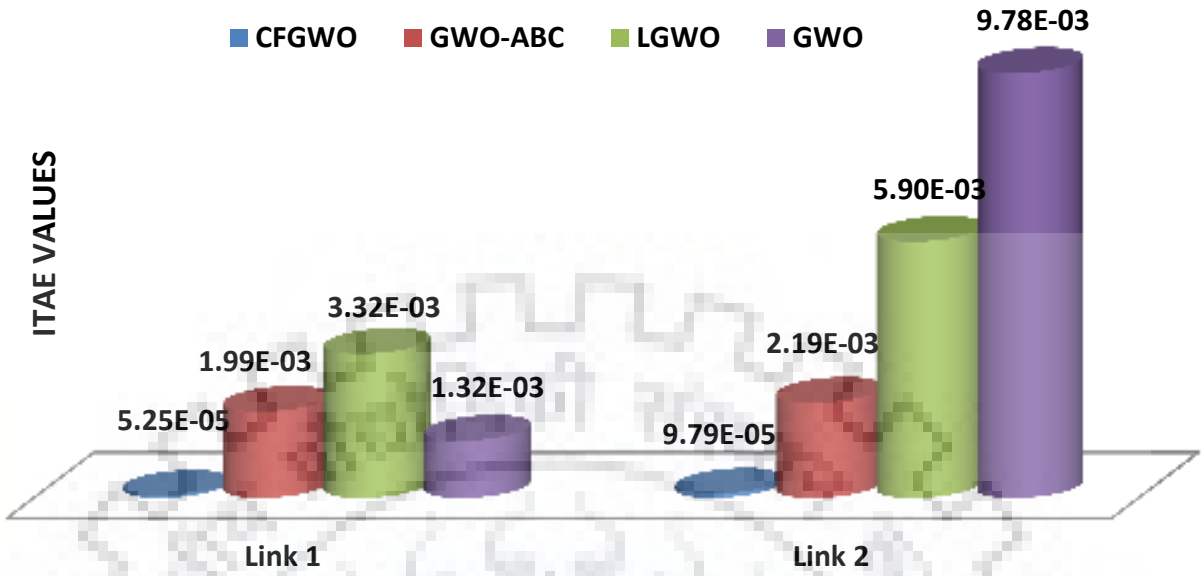


Fig. 4.15: Comparative illustrations of variation in *ITAE* for 2-link robotic manipulator applied with FOPID controllers optimized by different algorithms.

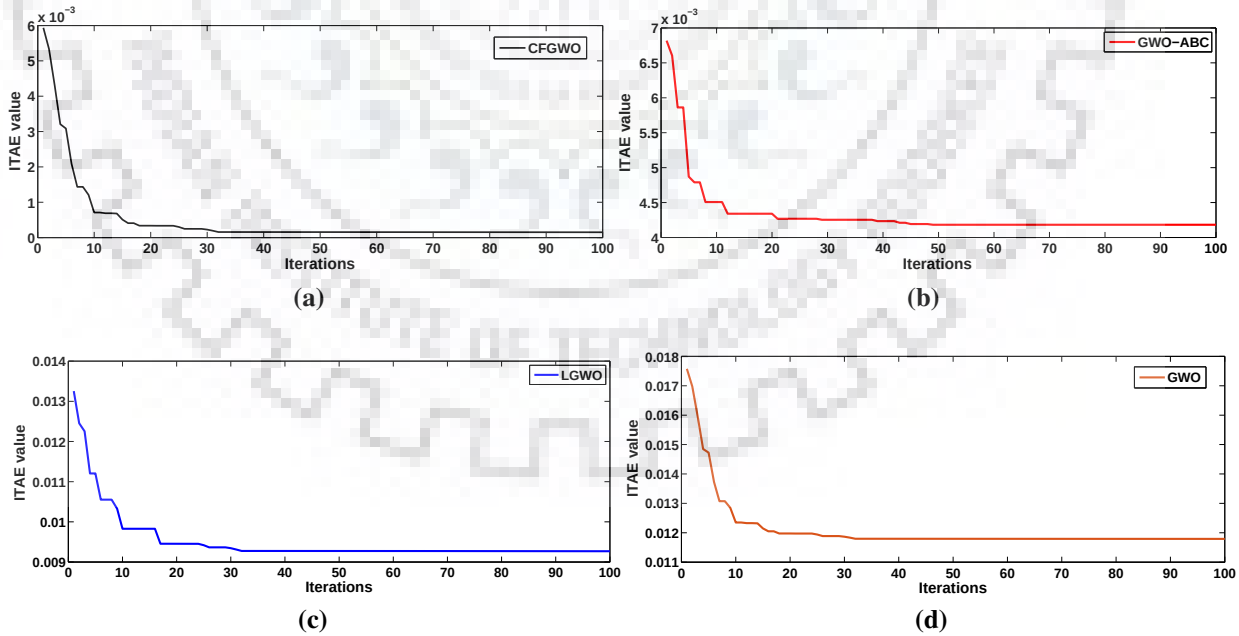
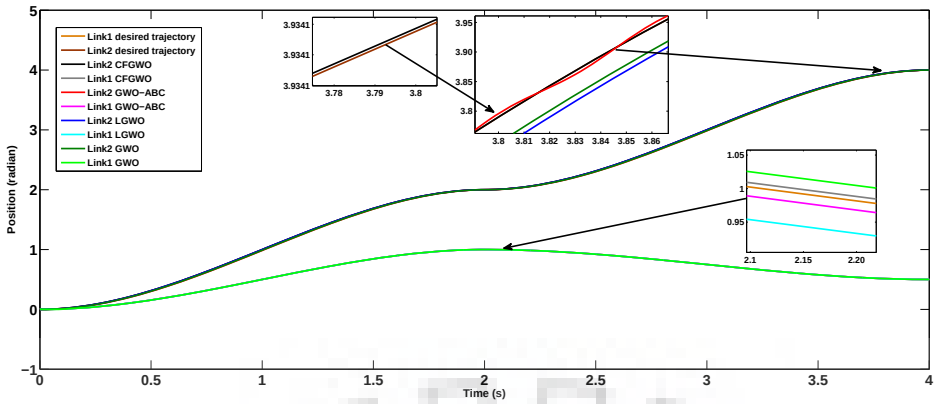
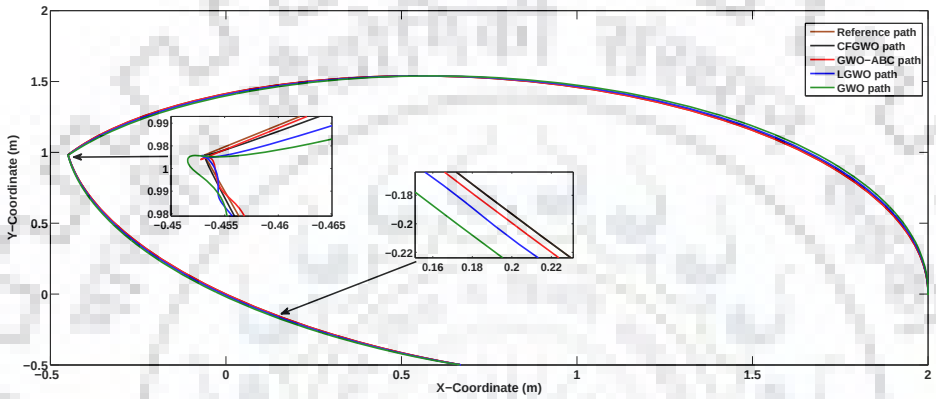


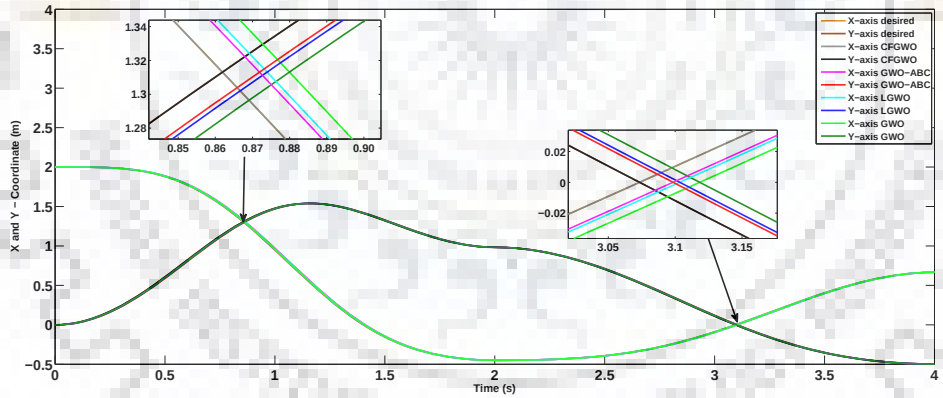
Fig. 4.16: Comparison of convergence curves of CFGWO and other algorithms obtained for FO-FPID controllers.



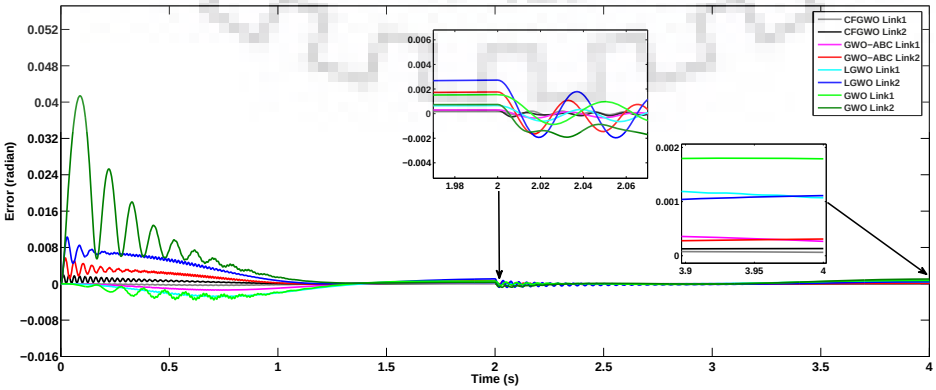
(a) Trajectory tracking performances



(b) Path tracked by end-effector



(c) X and Y versus time variation



(d) Position errors

Fig. 4.17: Various comparative illustrations demonstrating different performances of 2-link robotic manipulator with payload obtained by optimal FO-FPID controller.

To elucidate the performance of CFGWO, the graphs of trajectory tracking performance, path traced by end-effector, and X and Y coordinate versus time variations against their desired reference curves are depicted in Figs. 4.17 (a), (b), and (c), respectively. It can be clearly perceived from the plots (and enlarged sub-plots, therein) that the CFGWO obtained accurate results demonstrating precise overlapping on desired trajectories. Significantly, these plots provide indisputable evidences about superiority of CFGWO over other optimizers. Along with this, Fig. 4.17 (d) demonstrates the plots of position errors for both the links. Here, we can clearly depict that CFGWO optimizes the controller performance with low position errors and effective controller output. Ultimately, it can be inferred that the proposed modifications recommended in CFGWO ameliorate the conventional GWO to enhance the overall performance in controller design applications for complex systems. It gives better accuracy and precision in trajectory tracking problem.

4.5 Concluding Remarks

This chapter reported the experimental analysis to evince the performance of proposed algorithms for designing different controllers. Initially, FOPID controller is applied to four complex test bench process plants and have been optimized through GWO-ABC algorithm proposed in last chapter. Several time-domain specifications like low overshoot, better rise time, faster settling time, minimum steady-state error, and performance index are evaluated and tested against other equivalent counterparts. The results show that GWO-ABC outperform other algorithms and give optimal tuning parameters.

Eventually, the new optimizer CFGWO is presented by incorporating two communication signalling methods and new acceleration coefficient is introduced to manage the impact of signalling throughout iterations. CFGWO is also investigated for real-world optimization problem of tuning of FO-FPID controller parameters for robotic manipulator with payload at tip. Several trajectory tracking plots are demonstrated and compared among four best ranked algorithms. All the illustrations elucidate how the CFGWO surpass other algorithms while handling highly nonlinear, complex, and uncertain system. In this way, CFGWO incorporates fundamental in-depth behaviour of wolves as a methodological change and confirms its efficacy from an application point of view. Ultimately, CFGWO can also be established as a viable, simple, and fast alternative optimizer for complex control system design problems in real world applications via AI tools. Remarkably, CFGWO also processed in same computational time as it does not require extra fitness evaluations compared to conventional GWO.

Chapter 5

Design of Interval Type-2 Fuzzy Precompensated PID Controller

This chapter outlines the design procedure of novel interval type-2 fuzzy precompensated PID controller for trajectory tracking problem of 2-link robotic manipulator with the variable payload. After introduction in Section 5.1, the design procedure of IT2FP-PID controller and strategy to tune the various controller parameters are presented in Section 5.2. Later in Sections 5.3 and 5.4, the exhaustive experimental analysis and simulation results are reported. Finally, concluding remarks are drawn in last section.

5.1 Introduction

Various studies in the field of fuzzy control have manifested that the interval type-2 fuzzy logic controller (IT2-FLC), with footprint of uncertainty (FOU) in membership functions (MF), has increasingly recognized for controlling uncertainties and non-linearities. However, design of IT2-FLC based controllers and optimizing its overall parameters emerged as a time consuming, complex, high-dimensional, and constrained optimization problem. The prime contribution of the work in this chapter is to extend current knowledge of IT2-FLC to propose an advanced IT2FP-PID controller for a 2-link robotic manipulator for trajectory tracking problem and optimize it through tuning of several parameters including antecedent MFs. The overall control system design strategy addressed in this chapter is illustrated clearly in Fig. 1.1. It can be observed from the figure that the selection of suitable controller and appropriate tuning strategy plays an important role in control system design procedure. Within this framework, this chapter presents design of new IT2FP-PID controller and provides the systematic tuning strategy. In short, the major contributions of the work in this chapter are summarized in subsequent points.

The work outlined in this chapter has been disseminated in the following publication:

- P. J. Gaidhane, M. J. Nigam, A. Kumar, and P. M. Pradhan, "Design of interval type-2 fuzzy precompensated PID controller applied to two-DOF robotic manipulator with variable payload", *ISA Transactions*, Dec. 2018. (In press)

- Design of novel IT2FP-PID controller is presented for trajectory tracking problem of 2-link robotic manipulator with variable payload.

- The systematic strategy to tune the various controller parameters, scaling factors, and antecedent MF parameters of IT2FP-PID controller is presented to get optimized results and to make the most of FOU.

- The proposed GWO-ABC algorithm is effectively used to solve this high-dimensional constrained optimization problem.

- The comprehensive trajectory tracking analysis in terms of performance index *ITAE* of the optimized IT2FP-PID controller is carried out and compared to optimized T1FP-PID, FPID, and classical PID controllers.

- The efficacy of the proposed controller is also validated through exhaustive robustness analysis in presence of distinct non-linear dynamics such as (i) payload variations, (ii) model uncertainties, (iii) disturbance in signals, and (iv) random noise at feedback path.

5.2 Design and Optimization of IT2FP-PID Controller

In this section, the design of the proposed novel IT2FP-PID controller is presented. Later, elucidation of their MF structures is provided and the strategy of MF tuning used in this work is explained. Finally, implementation of optimization algorithm is presented.

5.2.1 Proposed IT2FP-PID Controller

The structural design of the proposed IT2FP-PID controller is illustrated in Fig. 5.1. Basically, it consists of two modules connected in cascade as seen in figure.

The initial module comprises of interval type-2 FLC based precompensated controller and the sole purpose of it is to change the control signal to compensate overshoots and undershoots in the transient response. It reduces the steady-state error and meliorates the output performance to counteract on disturbed parametric and external disturbances. It is mostly pertained when the

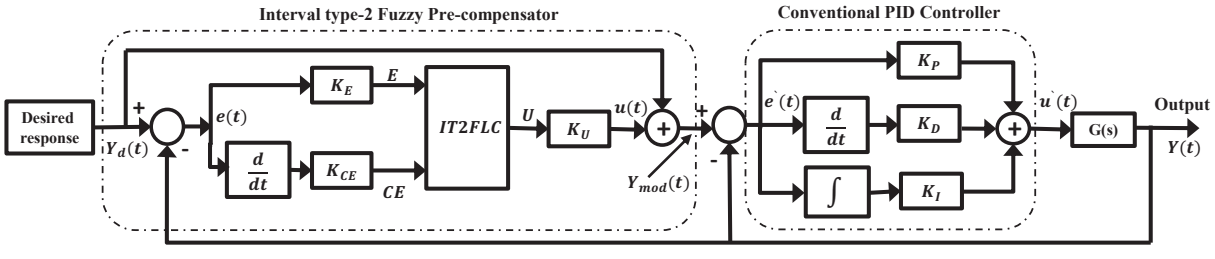


Fig. 5.1: Design scheme of the proposed IT2FP-PID controller applied to system $G(s)$.

system possesses unknown non-linearities which can result in significant overshoots and undershoots. FLC is incorporated for flexibility in decision making provided by defining the rules on the basis of problem [56].

As demonstrated in Fig. 5.1, the IT2-FLC is provided with two inputs normalized error $e(t)$ and rate of change of error $\Delta e(t)$. Both the input signals are normalized to E and CE using the scaling factors K_E and K_{CE}, respectively. This normalization is formulated by the following equations.

$$E(t) = K_E e(t) = K_E(Y_d(t) - Y(t)) \tag{5.1}$$

$$CE(t) = K_{CE}\frac{de(t)}{dt} = K_{CE}\Delta e(t) \tag{5.2}$$

where $e(t)$ is the difference between desired $Y_d(t)$ and the actual trajectory $Y(t)$. The IT2-FLC generates output U, which is also normalized using scaling factor K_U to obtain the compensation or correction term $u(t)$, given as

$$u(t) = K_U\, IT2FLC\,(E(t), CE(t)) \tag{5.3}$$

here the $IT2FLC\,(E, CE)$ is considered as the non-linear function of E and CE based on inference of IT2-FLC.

For the initial module, the compensated desired output is modified as

$$Y_{mod}(t) = u(t) + Y_d(t) \tag{5.4}$$

The modified error $e'(t)$ is calculated as the conflict between modified output response $Y_{mod}(t)$ and present response $Y(t)$.

$$e'(t) = Y_{mod}(t) - Y_(t) \tag{5.5}$$

$e'(t)$ is applied as the input to next module i.e. PID controller. Eventually, the regulated control law or torque $\tau_j(t)$ that actuates the link position of manipulator, is defined as

$$u'(t) = K_P e'(t) + K_D\frac{de'(t)}{dt} + K_I\int e'(t)dt \tag{5.6}$$

5.2.2 Proposed Optimization Strategy

The optimization strategy of IT2FP-PID controller is presented in this section. Both type-1 and type-2 FLCs are considered in this study and their parameters are optimized for comparative analysis.

As shown in Fig. 5.1, there are two inputs, error $e(t)$ and its derivative $\Delta e(t)$, applied to FLC. Each input is characterized by three triangular MFs denoted by linguistic labels as: 'Pos-

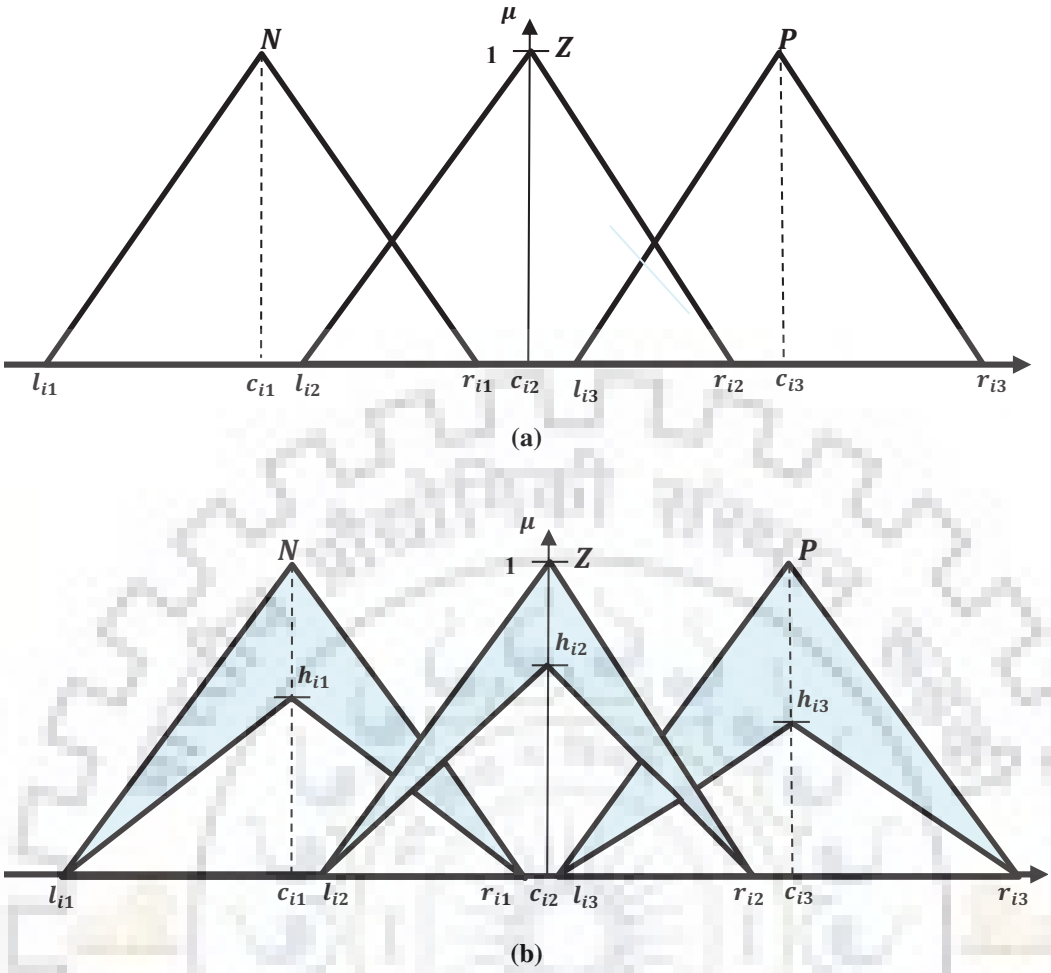


Fig. 5.2: General structures of antecedent MFs for the inputs of (a) T1-FLC and (b) IT2-FLC applied to both the links.

itive (P)', 'Zero (Z)', 'Negative (N)'. Generally, triangular MFs are preferred for their ease of implementation in hardware as depicted in Fig. 2.1 (c) and Fig. 2.2 (c) for type-1 and type-2 FLCs, respectively. On the other hand, the output consequent MFs are represented in five crisp singletons as: 'Positive Big (PB $= 1$)', 'Positive Medium (PM $= 0.8$)', 'Zero(Z $= 0$)', 'Negative Medium (NM $= -0.8$)', and 'Negative Big (NB $= -1$)', as depicted in Fig. 2.3.

Rule base defines the prime functioning of FLC inference mechanism. The rules are defined in accordance with the expertise knowledge, process dynamics, and nature of the response. The 3×3 rule base is designed and implemented in this work, for type-1 and type-2 FLCs, as represented in Table 5.3. Their non-linear surface plot is depicted in Fig. 5.7. Sugeno inference mechanism is used in this work and all design and simulations are carried out in Fuzzy Logic Toolbox in MATLAB.

The antecedent T1-FS is labelled by three parameters $(l_{im}, c_{im}, r_{im}; i = 1, 2, \ldots$ and $m = 1, 2, 3 \ldots)$. The present work considers two inputs $i = 2$ for each link and three triangular MFs $m = 3$ are used to define each input, as shown in Fig. 5.2 (a).

Similarly, the antecedent of IT2-FS is defined with four parameters $(l_{im}, c_{im}, r_{im}, h_{im};\ i =$

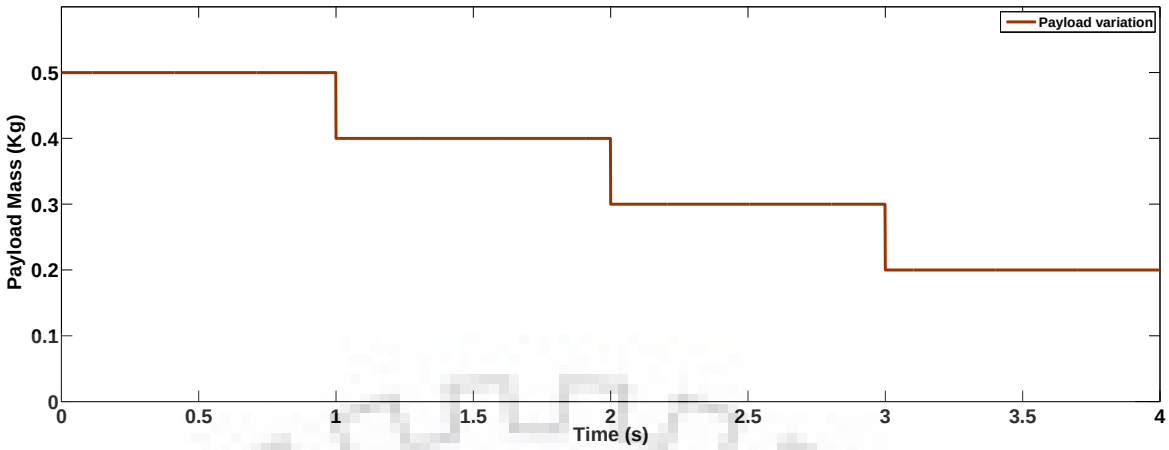


Fig. 5.3: Illustration of payload variations at the tip.

$1,2,\ldots$ and $m = 1,2,3\ldots$). Similar to T1-FS, here also, two inputs are considered $i = 2$ and each input is defined by three MFs $m = 3$, as shown in Fig. 5.2 (b). As discussed above, the outputs MFs variables for all FLCs used in this work are five crisp singleton consequent as shown in Fig. 2.3, to do fair comparison.

In this design strategy, following constraints have to be fulfilled by T1-FS and IT2-FS while optimizing the shape of the antecedent MFs.

$$\begin{aligned} c_{i1} &< c_{i2} < c_{i3} \\ l_{i1} &< c_{i1} < r_{i1} \\ l_{i2} &< c_{i2} < r_{i2} \\ l_{i3} &< c_{i3} < r_{i3} \end{aligned} \tag{5.7}$$

The additional constrain for IT2-FS is to maintain the height of LMF, such that

$$0 < h_{im} < 1 \tag{5.8}$$

All above parameters can be identified from Figs.5.2 (a) and (b).

5.2.3 Implementation of IT2FP-PID Controller on Robotic Manipulator

It is clear from the above discussion and modelling in Appendix A that the robotic arm is highly non-linear and coupled MIMO machine. Here, the additional complexity is introduced in the system model by applying variable payload at the tip. The pattern of variations in payload at end-effector is shown in Fig. 5.3. The required controller for 2-link robotic manipulator have 2 link positions (θ_1, θ_2) as inputs and the 2 torques (τ_1, τ_2) as outputs. In proposed work, control of independent joint design is considered separately as SISO models. Consequently, two individual controllers are independently employed to control both the links as shown in Figs.5.5 and 5.6.

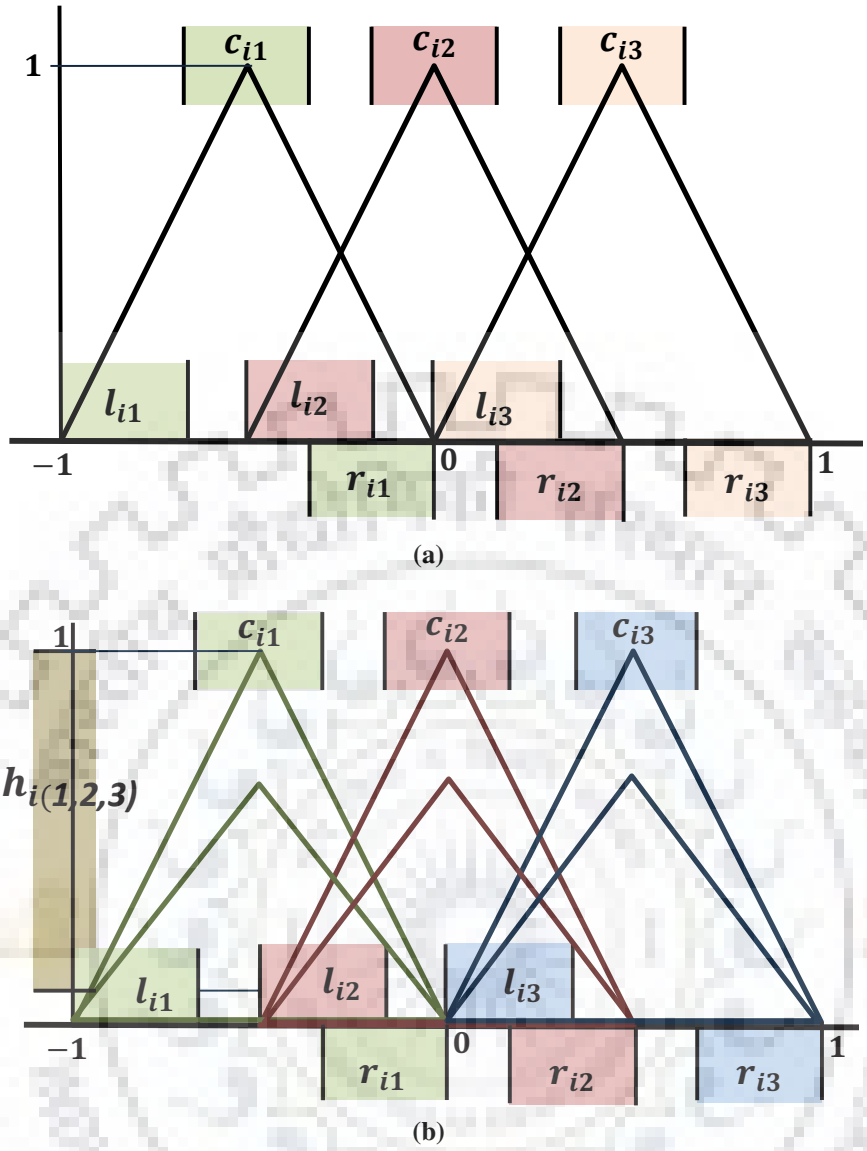


Fig. 5.4: Constrained handling scheme proposed for (a) T1-FLC and (b) IT2-FLC.

As we compared the performance of IT2FP-PID with PID and FPID controllers, brief discussion of their parameters is presented here. The PID controller implemented in this study is similar to the next module of IT2FP-PID controller as shown in Fig. 5.1. It does not have any fuzzy module and each link have only 3 parameters K_P, K_I, K_D as given in Table 5.2. Thus, 6 parameters are optimized for both the links.

The overall structure of FPID controller used for comparison is demonstrated in Fig. 5.5. It compromized of 4 parameters for each link given as $K_{E_1}, K_{CE_1}, K_{P_1}, K_{I_1}$ for Link1 and $K_{E_2}, K_{CE_2}, K_{P_2}$, K_{I_2} for Link2. Intentionally, general T1-MFs as shown in Fig. 2.1 (c) and rule base as defined in Table 5.3 are used and all 8 parameters are optimized.

As shown in figure, the various scaling parameters and MF parameters are required to optimize for desired response. Distinctly, different fuzzy sets, as depicted in Fig. 5.2 (a) and (b) are considered in the case of TIFP-PID and IT2FP-PID. Consequently, different numbers of tuning

Table 5.1: Initial upper and lower limits of antecedent MF parameters used in IT2-FLC.

Controllers ⇒	left		centre		right		height	
Parameter ⇓	LB	UB	LB	UB	LB	UB	LB	UB
$MF1U$	-1.000	-0.667	-0.666	-0.333	-0.332	0.000	1.000	1.000
$MF1L$	-1.000	-0.667	-0.666	-0.333	-0.332	0.000	0.150	0.980
$MF2U$	-0.500	-0.167	-0.1666	0.1666	0.5000	1.000	1.000	1.000
$MF2L$	-0.500	-0.167	-0.1666	0.1666	0.5000	1.000	0.150	0.980
$MF3U$	0.0000	0.3333	0.3334	0.6666	0.6667	1.000	1.000	1.000
$MF3L$	0.0000	0.3333	0.3334	0.6666	0.6667	1.000	0.150	0.980

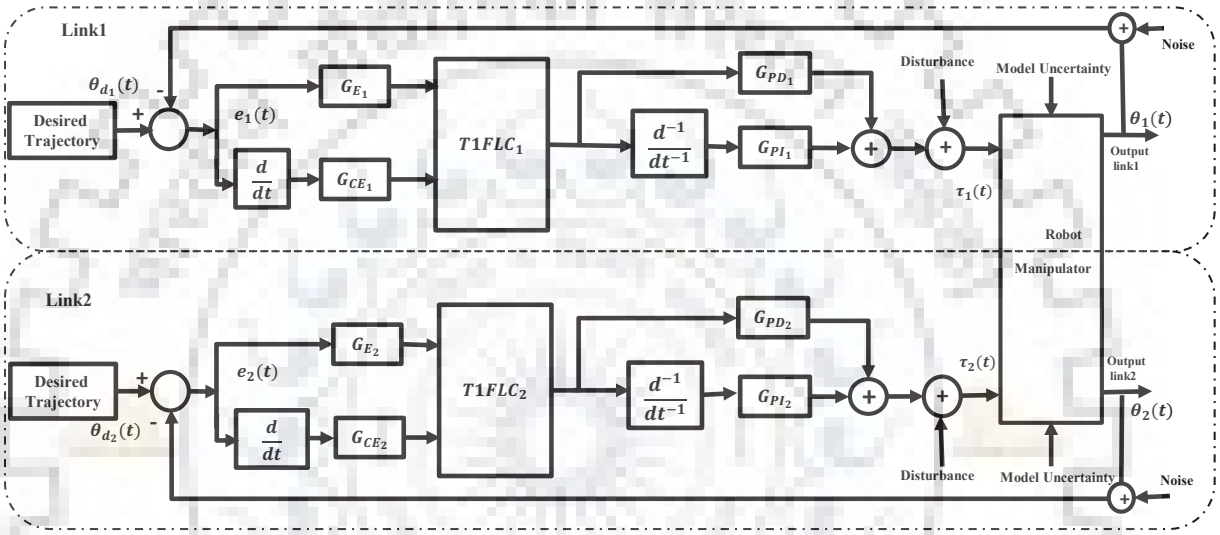


Fig. 5.5: Design scheme of FPID controller applied to 2-link robotic manipulator.

variables are required to optimize for both the controllers as discussed below.

The T1FP-PID controller is applied with two inputs (E, CE) defined by 3 MFs, 'N', 'Z', and 'P', each. MFs for first input are designed with parameters (l_{11}, c_{11}, r_{11}), (l_{12}, c_{12}, r_{12}), and (l_{13}, c_{13}, r_{13}), as shown in Fig. 5.4 (a). Similarly, second input MFs are designed with parameters (l_{21}, c_{21}, r_{21}), (l_{22}, c_{22}, r_{22}) and (l_{23}, c_{23}, r_{23}). As the controller have 2 inputs with 3 MFs each, and each MF have 3 parameters. There are total 18 MF parameters for each link. Additional 6 scaling factors ($K_E, K_{CE}, K_U, K_P, K_I$, and K_D) make the total 24 tuning variables per link. The same number of variables for second link makes the overall count of parameters to 48. These details are listed in Table 5.2.

The IT2-FLC in the proposed IT2FP-PID controller have two IT2-FS inputs (E, CE) with 3 MFs each ('N', 'Z', and 'P'). MFs for first input E are designed with $(l_{11}, c_{11}, r_{11}, h_{11})$, $(l_{12}, c_{12}, r_{12}, h_{12})$ and $(l_{13}, c_{13}, r_{13}, h_{13})$, as shown in Fig. 5.2 (b). Similarly, MFs for the second input CE are designed with $(l_{21}, c_{21}, r_{21}, h_{21})$, $(l_{22}, c_{22}, r_{22}, h_{22})$ and $(l_{23}, c_{23}, r_{23}, h_{23})$. Thus, 2 inputs $\times$ 3 MFs $\times$ 4 variables per MF $= 24$ parameters are requred to define the MFs to each link. Additional 6 scaling factors ($K_E, K_{CE}, K_U, K_P, K_I$, and K_D) make the total to 30 tuning parameters for each

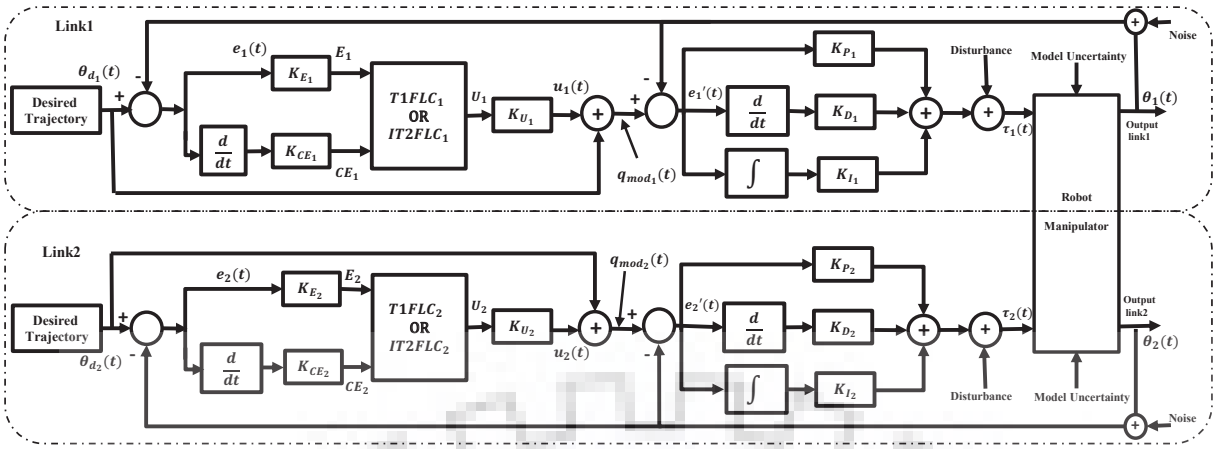


Fig. 5.6: Design scheme of T1FP-PID and the proposed IT2FP-PID controllers applied to robotic manipulator.

Table 5.2: Overall tuning parameters of PID, FPID, T1FP-PID, and IT2FP-PID controllers.

Controllers	Links	Antecedent MF parameters	Scaling factors	Total
PID	Link1	–	$K_{P_1}, K_{I_1}, K_{D_1}$	3
	Link2	–	$K_{P_2}, K_{I_2}, K_{D_2}$	3
FPID	Link1-I1/I2	Fixed Triangular MFs as shown in Fig. 2.3 (a)	$K_{E_1}, K_{CE_1}, K_{P_1}, K_{I_1}$	4
	Link2-I1/I2		$K_{E_2}, K_{CE_2}, K_{P_2}, K_{I_2}$	4
T1FP-PID	Link1-I1	$l_{11_1}, c_{11_1}, r_{11_1}, l_{12_1}, c_{12_1}, r_{12_1}, l_{13_1}, c_{13_1}, r_{13_1}$	$K_{E_1}, K_{CE_1}, K_{U_1}, K_{P_1}, K_{I_1}, K_{D_1}$	24
	Link1-I2	$l_{21_1}, c_{21_1}, r_{21_1}, l_{22_1}, c_{22_1}, r_{22_1}, l_{23_1}, c_{23_1}, r_{23_1}$		
	Link2-I1	$l_{11_2}, c_{11_2}, r_{11_2}, l_{12_2}, c_{12_2}, r_{12_2}, l_{13_2}, c_{13_2}, r_{13_2}$	$K_{E_2}, K_{CE_2}, K_{U_2}, K_{P_2}, K_{I_2}, K_{D_2}$	24
	Link2-I2	$l_{21_2}, c_{21_2}, r_{21_2}, l_{22_2}, c_{22_2}, r_{22_2}, l_{23_2}, c_{23_2}, r_{23_2}$		
IT2FP-PID	Link1-I1	$l_{11_1}, c_{11_1}, r_{11_1}, h_{11_1}, l_{12_1}, c_{12_1}, r_{12_1}, h_{12_1}, l_{13_1}, c_{13_1}, r_{13_1}, h_{13_1}$	$K_{E_1}, K_{CE_1}, K_{U_1}, K_{P_1}, K_{I_1}, K_{D_1}$	30
	Link1-I2	$l_{21_1}, c_{21_1}, r_{21_1}, h_{21_1}, l_{22_1}, c_{22_1}, r_{22_1}, h_{22_1}, l_{23_1}, c_{23_1}, r_{23_1}, h_{23_1}$		
	Link2-I1	$l_{11_2}, c_{11_2}, r_{11_2}, h_{11_2}, l_{12_2}, c_{12_2}, r_{12_2}, h_{12_2}, l_{13_2}, c_{13_2}, r_{13_2}, h_{13_2}$	$K_{E_2}, K_{CE_2}, K_{U_2}, K_{P_2}, K_{I_2}, K_{D_2}$	30
	Link2-I2	$l_{21_2}, c_{21_2}, r_{21_2}, h_{21_2}, l_{22_2}, c_{22_2}, r_{22_2}, h_{22_2}, l_{23_2}, c_{23_2}, r_{23_2}, h_{23_2}$		

I1:- Input1'$e(t)$', I2:- Input2 '$\Delta e(t)$'

link. The similar parameters for two links make total $30 \times 2 = 60$ tuning variables for whole control system applied to robotic manipulator. Thus, it is clear that the structure of IT2FP-PID controllers has 12 additional tuning parameters than the structure of T1FP-PID controllers. Thus, designers get extra freedom to tune, optimize, and handle the uncertainty in the problem. All the variables are tuned while fulfilling the constraints given in Eqs. (5.7) and (5.8).

In this design strategy, the proposed GWO-ABC algorithm is applied to tune overall parameters. It is important to mention here that the output consequent MFs are not optimized in all the FLCs presented in this study and will be kept same for the T1-FLC, and IT2-FLC structures as shown in Fig. 2.3 (c). Though, the optimization problems for T1FP-PID and IT2FP-PID controllers emerged as high-dimensional problems with constraints, the GWO-ABC is efficiently applied. Also, as the number of tuning parameters of IT2FP-PID is highest, it provides better

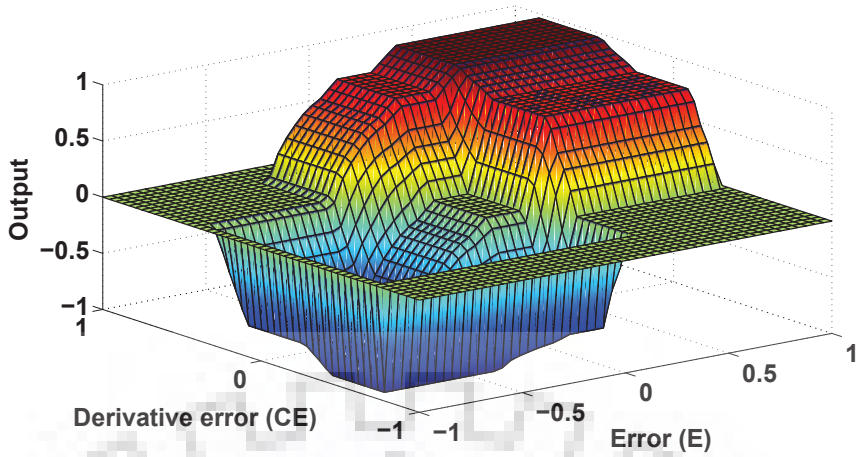


Fig. 5.7: Surface plot for rule base defined in Table 5.3.

DOF to design and results in better results.

The hardware implementation of the controller is a prominent research area in itself and some work using computers, microcontrollers, processors, Field Programmable Gate Array (FPGAs), etc., has been reported in several studies [132–135]. Similarly, in [136], authors reviewed and reported the different hardware implementations of fuzzy and neuro-fuzzy systems for various applications. As robotic model used in this study is having stationary base, computer interfaced controllers with suitable software are preferred. In case of mobile autonomous robots [74, 137], re-programmable embedded processors dedicated for particular application are utilized.

Table 5.3: Rule base for all FLC used in this study.

E ⇓ / CE ⇒	**N**	**Z**	**P**
N	NL	NM	Z
Z	NM	Z	PM
P	Z	PM	PL

5.3 Simulation Results and Discussion

In this work, we evaluated the performance of proposed IT2FP-PID controller against the PID, FPID, and T1FP-PID controllers for 2-link robotic manipulator after optimizing the various parameters using GWO-ABC algorithm. As problem definition demands to control trajectory of 2-link manipulator with variable payload, two separate controllers are applied for each link as shown in Figs. 5.5 and 5.6.

5.3.1 Problem Definition

In pursuance of implementing the optimization algorithms, proper selection of objective function (Obj_fun) is mainly required. As discussed earlier in Section 2.3, time-domain performance measures are employed to achieve desired performance in controller designing. This work prefers minimization of $ITAE$ - integral time absolute error for having the edge over other indices. Accordingly, the minimized value of $ITAE$ signifies the negligible peak overshoots, smaller rise time (t_r), and also indicates the ability of the response to reach the zero steady-state error E_{ss} within faster settling time (t_s). The main objective function is defined by the overall weighted sum as

$$Obj\_fun = w_1 \times \int t|e_{l1}(t)|dt + w_2 \times \int t|e_{l2}(t)|dt \tag{5.9}$$

here t indicates time and $e(t)$ is an error between current output and reference output. w_1 and w_2 are respective weights notified to Obj_fun_{l1} and Obj_fun_{l2}, which are considered as unity in this work.

As the robotic manipulator is controlled by two individual IT2FP-PID controllers for each link, the objective function is consolidated by separate $ITAE$ for Link1 Obj_fun_{l1} and Link2 Obj_fun_{l2} defined in previous chapter.

5.3.2 Parameter Settings

The parameter settings for conducting the optimization procedure of all the controllers are kept similar for fair comparison. In GWO-ABC algorithm, population of 30 search agents is taken and maximum iterations are fixed to 100. The final count of iteration also acts as a stopping criterion. The optimization procedure is executed for 20 independent runs for all the controllers and optimal results are reported. To render wider high-dimensional search space, all the parameters are bounded in the range (L_b, U_b) of these parameters, specified as, $K_{E_i} \in [0,500]$, $K_{CE_i} \in [0,500]$, $K_{U_i} \in [0,500]$, and $K_{P_i}, K_{I_i}, K_{D_i} \in [0,500]$, where $i = 1,2$.

The complete system model is designed in MATLAB Simulink environment and the optimization procedure is carried out in MATLAB version 2014b. Simultaneously, Fuzzy Logic Toolbox by MATLAB for T1-FLS and modified open source Type-2 Fuzzy Toolbox [72] for IT2-FLS are employed for corresponding simulations.

5.3.3 Results and Discussion

The optimized scaling parameters and accomplished performance index values are reported in Table 5.4. The general structures of MFs for different inputs in FPID and T1FP-PID controllers are shown in Fig. 5.2 (a). Similarly, the general MF structures applied to IT2FP-PID controller are shown in Fig. 5.2 (b). After optimization of T1FP-PID and IT2FP-PID controllers, the optimized MF structures are obtained as shown in Figs. 5.8 and 5.9, respectively.

Table 5.4: Optimal controller parameters and comparative results of the IT2FP-PID controller and others.

Controllers ⇒	PID		FPID		T1FP-PID		IT2FP-PID	
Parameter ⇓	Link1	Link2	Link1	Link2	Link1	Link2	Link1	Link2
K_P	469.12	490.34	134.91	20.574	423.28	494.27	19.268	192.12
K_I	99.767	1.0952	13.081	0.0067	81.996	45.037	458.62	9.7808
K_D	249.50	486.21	–	–	0.0642	0.3725	0.0015	0.0023
K_E	–	–	26.599	15.574	65.862	474.51	486.41	226.37
K_{CE}	–	–	0.1815	1.0008	0.0001	0.0019	0.0621	0.0031
K_U	–	–	–	–	98.527	2.9975	72.780	23.9291
ITAE	0.09120	0.02140	0.01937	0.01314	0.003572	0.000159	**0.00000241**	**0.00000577**

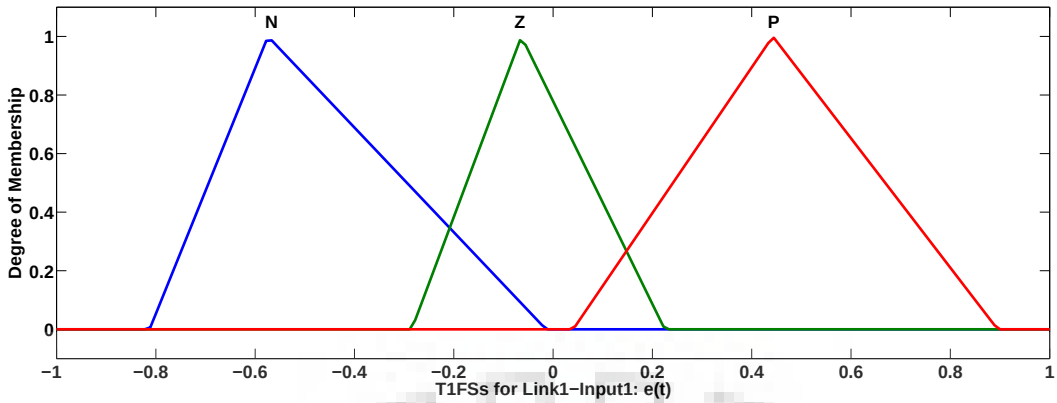
The "best results" are indicated by bold values.

We can observe that the *ITAE* values by IT2FP-PID controller after completion of stopping criterion of GWO-ABC algorithm for Link1 and Link2 are 2.41×10^{-06} and 5.76×10^{-06}, respectively. To substantiate the excellence of IT2FP-PID controller, *ITAE* values by all other controllers for both the links are demonstrated in Fig. 5.11. It is appreciated that the values of *ITAE* for both the links by IT2FP-PID controller are much smaller compared to values produced by other controllers. These minimal values clearly evince the accurate tracking of desired trajectories by the end-effector of robotic arm.

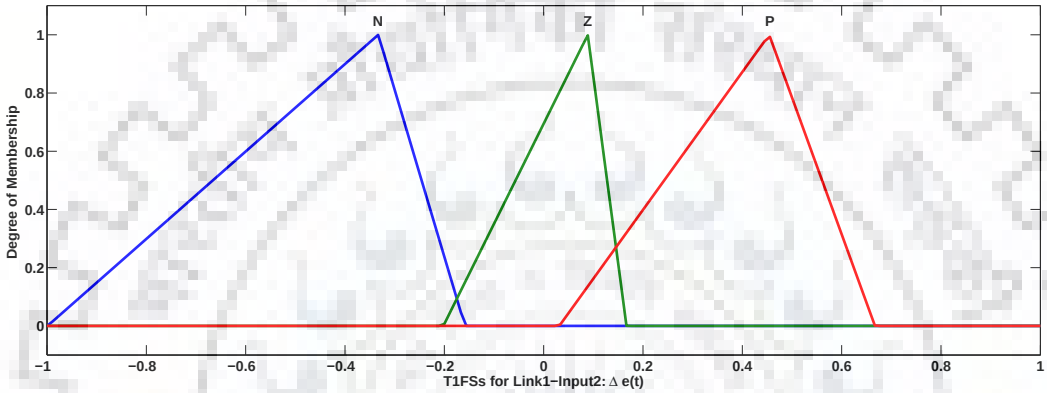
Ultimately, the most remarkable observation to emerge from the data is that the scaling factors and MF parameters of IT2FP-PID controllers are excellently tuned to optimal values using the proposed GWO-ABC algorithm to accomplish the best values of performance metric, and hence, meticulous tracking of defined desired path is achieved.

To provide further evidence of superior performance of IT2FP-PID controller, the plots of desired trajectory tracking by the links, X and Y coordinate versus time variations, and path traced by manipulators end-effector against the predefined desired reference path are displayed in Figs. 5.10 (a), (b), and (c), respectively. It can be apparently convinced from the plots and enlarged illustrations therein that the IT2FP-PID controller produces superior results than other counterparts. Additionally, in support to above findings, Fig. 5.10 (d) demonstrates the plot of position errors for both the links. Here, we can clearly depict that IT2FP-PID controller effectively minimizes the position errors from start of the process. Ultimately, it can be stated that the incorporation of IT2-FLC based precompensator in the proposed controller effectively eradicates the overshoots and undershoots and accordingly enhance the performance.

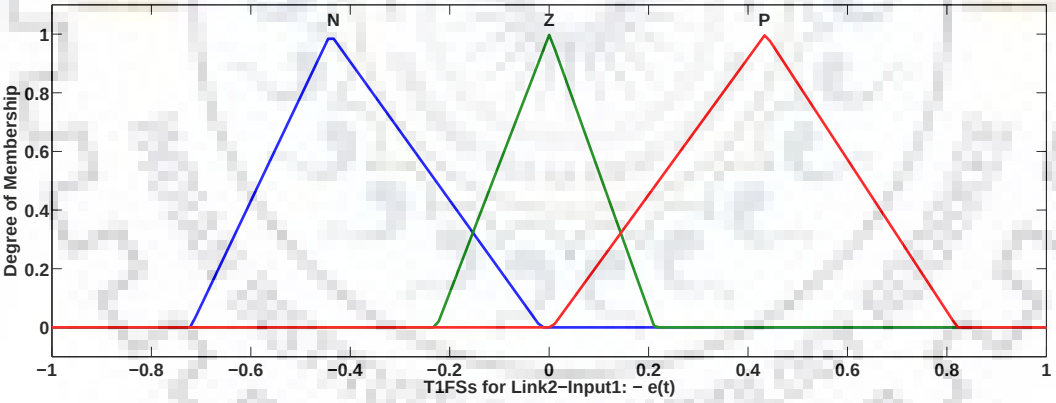
In sum, all these plots provide indisputable evidences about superiority of IT2FP-PID controller over other controllers and conforms the exceptional agreement between the IT2-FLC and precompensator. Also, the overall illustrations and results led us to confirm the efficacy of GWO-ABC algorithm in solving complex high-dimensional optimization problems.



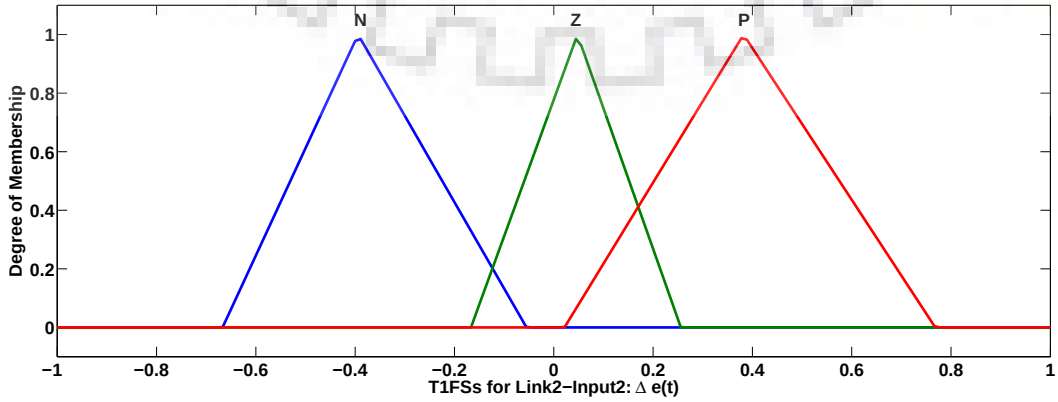
(a) Link1-I1



(b) Link1-I2

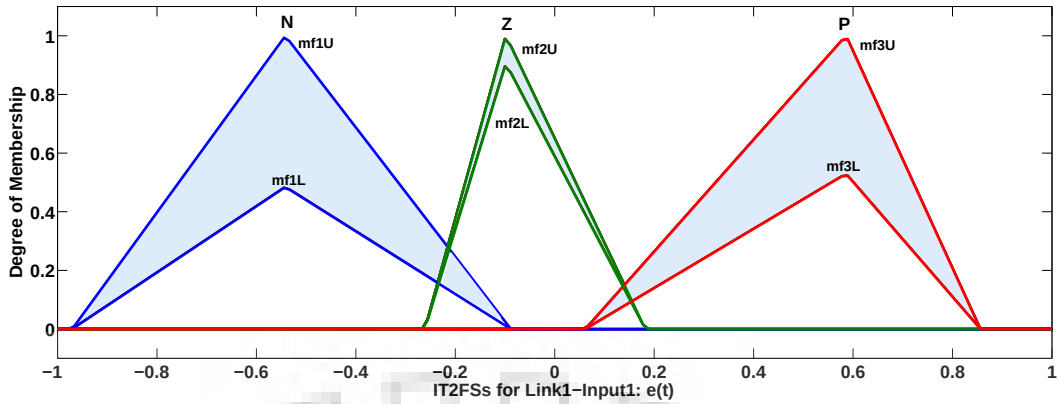


(c) Link2-I1

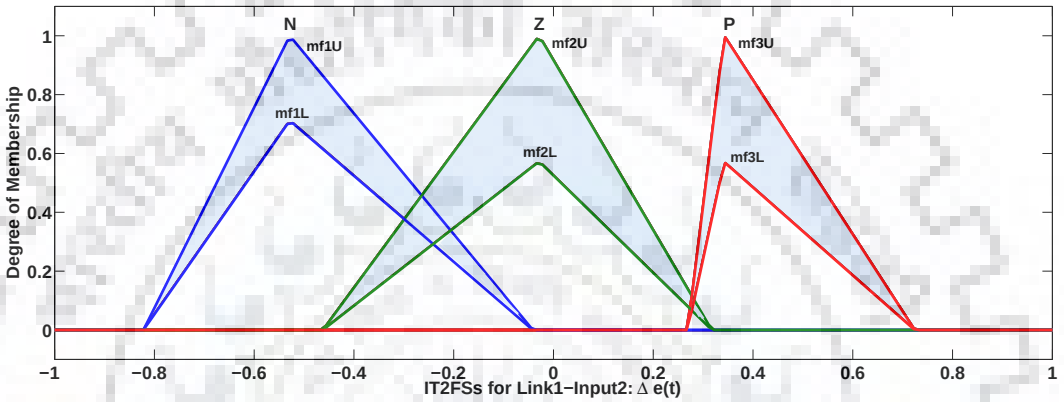


(d) Link2-I2

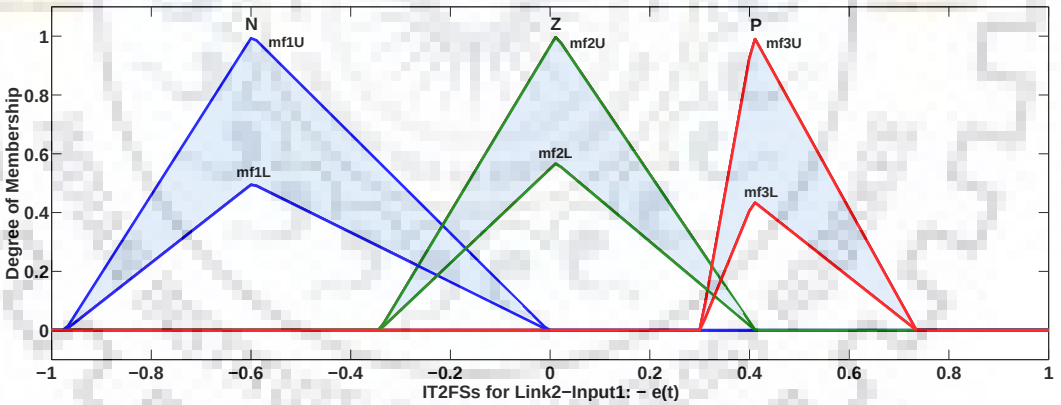
Fig. 5.8: Illustration of optimized antecedent MFs applied to T1FP-PID controller.



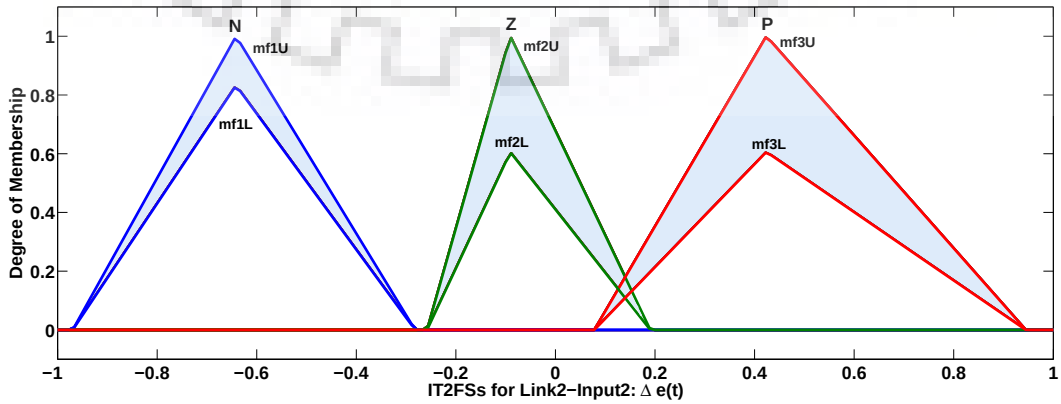
(a) Link1-I1



(b) Link1-I2

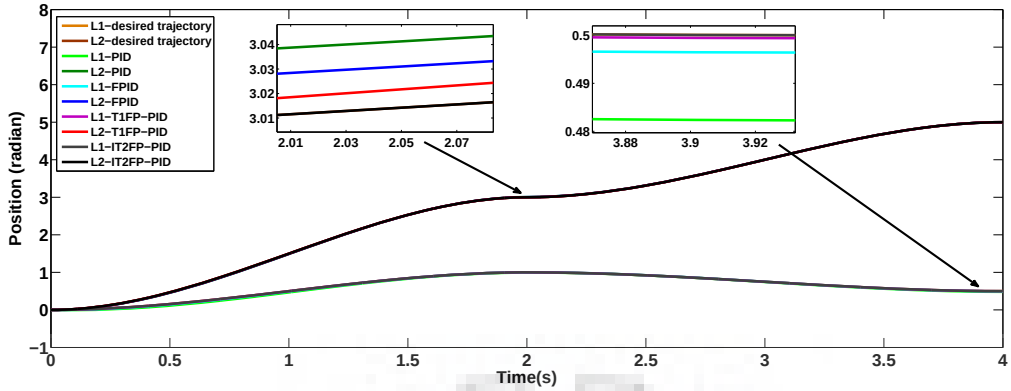


(c) Link2-I1

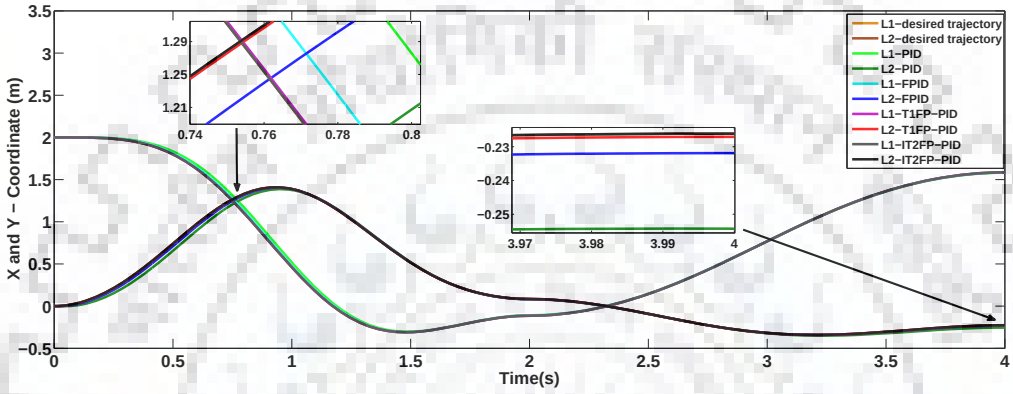


(d) Link2-I2

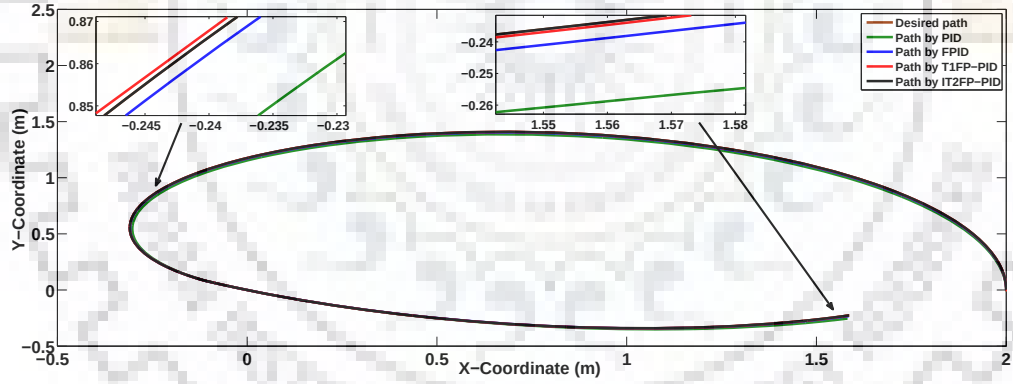
Fig. 5.9: Illustration of optimized antecedent MFs applied to IT2FP-PID controller.



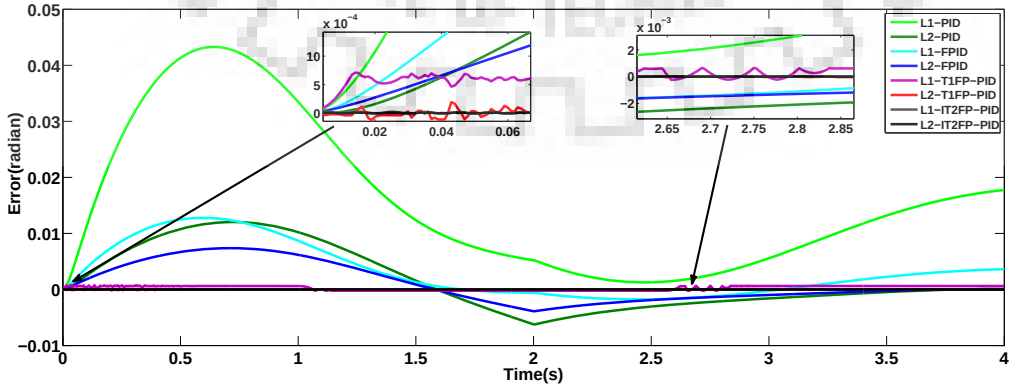
(a) Trajectory tracking performances



(b) X and Y coordinates versus time variation



(c) Path tracked by end-effector



(d) Position errors

Fig. 5.10: Comparative illustrations of different performances of 2-link robotic manipulator with variable payload.

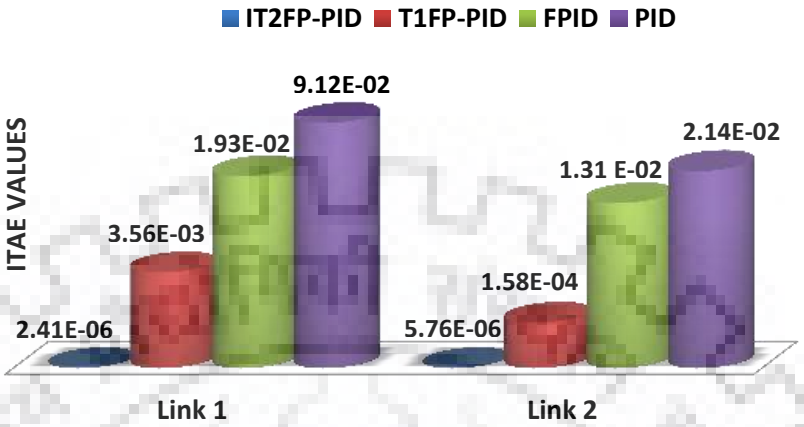


Fig. 5.11: Comparative illustrations of variation in *ITAE* for 2-link robotic manipulator by different controllers.

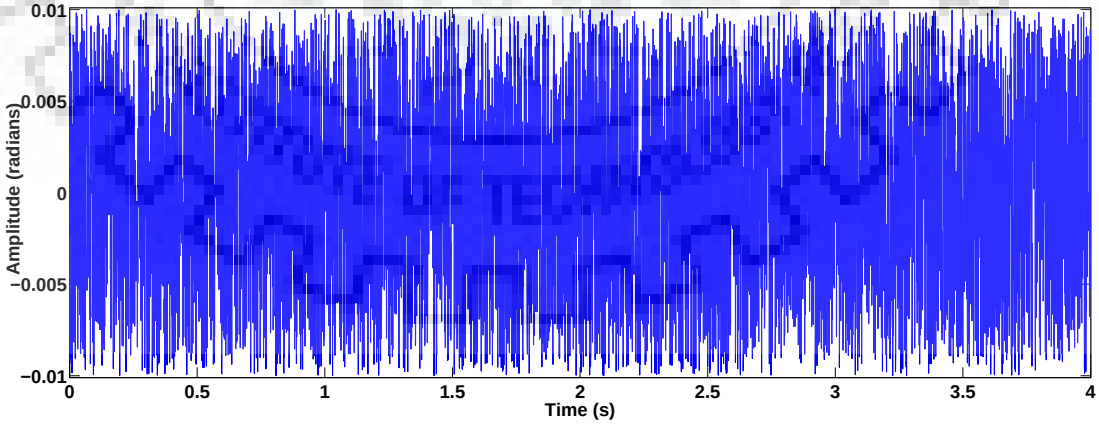


Fig. 5.12: Noise profile.

5.4 Robustness Analysis

In control system design, robustness analysis of systems under consideration is absolutely essential under unusual circumstances. If various parameters in the system are highly uncertain, control decisions must be staged accordingly. Although, the relevance of IT2FP-PID controller to minimize *ITAE* is clearly supported by the above findings, the effectiveness of IT2FP-PID controller is further validated by robustness analysis for model uncertainties, external disturbances, and random noise in the system.

5.4.1 Impact of Model Uncertainties

The variation in physical parameters of robotic manipulator may influence its orderly working and affect the performance thereupon. All rotating mechanisms are affected by static and dynamic frictional forces. Some of the friction coefficients are reflected in the robotic mathematical model in Appendix A. These frictions, including Viscous and Coulomb friction, vary with joint lubricants and other effects [111]. The other physical parameters like mass and length may also vary with the environmental changes. Consequently, the employed controller needs to be robust against such physical model variations and uncertainties.

In virtue of this, the proposed IT2FP-PID controller is investigated for four crucial parameter variations. The different parameters are varied as (a) masses m_1, m_2 are varied by $\pm 5\%$, (b) lengths l_1, l_2 are varied by $\pm 5\%$, (c) coefficients of viscous friction v_1, v_2 are varied by $\pm 20\%$, and (d) coefficients of dynamic friction d_1, d_2 are varied by $\pm 20\%$. These parameters are individually and collectively regulated alternately by increasing or decreasing their values and the performance index *ITAE* is observed and reported in Table 5.5. Finally, the overall average percentage variation with respect to original *ITAE* values are calculated and noted in last row. From the Table 5.5, it is observed that the variations in IT2FP-PID controller results are having negligible change (1.25%). Whereas, other controller performances like PID (6.778%), FPID (5.124%), and T1FP-PID (4.796%) are varied by considerably larger values which affect their robustness against the physical parameter variations. Ipso facto, the overall findings clarify the superiority of IT2FP-PID controller over others and authenticate the impression of FOU in MFs to ensure the boundedness of the position tracking error.

5.4.2 Rejection of External Disturbances

The industrial process control system and the equipment, instruments, devices, and machines therein always suffer from the influences of external disturbances. In this section, the performance of proposed IT2FP-PID controller is examined for the disturbances incorporated at control output of Link1, Link2, and both as illustrated in Figs. 5.5 and 5.6.

Distinct disturbance signals of the nature '$A\ sin(\omega t)$' *N-m* were applied for the entire 4s and the values of *ITAE* for IT2FP-PID, T1FP-PID, FPID, and PID controllers are noted. The results

Table 5.5: Comparative *ITAE* values obtained by different controllers for variations in modelling parameters.

Controllers ⇒	PID		FPID		T1FP-PID		IT2FP-PID	
Parameter ⇓	Link1	Link2	Link1	Link2	Link1	Link2	Link1	Link2
Optimum *ITAE*	0.09120	0.02140	0.01937	0.01314	0.00357	0.000159	0.00000241	0.00000577
m_1 (-5%)	0.08610	0.02031	0.01801	0.01242	0.00369	0.000172	0.00000241	0.00000576
$m_2(-5\%)$	0.08620	0.02053	0.01811	0.01201	0.00329	0.000161	0.00000240	0.00000571
m_1+m_2 (-5%)	0.08630	0.02041	0.01836	0.01205	0.00366	0.000171	0.00000238	0.00000571
l_1 (-5%)	0.08779	0.02393	0.01795	0.01357	0.00381	0.000162	0.00000238	0.00000575
l_2 (-5%)	0.08785	0.02318	0.01803	0.01219	0.00380	0.000161	0.00000237	0.00000574
l_1+l_2 (-5%)	0.08789	0.02218	0.01875	0.01394	0.00371	0.000158	0.00000239	0.00000599
All m and l (-5%)	0.08316	0.02037	0.01777	0.01251	0.00365	0.000169	0.00000242	0.00000601
v_1+v_2 (-20%)	0.09124	0.02158	0.01936	0.01326	0.00363	0.000155	0.00000235	0.00000568
d_1+d_2 (-20%)	0.09134	0.02155	0.01939	0.01324	0.00357	0.000154	0.00000233	0.00000567
All d and v (-20%)	0.09137	0.02175	0.01930	0.01337	0.00373	0.000182	0.00000235	0.00000566
All above	0.08332	0.02075	0.01779	0.01274	0.00369	0.000175	0.00000242	0.00000601
m_1 $(+5\%)$	0.09611	0.02233	0.01810	0.01211	0.00381	0.000161	0.00000240	0.00000576
m_2 $(+5\%)$	0.09614	0.02237	0.01790	0.01269	0.00381	0.000158	0.00000241	0.00000575
m_1+m_2 $(+5\%)$	0.09613	0.02236	0.01836	0.01252	0.00377	0.000158	0.00000237	0.00000570
l_1 $(+5\%)$	0.09445	0.02236	0.02010	0.01380	0.00381	0.000189	0.00000242	0.00000577
l_2 $(+5\%)$	0.09450	0.02217	0.02102	0.01357	0.00391	0.000185	0.00000245	0.00000576
l_1+l_2 $(+5\%)$	0.09451	0.02343	0.02002	0.01355	0.00374	0.000156	0.00000237	0.00000565
All m and l $(+5\%)$	0.09261	0.02238	0.02106	0.01376	0.00374	0.000168	0.00000242	0.00000571
v_1+v_2 $(+20\%)$	0.09180	0.02119	0.01936	0.01303	0.00366	0.000156	0.00000233	0.00000576
d_1+d_2 $(+20\%)$	0.09108	0.02122	0.01935	0.01305	0.00364	0.000163	0.00000237	0.00000576
All v and d $(+20\%)$	0.09105	0.02105	0.01935	0.01294	0.00364	0.000166	0.00000233	0.00000577
All above	0.09946	0.02204	0.02104	0.01357	0.00373	0.000158	0.00000244	0.00000581
Overall % variation	6.778%		5.124%		4.796%		**1.25%**	

All variables are as per Table A.1 for respective links. The "best result" is indicated by bold.

are reported in Table 5.6. The variation in *ITAE* for all four controllers is observed by SD values mentioned separately for each link. We can clearly observed that the SD values obtained for IT2FP-PID controllers are much smaller than that of others, and so, shows the performance stability against disturbance variations. Thus, IT2FP-PID controller exhibits superiority over others in terms of variations due to disturbances. To elaborate further details, the graphs of desired trajectory tracking by the links, X and Y coordinate versus time variations, path traced, and position errors - for disturbance of $2\,sin\,(50t)$ *N-m* at both the links - are illustrated in Figs. 5.13 (a), (b), (c), and (d), respectively. Taken together, the SD values in Table 5.6 and the plots in Fig. 5.13 provide important insights to evidence the superiority of IT2FP-PID controller for the disturbance rejection.

Table 5.6: Comparative *ITAE* values obtained by different controllers for variable disturbance.

Controllers ⇒		PID		FPID		T1FP-PID		IT2FP-PID	
Disturbance ⇓		Link1	Link2	Link1	Link2	Link1	Link2	Link1	Link2
Link1	$1\,sin\,(50t)$	0.09123	0.02138	0.01936	0.01314	0.00361	0.000177	0.00000325	0.00000570
	$1.5\,sin\,(50t)$	0.09124	0.02138	0.01936	0.01340	0.00369	0.000181	0.00000347	0.00000571
	$2\,sin\,(50t)$	0.09226	0.02338	0.02035	0.01404	0.00363	0.000164	0.00000330	0.00000571
	$1\,sin\,(25t)$	0.09125	0.02138	0.01934	0.01314	0.00367	0.000177	0.00000358	0.00000569
	$1.5\,sin\,(25t)$	0.09125	0.02138	0.01933	0.01324	0.00364	0.000162	0.00000332	0.00000575
	$2\,sin\,(25t)$	0.09221	0.02218	0.02134	0.01391	0.00363	0.000161	0.00000331	0.00000572
	SD	4.68E-4	7.45E-4	7.62E-4	3.64E-4	2.69E-5	8.16E-6	**1.15E-07**	**1.89E-08**
Link2	$1\,sin\,(50t)$	0.09121	0.02137	0.01937	0.01314	0.00366	0.000196	0.00000252	0.00000648
	$1.5\,sin\,(50t)$	0.09121	0.02137	0.01937	0.01314	0.00353	0.000198	0.00000244	0.00000689
	$2\,sin\,(50t)$	0.09121	0.02137	0.01997	0.01314	0.00356	0.000208	0.00000239	0.00000633
	$1\,sin\,(25t)$	0.09121	0.02137	0.01937	0.01315	0.00363	0.000197	0.00000240	0.00000637
	$1.5\,sin\,(25t)$	0.09121	0.02144	0.01937	0.01316	0.00373	0.000200	0.00000238	0.00000680
	$2\,sin\,(25t)$	0.09123	0.02444	0.01936	0.01312	0.00371	0.000201	0.00000231	0.00000678
	SD	7.45E-6	1.14E-3	2.24E-4	1.21E-5	7.30E-5	3.96E-6	**6.37E-08**	**2.22E-07**
Both	$1\,sin\,(50t)$	0.09123	0.02237	0.01936	0.01314	0.00365	0.000198	0.00000308	0.00000647
	$1.5\,sin\,(50t)$	0.09125	0.02037	0.01943	0.01334	0.00360	0.000199	0.00000331	0.00000646
	$2\,sin\,(50t)$	0.09186	0.02014	0.01935	0.01399	0.00364	0.000204	0.00000332	0.00000634
	$1\,sin\,(25t)$	0.09125	0.02038	0.01934	0.01315	0.00373	0.000197	0.00000337	0.00000643
	$1.5\,sin\,(25t)$	0.09124	0.02008	0.01953	0.01315	0.00363	0.000194	0.00000333	0.00000649
	$2\,sin\,(25t)$	0.09224	0.02214	0.01913	0.01443	0.00362	0.000191	0.00000333	0.00000658
	SD	3.96E-4	9.57E-4	1.21E-4	5.00E-4	4.11E-5	4.06E-6	**9.57E-08**	**7.15E-08**

The "best results" are indicated by bold values.

5.4.3 Suppression of Random Noise

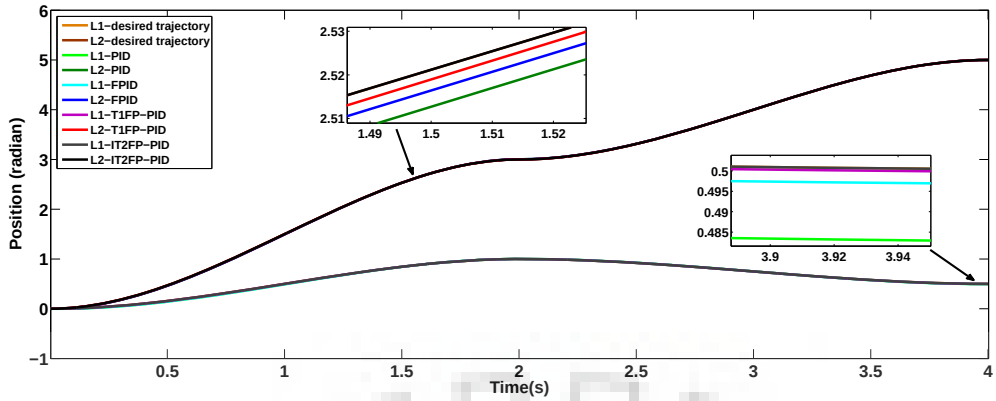
Sensors used in feedback path of control system used to add high frequency random noise with the measured signals. The source of this feedback noise signal is also illustrated in Figs. 5.5 and 5.6. The feedback noise induces some repercussions in overall control system and have to be eliminated by controllers. Here, we investigated the robustness of proposed IT2FP-PID controller for random noise. The low amplitude noise profile used in this study is shown in Fig. 5.12. The random noise is added in feedback path of Link1, Link2, and both links and performance index *ITAE* is measured. Thus, the values of *ITAE* for IT2FP-PID, T1FP-PID, FPID, and PID controllers in presence of random noise are summarized in Table 5.7. For comprehensive analysis, the illustrations showing desired trajectory tracking by the links, X and Y coordinate versus time variations, path traced, and position errors in presence of random noise in both the links are demonstrated in Figs. 5.14 (a), (b), (c), and (d), respectively. Ultimately, the results in Table 5.7 and plots in Fig. 5.14 lead us to conclude that the IT2FP-PID controller with optimized IT2-FS structures have succeeded in suppressing the added noise efficiently than other counterparts.

Significantly, the modified structures of IT2-MFs, as depicted in Fig. 5.9, utilized benefits of

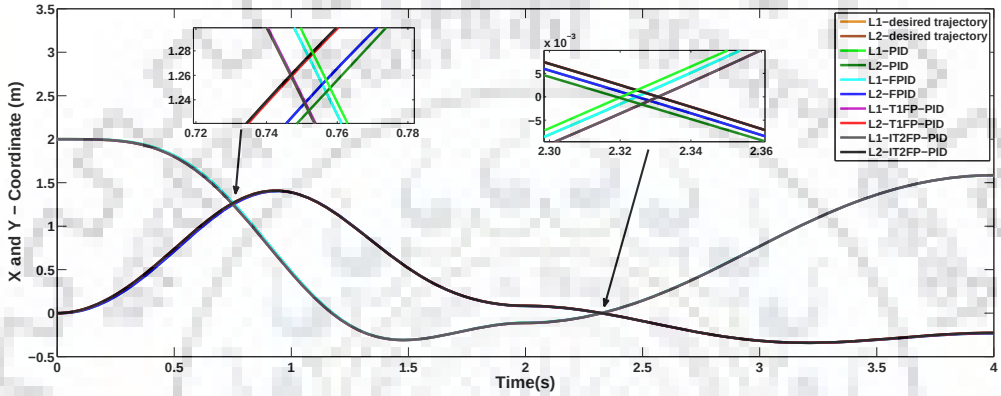
optimized FOU to restrain various uncertainties in the input signals. Even though the optimized shapes of T1-MFs are also applied through tuning of MF parameters for T1FP-PID controller as demonstrated in Fig. 5.8, it could not handle the uncertainty in equally efficient manner due to absence of FOU. As inferred, the performance of classical PID and FPID controllers with uniform MFs are rather disappointing and found unsuited for the complex systems with uncertainties.

Table 5.7: Comparative *ITAE* values obtained by different controllers in presence of random noise.

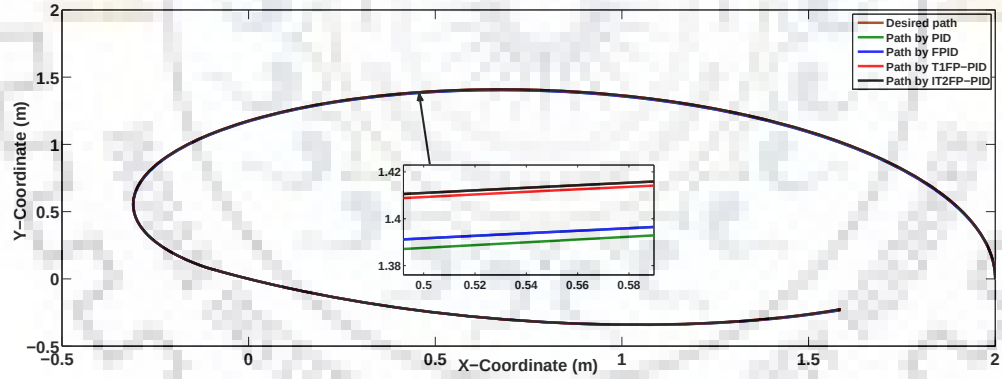
Controllers ⇒	PID		FPID		T1FP-PID		IT2FP-PID	
Random noise at ⇓	Link1	Link2	Link1	Link2	Link1	Link2	Link1	Link2
Link1	0.1241	0.03994	0.04895	0.01034	0.00415	0.006123	0.0004956	0.0004346
Link2	0.1045	0.04229	0.05134	0.01004	0.00409	0.006121	0.0005121	0.0004345
Both links	0.11373	0.04771	0.05782	0.01241	0.00422	0.006245	0.0005071	0.0004476



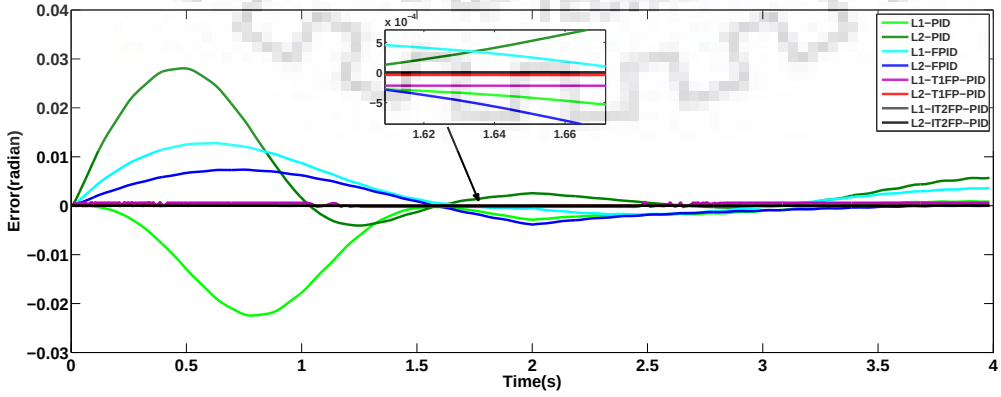
(**a**) Trajectory tracking performances



(**b**) X and Y coordinates versus time variation

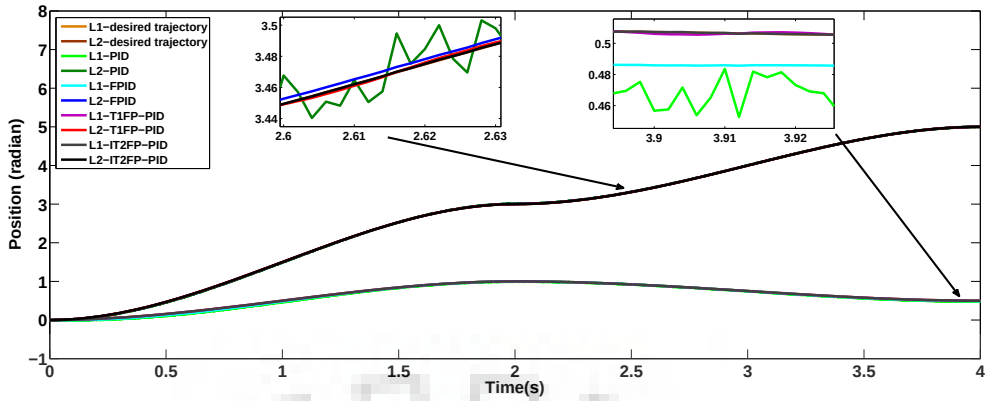


(**c**) Path tracked by end-effector

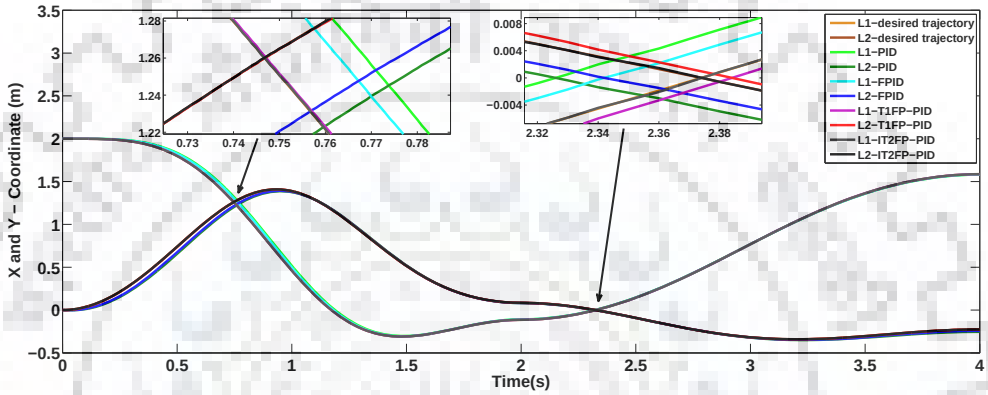


(**d**) Position errors

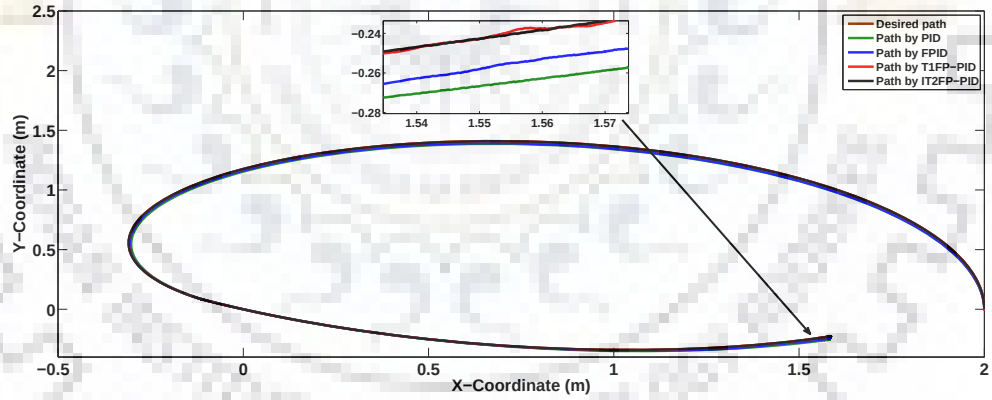
Fig. 5.13: Comparative illustrations of different performances of 2-link robotic manipulator with disturbance $2\,sin\,(50t)$ in both the links.



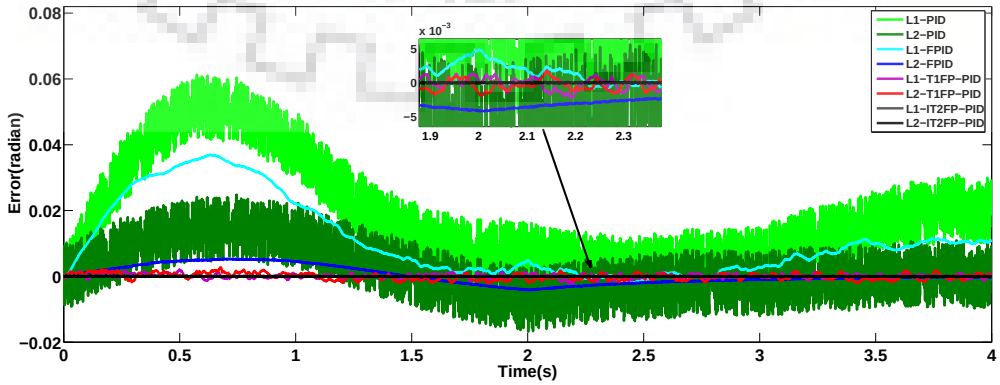
(a) Trajectory tracking performances



(b) X and Y coordinates versus time variation



(c) Path tracked by end-effector



(d) Position errors

Fig. 5.14: Comparative illustrations of different performances of 2-link robotic manipulator with noise in both the links.

5.5 Concluding Remarks

The main aim of this chapter is to design controller benefited by attributes of the IT2-FLC and FOU. Hence, we propose an efficient IT2FP-PID controller and present a systematic strategy for optimizing the controller parameters. It is clearly observed that IT2FP-PID controller effectively minimizes the performance index from start of the process. Ultimately, it can be stated that the incorporation of IT2-FLC based precompensator in the proposed controller effectively eradicates the overshoots and undershoots. The FLC based precompensator exhibits exceptional agreement between the IT2-FLC and precompensator and regulates the control signal to enhance the performance when the system has unknown uncertainties.

All illustrations and result analysis provide indisputable evidences about superiority of IT2FP-PID controller over other controllers. The efficacy of GWO-ABC algorithm for solving the complex high-dimensional optimization problems is demonstrated. The comparative illustrations are showing exact trajectory tracking by IT2FP-PID controller with minimum position errors. The exhaustive robustness analysis in presence of distinct non-linear dynamics evinces the superiority of IT2FP-PID controller in terms of robustness for (i) payload variations, (ii) model uncertainties, (iii) disturbance in signals, and (iv) random noise at feedback path. The overall illustrations and results confirm that optimized structures of antecedent MFs utilized benefits of FOU to restrain various uncertainties in the input signals. Even though the optimized shapes of T1-MFs are also applied through tuning of MF parameters for T1FP-PID controller, it could not handle the uncertainty in equally efficient manner due to absence of FOU. As inferred, the performance of classical PID and FPID controllers with uniform MFs are rather disappointing and found unsuited for the complex systems with uncertainties.

As a whole, it can be claimed that (a) additional tuning parameters provide extra degree of freedom to get better performance in optimal controller design, (b) in case of IT2-FLC, the systematic strategy to optimize the shapes of MFs derive maximum benefits of FOU to handle uncertainty (c) the proposed IT2FP-PID controller revealed as viable alternative to control complex non-linear systems with high uncertainties, (d) the proposed GWO-ABC algorithm can efficiently solve the low-dimensional and high-dimensional constrained optimization problems.

Chapter 6

Constrained Multi-objective Optimization Approach for Control System Design

The enhanced soft computing techniques proposed in previous chapters consider the optimization problems with single objective function. This chapter establishes the multi-objective problem based on two conflicting objective functions of control system and use the sensitivity function to determine the robustness. After introductory comments in Section 6.1, the MOO approach is discussed in Sections 6.2. Later, Section 6.3 explained the MLS and problem is specified in Section 6.4. Finally, the results and comparative performance analysis is reported in Section 6.5.

6.1 Introduction

In last two decades, many multi-objective evolutionary algorithms (MOEAs) were suggested and successfully applied to solve various optimization problems in engineering, computer science, industry, economics and operation research. Most of the MOEAs are developed to solve the optimization problems having two or three objective functions [138, 139]. Recently, many-objective optimization algorithms are presented to solve the problems with four or more objectives [140, 141]. While the optimization approaches using SOO methods give single optimal solution, MOO based approaches generate number of optimal solutions in the form of Pareto-optimal front [8, 142]. As discussed in previous chapters, the most of the control problems are single objective problems and SOO algorithms are successfully applied to solve them. If there are multiple objectives, weighted sum of different objectives is calculated to obtain single objective

The work outlined in this chapter has been disseminated in the following publications:

- P. J. Gaidhane, A. Kumar, and M. J. Nigam, "Tuning of two-DOF-FOPID controller for magnetic levitation system: A multi-objective optimization approach", *in Proceedings of Computer Application In Electrical Engineering - Recent Advances (CERA)*, pp. 479-484, 2017.
- P. J. Gaidhane and M.J. Nigam, "Multi-Objective robust design and performance analysis of two-DOF-FOPID controller for magnetic levitation system," *in the Proceedings of 14th IEEE India Council International Conference (INDICON)*, pp. 1-6, 2017.

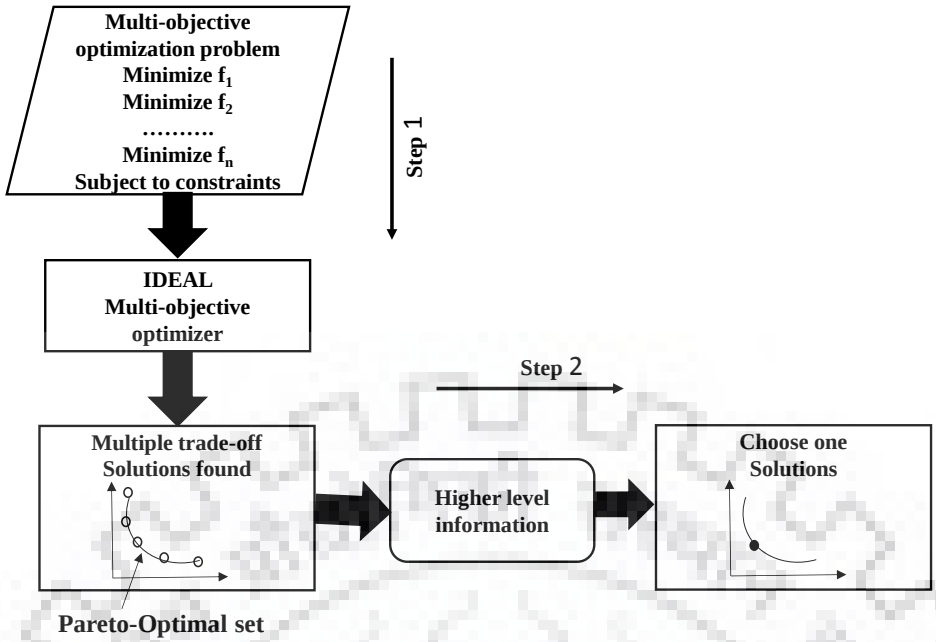


Fig. 6.1: Schematic of multi-objective optimization procedure.

function using Eq. (2.19) [143, 144]. In this chapter, two conflicting objectives are considered separately and MOO algorithm is employed.

The main aim of this chapter is to present a simple design and controller parameter tuning scheme using MOO approach. Initially, the multi-objective control system design problem is established with the constraints based on limits of sensitivity function. The performance index *IAE* for set-point tracking and external disturbance are considered as two conflicting objective functions. The NSGA-II algorithm is employed to tune the 2-DOF-FOPID controller parameters for MLS and the Pareto-optimal front is obtained. Finally, the comparative performance analysis of 2-DOF-FOPID controller is carried out against classical PID, FOPID, and 2-DOF-PID controllers.

6.2 Multi-Objective Optimization

In MOO, the diverse solutions may produce trade-off (conflicting scenario) between different objectives. In this condition, decision on single best solution requires a compromise in other objectives. The principles of an ideal multi-objective optimization procedure can be seen in schematic given in Fig. 6.1

Fundamentally, the MOO problem with conflicting objectives produces many optimal solutions satisfying one or many objectives. Since number of optimal solutions are obtained, it is always challenging to conclude the single best solution with respect to all the objectives. Many approaches have been suggested using higher level information for finding the set of trade-off optimal solutions. Generally, the MOO procedure is defined in following steps.

1. Find a set of optimal solutions in the form of Pareto-optimal front.

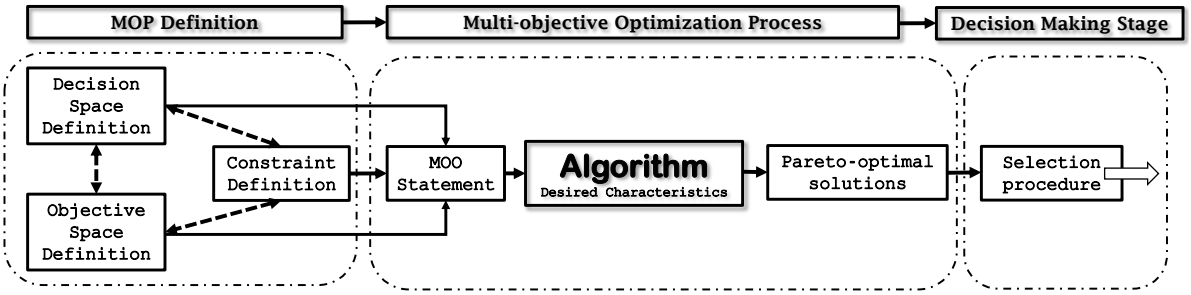


Fig. 6.2: A multi-objective optimization design procedure for control system problems.

2. Choose single optimal solutions using higher-level information.

The MOEAs are widely applied to find the set of optimal solutions. Thus, the step 1 in Fig. 6.1 is executed to generate the Pareto-optimal solution set. This set is demonstrated as Pareto-optimal front in the objective space. In MOO procedure, the obtained Pareto-optimal solution set should show (a) convergence - solutions are close to the actual Pareto-optimal front and (b) divergence - solution set is spread equally over Pareto-optimal front. A MOO design procedure for control systems problems is demonstrated in Fig. 6.2. The constraints have to be defined meticulously in such problems to obtain the feasible optimal solutions [145]. The MOO problem is defined using two conflicting objective functions. In this study, the constraint handling scheme based on penalty function is adopted to solve these problems [146].

6.3 Mathematical Model of Magnetic Levitation System

The MLS is shown in Fig. 6.3 where the steal ball is levitated in air with the help of electro-magnets. The magnetic force is balanced with gravitational force mg. The position of the ball is sensed and passed as feedback signal for closed loop control. Mostly, light sensors are used to detect the distance variations and control current is used as controller output.

Mathematical model of MLS can be obtained from the ball kinematics and electrodynamics equations [9, 147]. The balancing equation between applied magnetic force F and gravitational force acting on the ball is expressed as

$$m\frac{d^2x}{dt^2} = F(i,x) + mg \tag{6.1}$$

where x is the distance between magnet and center of the ball, m is mass of steel ball, i is current to electromagnet, and g is acceleration of gravity.

The electromagnetic force generated by controlling current signal is given by

$$F(i,x) = k(\frac{i}{x})^2 \tag{6.2}$$

where k is constant proportional to physical parameters.

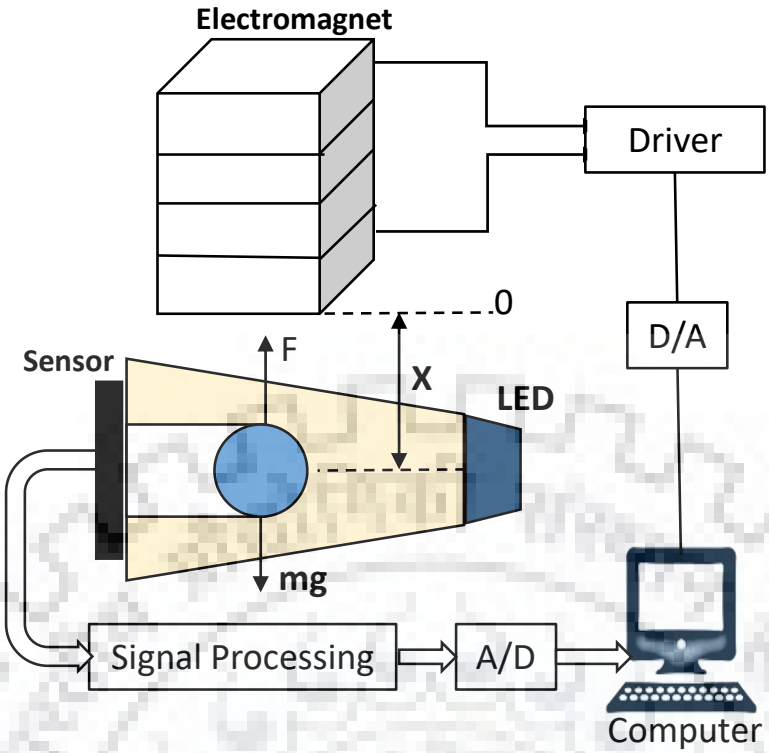


Fig. 6.3: Model of magnetic levitation system.

As seen above, air gap x and the current i are non-linearly related, hence, first the system have to be linearized at equilibrium point (i_0, x_0). Using Tylor's expansion and ignoring higher-order terms, (6.2) becomes

$$F(i,x) = F(i_0,x_0) + F_i(i_0,x_0)(i-i_0) + F_x(i_0,x_0)(x-x_0) \tag{6.3}$$

In this, $F(i_0,x_0)$ represents the magnetic force required to balance the ball gravitational force, when the current is i_0 and air gap is x_0.

$$F(i_0,x_0) = mg \tag{6.4}$$

$$K_i = F_i(i_0,x_0) = \frac{\delta F(i,x)}{\delta i}|(i_{i_0,x=x_0}) = \frac{2K_{i_0}}{x_0^2} \tag{6.5}$$

$$K_x = F_x(i_0,x_0) = \frac{\delta F(i,x)}{\delta x}|i_{i_0,x=x_0} = -\frac{2K_{i_0}^2}{x_0^3} \tag{6.6}$$

here k_i is stiffness coefficient of the magnetic force to current and k_x represents stiffness coefficient due to air gap obtained at the equilibrium point. Therefore, the equation of whole system can be represented as

$$F(i,x) = K_i i + K_x x + F(i_0,x_0) \tag{6.7}$$

$$m\frac{d^2x}{dt^2} = K_i(i-i_0) + K_x(x-x_0) \tag{6.8}$$

The voltage equation of electromagnetic coil can be written as

$$U(t) = Ri(t) + L(\frac{di}{dt}) \tag{6.9}$$

here L represents the static inductance between magnetic field and ball.

All above equations are used to define state variable model of the MLS.

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{2g}{x_0} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{2gK_S}{K_a} \end{bmatrix} u_{in} \tag{6.10}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = x_1 \tag{6.11}$$

The dimensions and other parameters of the MLS model used in this study are given as [148], mass of steel ball $m = 0.22(kg)$, radius of ball $r = 0.0125(m)$, $K_a = 5.8929$, $K_s = 458.7204$, and $g = 9.8(m/s^2)$.

Also, it is found that, at equilibrium point $i_0 = 0.6105(A)$ and $x_0 = 0.03(m)$. Putting these system parameter values in (6.10) and (6.11), we obtained

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 980.0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 2499.1 \end{bmatrix} u_{in}$$

$$y = x_1$$

Applying state variable to transfer function conversion, finally we have

$$G(s) = \frac{77.8421}{0.0311s^2 - 30.5250} \tag{6.12}$$

The above equation clearly shows that their is an open loop pole at right side plane, which makes the MLS essentially unstable system.

6.4 Implementation of Multi-Objective Algorithm

In recent years, several types of FOPID controllers were suggested for controlling various industrial processes. In [80], the 2-DOF-FOPID controller is presented for robotic manipulator with payload. Cuckoo search algorithm (CSA) is applied for parameter tuning and the enhancement of robustness of controller towards trajectory tracking, parameter variation, and disturbance rejection is claimed. This section explains the same structure of 2-DOF-FOPID controller and

proposed its implementation to MLS system. The basic structure of 2-DOF-controller applied to MLS system is shown in Fig. 6.4.

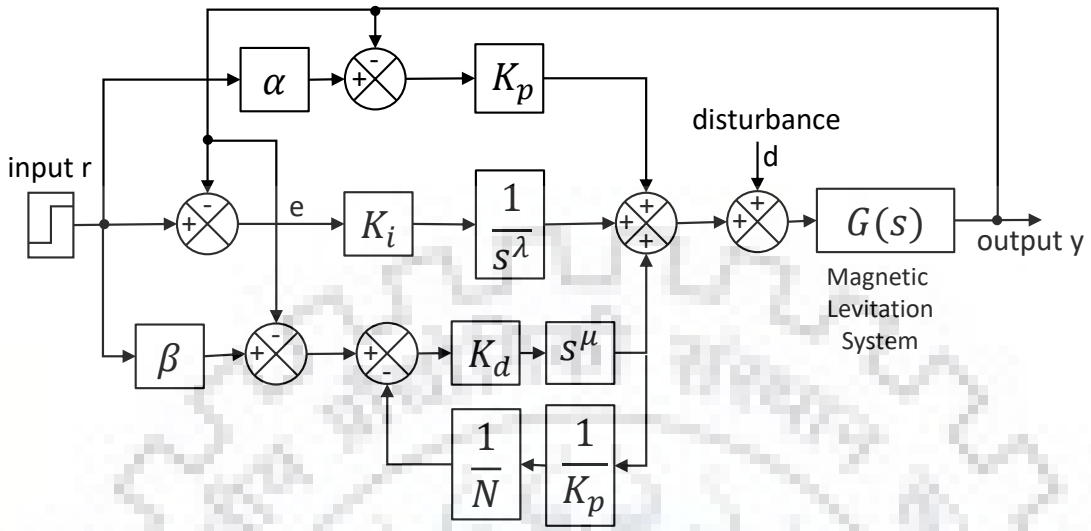


Fig. 6.4: Design scheme of 2-DOF FOPID controller applied to magnetic levitation system.

In general, the output control signal of single-DOF-FOPID controller is given as

$$U^1(s) = (K_p + K_i\frac{1}{s^\lambda} + K_d s^\mu)E_i(s) \tag{6.13}$$

where K_p, K_i, and K_d are proportional gain, integral gain, and derivative gain, respectively; μ is the fractional derivative value; λ represents fractional integral value; and e is the error signal between desired set point and actual output.

It can be observed that, two-closed loop transfer functions are introduced in the 2-DOF-FOPID controller to provide effective simultaneous control for both set point tracking and external disturbance rejection. In the proposed controller, two parameters α and β are assigned for differentiating the desired signal from actual output and N is derivative filter coefficient. The equation for overall control output of the 2-DOF-FOPID controller can be expressed as [77]

$$U^2(s) = K_p(\alpha R(s) - Y(s)) + \frac{K_i}{s^\lambda}(e(s)) + \frac{K_d s^\mu(\beta R(s) - Y(s))}{1 + \frac{K_d s^\mu}{K_p N}} \tag{6.14}$$

where $U^2(s)$ is controller output of 2-DOF-FOPID controller; $R(s)$ is desired set point and $Y(s)$ is actual output; and $e(s) = R(s) - Y(s)$. The controller have eight tunable parameters, given as $[Kc, K_i, K_d, N, \alpha, \beta, \lambda, \mu]$. Different controller structures can be designed by proper selection of these parameters [146]. The PID, 2-DOF-PID, and FOPID controllers, used for performance comparison, are formed by assigning the following values to the parameters. The remaining parameters are tuned by MOO technique for getting optimal performance index and the results are compared.

$$\text{PID: } \Rightarrow U(s) = [Kc, K_i, K_d],\ \alpha = \beta = 1,\ \frac{1}{N} = 0,\ \lambda = \mu = 1.$$
$$\text{2-DOF-PID: } \Rightarrow U(s) = [Kc, K_i, K_d, N, \alpha, \beta],\ \lambda = \mu = 1.$$
$$\text{FOPID: } \Rightarrow U(s) = [Kc, K_i, K_d, \lambda, \mu],\ \alpha = \beta = 1,\ \frac{1}{N} = 0.$$

6.4.1 Non-dominated Sorting Genetic Algorithm-II

Though, several MOEA were proposed in last two decades, a fast and elitist NSGA-II [8] algorithm is widely studied and used in several applications. In this algorithm innovative fast non-dominated sorting approach, an elitist-preserving approach, crowding distance mechanism,and a parameter-less niching operator are used for better convergence and diversity of solutions. Brief stepwise explanation of NSGA-II algorithm is given below and readers are encouraged to refer [8] for detailed description.

- Initially, a random parent population P_0 is created for variables $\mathbf{x} = (x_1, x_2, .., x_n)^T$ between respective lower bound l_b and upper u_b bound, using
  $x_{ij} = l_{b_j} + rand(0,1) * (l_{u_j} - l_{b_j})$.
  Where $i = 1, 2, ..N$ (size of populations) and $j = 1, 2, ..n$ (number of variables).
- Each solution is sorted based on non-dominated sorting and rank is assigned to them. The lowest rank is considered as the best fitness.
- The GA is applied to produce the offspring population Q_0 of size N. In GA only reproduction and mutation operators are used. These Q_0 is also assigned the fitness using non-dominated sorting.
- The elitism is adopted by combining the parent and offspring populations as : $R_t = P_t \cup Q_t$, where t indicates the iteration count and its size is $2N$. Then, sort R_t using non-dominated sorting to identify relevant fronts: $F_i = 1, 2, .., etc$.
- To create new population $P_{(t+1)}$ of size N for next generation, solution of low ranked fronts (better) are selected first until the size is less than or equal to N. If after adopting some fronts, still the size is less than N, the next front solutions are selected based on their higher crowding distance.
- Again, use GA operators to produce new offspring solutions $Q_{(t+1)}$ from $P_{(t+1)}$. Repeat the same process for defined number of iterations or up to exit criteria.

6.4.2 Problem Definition

Several specifications and requirements are to be accomplished in design of control engineering problems. As different conflicting objective functions are considered simultaneously in con-

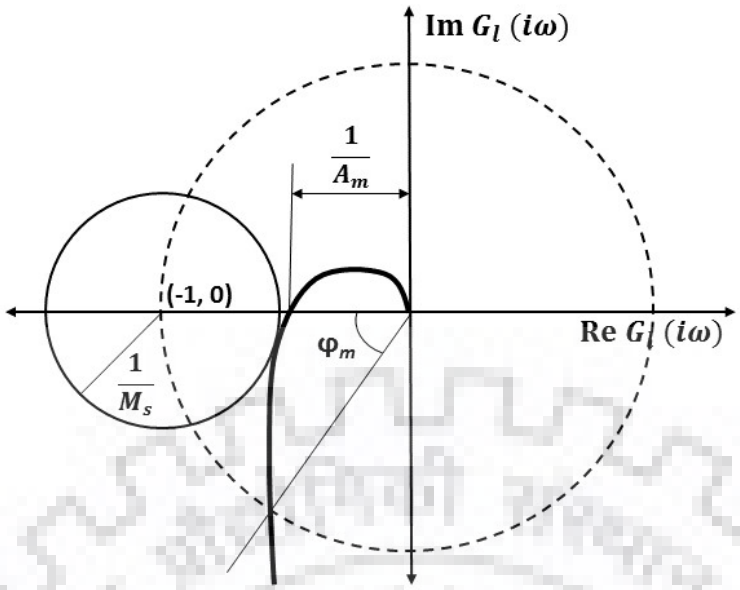


Fig. 6.5: Illustration of maximum sensitivity function M_s, phase margin ϕ_m, and gain margin A_m.

trol system design, the optimization problem is defined as multi-objective problem [146]. The MOEAs provide better approach to solve such problems and give a set of Pareto-optimal solutions. In this proposed work, minimization of *IAE* for set point and the load disturbance step response are considered as two conflicting objective functions, [149].

The objective function when only set-point step response is considered, is given by

$$Obj\_fun_1(x_i) = IAE_{sp} = \int_0^\infty \mid e(t) \mid dt, \quad d = 0 \tag{6.15}$$

The objective function when only load disturbance step response is considered, is given by

$$Obj\_fun_2(x_i) = IAE_{ld} = \int_0^\infty \mid e(t) \mid dt, \quad r = 0 \tag{6.16}$$

here, $i = 1, 2, \ldots, N$ represents the number of parameters that are to be optimized.

The robustness is a primary concern in controller design problem to avoid uncertainties created by system parameter variations. The maximum sensitivity function M_s is used to ensure the robustness of the system. The maximum sensitivity function is given by

$$M_s = \max_{\omega \in (0,+\infty)} \left| \frac{1}{1 + C(s)G(s)} \right|_{s=j\omega} \tag{6.17}$$

The illustration of sensitivity function and stability is demonstrated in Fig. 6.5. We can observe that M_s value is equivalent to the reciprocal of the smallest distance of the Nyquist plot from point (-1, 0). Therefore, this value indicates the system robustness in terms of how far the system is from verge to instability. For appropriate relative stability this function is bounded between 1.4 and 2.0 [3]. In case of 2-DOF controllers extra parameters are included to differentiate refer-

Table 6.1: Optimal parameters for different controllers.

Parameters	PID	FOPID	2-DOF-PID	2-DOF-FOPID
K_p	46.5892	49.8074	15.0251	16.4258
K_i	199.54	198.47	199.94	199.95
K_d	0.2489	0.7928	0.2672	0.3027
μ	-	0.9975	-	0.9112
λ	-	0.8076	-	0.9937
α	-	-	0.7461	0.9553
β	-	-	0.4198	0.3720
N	-	-	33.6925	64.9509

ence input from output signal, therefore the closed loop transfer function of 1-DOF and 2-DOF controllers are different. It is important to note here that, both the 1-DOF and 2-DOF controllers model to same open loop transfer function $C(s)G(s)$ (after breaking the loop at either side of system $G(S)$) [112]. Therefore, the same value of maximum sensitivity function M_s is considered in both the controllers with similar set of variables. It is claimed in [112] that the 1-DOF controller exhibits more overshoots and poor response due to extra zeros in transfer function. These zeros are expelled by 2-DOF structure to show better results.

For practical implementation, NSGA-II is applied in a constrained optimization mode and penalty function constraint handling technique is used. If the value of M_s goes beyond the specified range, it will create a penalty equivalent to its depth of violation. The NSGA-II algorithm will superintend the solutions towards Pareto-optimal front with satisfactory constraints. In the final stage, most reasonable solution is to be find out from obtained acceptable Pareto optimal solution set. Although various methods are suggested for this decision making, the Nash solution criteria [150] is used in this work for its ease of implementation.

6.5 Simulation Results and Discussion

In this section, performance analysis of 2-DOF-FOPID controller is illustrated with PID, FOPID and 2-DIF-PID controllers. Multi-objective NSGA-II algorithm with constraint handling approach is applied to these controllers for 50 iterations and 50 population size for 10 runs each. Among them the best results are demonstrated in this section. All the simulations and results are obtained in MATLAB version R2014b. The ranges for all the designed variables are kept same for each of the controllers and other parameters are set as: Mutation probability = 0.01, Crossover Probability = 0.9. The numbers of variables optimized for PID, FOPID, 2-DOF-PID, and 2-DOF FOPID controllers are 3, 5, 6, and 8 respectively. Considering the structure of MLS and ball

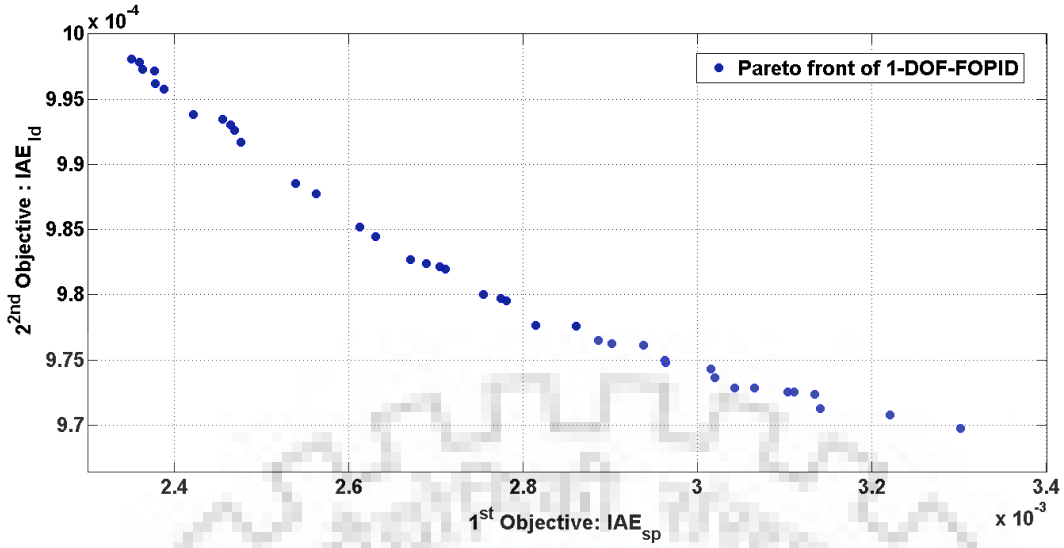


Fig. 6.6: Pareto-optimal front obtained by MOO for 1-DOF-FOPID controller applied to MLS.

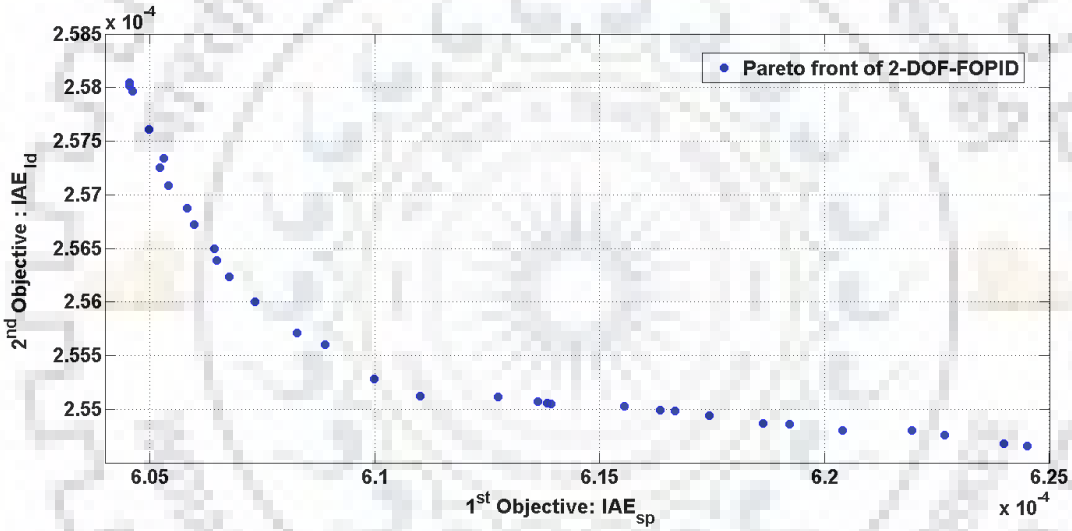


Fig. 6.7: Pareto-optimal front obtained by MOO for 2-DOF-FOPID controller applied to MLS.

position, the system is applied with step input of $r = 0.2$ amplitude. As represented in (6.15) and (6.16), minimization of integrated absolute error for set point and disturbance are considered as objective functions, however, the fitness of function and its feasibility is checked by the value of M_s in (6.17). The optimal controller parameters obtained for different controllers are listed in Table 6.1. All the results in this work are generated using this set of parameter values. The set of optimal solutions are obtained after completion of final iteration. After completion of final iteration, the set of best solutions are obtained as Pareto optimal fronts. Figs. 6.6 and 6.7 demonstrate the Pareto-optimal fronts obtained by these sets of optimal solutions for 1-DOF-FOPID, and 2-DOF-FOPID controllers, respectively. It can be seen that, in case of 2-DOF-FOPID controller the Pareto front obtained for lower values of both IAE_{sp} and IAE_{ld} than the FOPID controller and others. As many competitive solutions form a Pareto-optimal front, therefore, NASH solution approach is applied for finding single most prominent solution in this work.

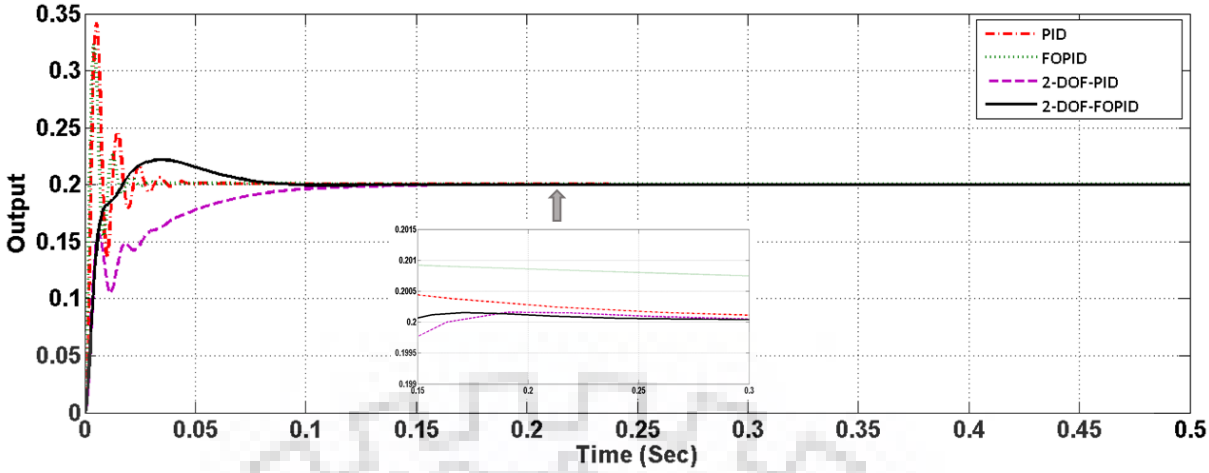


Fig. 6.8: Comparative illustration of step response.

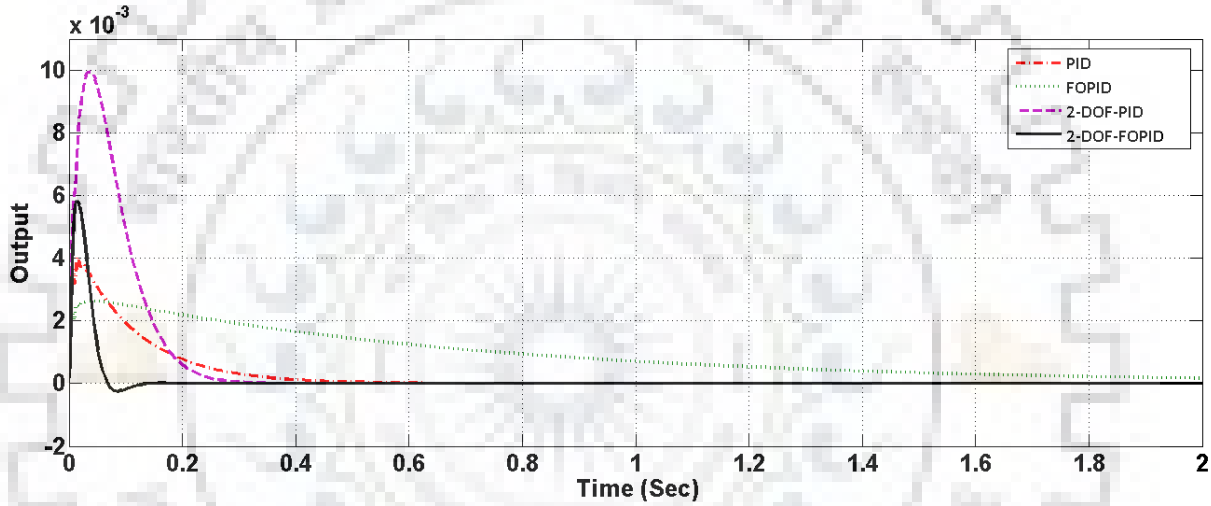


Fig. 6.9: Comparative illustration of step disturbance response.

The final values of *IAE* are listed in Table 6.2. Various experimental analysis are carried out and the values of *IAE* are reported for set point input ($r = 0.2$ at $t = 0$ s), load disturbance input ($d = 0.1$ at $t = 0$ s) and both ($r = 0.2$ at $t = 0$ s and $d = -0.1$ at $t = 1$ s) for different type of controllers.

The step response for the different controllers applied to MLS is shown in Fig. 6.8. From this comparative illustration, it can be observed that the 2-DOF-FOPID controller has low settling time with minimum overshoot and smoother response. All the other controllers execute oscillatory or overdamped behaviour. As MOO optimization is applied with the aim to minimize *IAE* for both set point and disturbance, the optimal tuning of parameters gives the desired results.

To evaluate the disturbance rejection of controllers, only disturbance signal $d = 0.1$ at $t = 0$ is applied to the controllers. From Fig. 6.9 and Table 6.2 we can conclude that the proposed 2-DOF-FOPID controller exhibits better disturbance rejection.

To witness the performance of 2-DOF-FOPID controller applied to MLS system for variable set point tracking, the response of system is presented in Fig. 6.10. Here, the system is applied

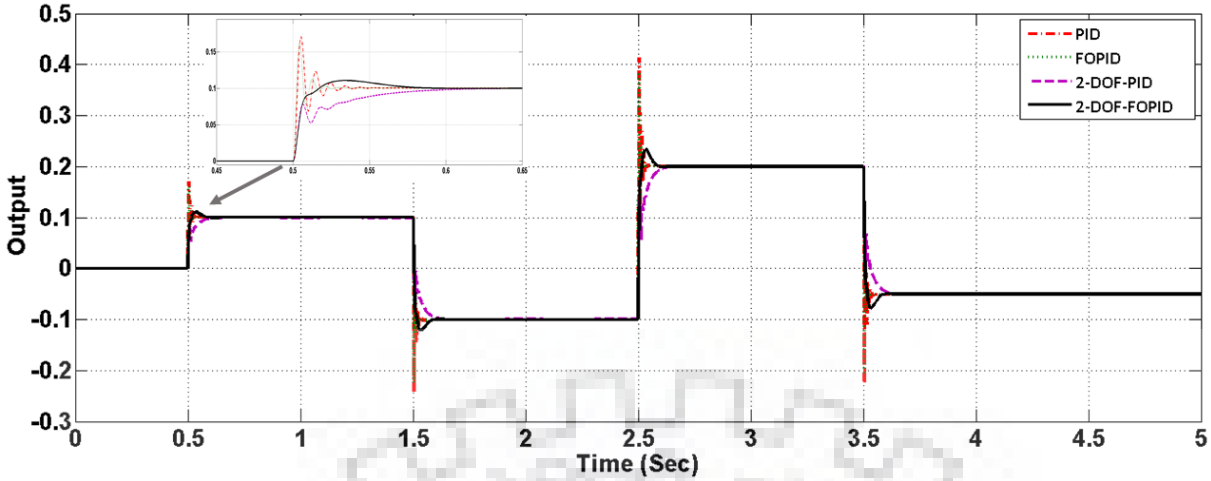


Fig. 6.10: Comparative illustration of response for variable input signal.

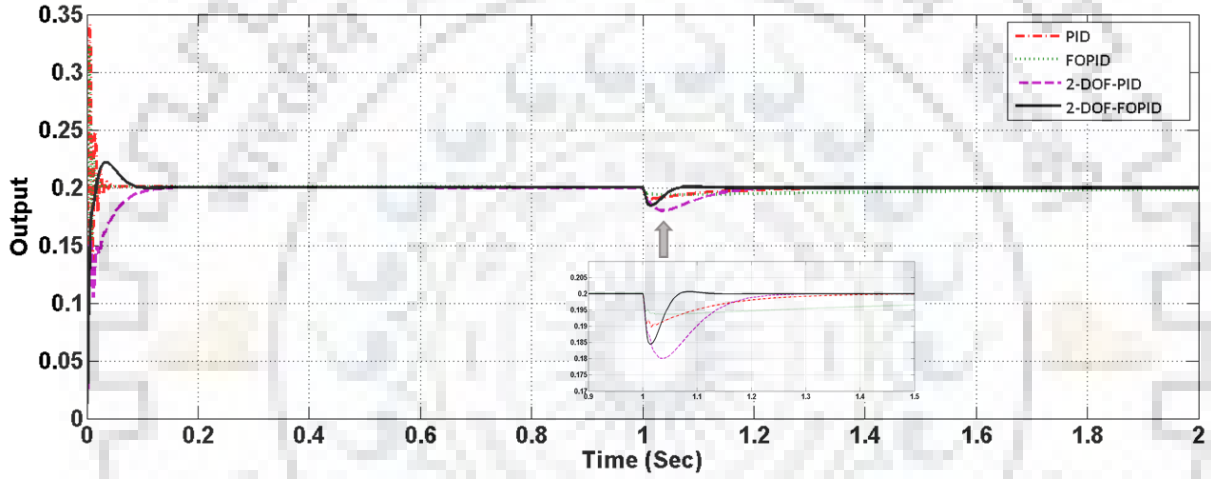


Fig. 6.11: Comparative illustration of step response for disturbance $d = -0.1$ at $t = 1$ s.

through variable set point signal and the response is obtained with same optimal parameters. The zoomed view at the signal variation shows that 2-DOF-FOPID controller exhibits lower oscillatory behavior and settles at set point faster than other controllers. It can be observed that, PID controller tracks the set point signal well but exhibits a steady-state error. It can be investigated form the overall values that the 2-DOF-FOPID controller outperforms other and shows more robust and effective behavior.

The comparative illustration of step response for disturbance $d = -0.1$ at $t = 1s$ is presented in Fig. 6.11. Here, it is seen that, the 2-DOF-FOPID controller provides better external disturbance rejection along with set point following.

Table 6.2: *IAE* values of controllers.

Input mode	Set point	Load disturbance	Step + Disturbance
PID	8.55×10^{-3}	9.1×10^{-4}	2.82×10^{-3}
FOPID	2.35×10^{-3}	9.7×10^{-4}	3.33×10^{-3}
2-DOF-PID	3.55×10^{-3}	5.1×10^{-4}	2.78×10^{-3}
2-DOF-FOPID	$\mathbf{6.04 \times 10^{-4}}$	$\mathbf{2.55 \times 10^{-4}}$	$\mathbf{1.34 \times 10^{-3}}$

6.6 Concluding Remarks

The chapter presents multi-objective robust design procedure of 2-DOF-FOPID controller and investigates its performance for MLS. The sensitivity function is used as a constraint to ensure robust design and minimization of *IAE* for set-point and disturbance are defined as two conflictive objective functions. It is evinced from the results that 2-DOF-FOPID controller exhibits prominent set point tracking and external disturbance rejection. It can be observed from the illustrations and tables that the extra flexibility in parameter tuning is provided through incorporation of 2-DOF structure with fractional order operators. Consequentially, robustness towards the external disturbances and desired set point tracking are improved. It can be concluded from the results that the 2-DOF-FOPID controllers are more robust, fast and effective to eliminate steady-state error. Furthermore, this work also manifests the applicability of MOO methodology for tuning of controller parameters for multi-variable and complex control systems.

Chapter 7

Conclusion and Future Scope

Previous chapters have reported the entire work carried out in this thesis. This chapter finally summarizes the conclusions that are drawn from comprehensive result analysis and observations. Also, it recommends the directions for future applications and studies.

7.1 Conclusion

The work presented in this thesis mainly focuses on the enhancement of various soft computing techniques, such as metaheuristic algorithm and type-2 fuzzy logic system, and incorporate them to design high performance intelligent controllers. An effort has been made to propose modifications and new developments in different soft computing techniques and apply them for controller design problems. The controllers are implemented on different complex systems and comprehensive experimental analysis is carried out.

The following conclusions are drawn from the overall work in this thesis:

- As a preliminary objective, a hybrid GWO-ABC algorithm is proposed by hybridizing conventional GWO algorithm with information sharing property of employed bees in ABC algorithm to comprehend benefits of both the algorithms. The new population initialization scheme, based on chaotic mapping and OBL techniques, is employed to give a better start with fitter initial candidate solutions. The amalgamation of these strategies overcome the shortcomings of the conventional GWO algorithm by improving exploration capability and convergence rate. It also reduces the chances of entrapment at local optima. The performance of the GWO-ABC algorithm is substantiated through exploitation and exploration analysis, convergence rate analysis, and non-parametric Wilcoxon rank-sum test over well established test bed. The statistical analysis and convergence curves reveal the outstanding performance of proposed algorithm over other state-of-the-art algorithms. It is exceptionally noted that no additional function evaluations are required in proposed GWO-ABC, hence, the algorithm is enhanced with similar computational complexity of GWO. Overall,

it is observed that the proposed GWO-ABC algorithm manifests improved exploitation and exploration tendencies with maintaining the prime features of original GWO.

- The tuning of controllers is considered as a high-dimensional, complex, multimodal numerical optimization problem, as many locally optimal solutions can be obtained for different combinations of the parameter values. The proposed GWO-ABC algorithm is employed to solve distinct controller tuning problems. A variety of linear and non-linear systems, with and without delay, are applied with FOPID controller and the parameters are optimized to get optimum performance index. From experimental analysis, it is observed that proposed GWO-ABC tuned FOPID controllers outperform the controllers tuned by other well established algorithms in terms of minimum percent overshoot ($M_p\%$), lower rise time (t_r), and achieve null steady-state error (E_{ss}) in faster settling time (t_s). Thus, it is manifested that the proposed GWO-ABC algorithm succeeded as a better option for multimodal controller tuning problems of complex control systems.

- Further, the cooperative foraging strategy of animals is incorporated in the conventional GWO to develop CFGWO algorithm. Two communication signalling schemes, based on chaotic mapping and OBL strategy, are amended to model the distinct signalling behaviours of wolves. These schemes ameliorate global search ability of all individual search-agents by bestowing them opportunity to share information with other search-agents and enhance the exploration ability. The gradually decreasing acceleration coefficient balances the exploration and exploitation behaviours throughout the iterations. The proposed CFGWO algorithm is used to design the fuzzy-based FOPID controller for the trajectory tracking problem of 2-link robotic manipulator. The trajectory tracking plots clarify the superiority of CFGWO algorithm over the conventional GWO and its recent upgraded versions like GWO-ABC and LGWO. It can be clearly perceived from the plots that the CFGWO obtained accurate results demonstrating precise overlapping on desired trajectories and optimized the controller performance with low position errors and effective controller output. Ultimately, it can be inferred that the proposed modifications recommended in CFGWO ameliorate the conventional GWO to enhance the overall performance in controller design applications for complex systems.

- An efficient IT2FP-PID controller for 2-link robotic manipulator is proposed with a systematic strategy for optimizing the controller parameters and MF structures. The incorporation of IT2-FLC based precompensator in the proposed controller eradicates the overshoots and undershoots in system response. The FLC based precompensator exhibits exceptional agreement between the IT2-FLC and precompensator and regulates the control signal to enhance the performance when the system has unknown uncertainties. Various experimental analysis provide evidences of exact trajectory tracking of 2-link robotic manipulator with minimum position errors by the proposed IT2FP-PID controller. The exhaustive robustness analysis in presence of distinct non-linear dynamics evinces the superiority of IT2FP-PID

controller in terms of robustness for (i) payload variations, (ii) model uncertainties, (iii) disturbance in signals, and (iv) random noise at feedback path. The overall illustrations and results confirm that optimized structures of antecedent MF utilized benefits of FOU to restrain various uncertainties in the input signals. Even though the optimized shapes of T1-MF are also applied through tuning of MF parameters for T1FP-PID controller, it could not handle the uncertainty in equally efficient manner due to absence of fuzziness in T1-MF. As inferred, the performance of classical PID and FPID controllers with uniform MF are rather disappointing and found unsuited for the complex systems with uncertainties. To sum up, it can be claimed that (a) additional tuning parameters provide extra degree of freedom to get better performance in optimal controller design,(b) in case of IT2-FLC, the systematic strategy to optimize the shapes of MF derive maximum benefits of FOU to handle uncertainty (c) the proposed IT2FP-PID controller revealed as viable alternative to control complex non-linear systems with high uncertainties, (d) the proposed GWO-ABC algorithm can efficiently solve the low-dimensional and high-dimensional constrained optimization problems.

- The last chapter presented multi-objective robust design procedure of 2-DOF-FOPID controller and investigated its performance for MLS. The sensitivity function is used as a constraint to ensure robust design and minimization of IAE for set-point and disturbance are defined as two conflicting objective functions. The controller parameters are tuned using multi-objective NSGA-II algorithm with constraint handling methodology. The major robustness investigations are carried out by applying variable input and external disturbance and observed that the 2-DOF-FOPID controller exhibits prominent set point tracking and external disturbance rejection. Also, it can be concluded from the results that the 2-DOF-FOPID controller is more robust, fast, and effective to eliminate steady-state error. In most of the cases, the 2-DOF-FOPID controller exhibits lower oscillatory behavior and settles at set point faster than other controllers. Finally, this work justifies the applicability of MOO methodology for tuning of controller parameters for multi-variable and complex control systems design problems.

7.2 Future Scope

Research and developments in various soft computing techniques always provide better prospects for designing intelligent systems. The researchers from different fields are motivated to design and implement various soft computing techniques to broad range of applications in engineering and science fields.

With regards to the overall work presented in this thesis, following suggestions are identified for future developments.

- The proposed GWO-ABC algorithm can be validated for interdisciplinary and multi-domain complex engineering and science optimization problems. It can also be utilized for high-dimensional, constrained, optimization problems.
- Further advance strategies can be suggested using chaos theory, random walk, levy flight, Cauchy operator, and OBL based methods to design improved optimization algorithms. The fuzzy logic based adoptive parameter tuning approach can be incorporated to balance the exploration and exploitation behaviour.
- The IT2-FLC overcomes the limitations of T1-FLCs in highly uncertain environments. As most of the real-world applications are influenced with high uncertainties, there is great scope to investigate the proposed IT2-FLC based control schemes for real-world applications. Experimental investigations can be carried out to estimate the efficacy of IT2FP-PID for controlling other complex engineering problems with higher uncertainties.
- The hardware design, development, and implementation of proposed IT2FP-PID provides an interesting opportunity for further research. The control schemes can be developed and implemented through real-time hardware applications in robotics and process industry.
- In theoretical research approach, a new type-reduction technique can be developed to reduce the computation time of IT2-FLS. Analytical methods can be developed to study the relationship between robustness and FOU.
- The MOO approach has marked superiority over SOO to solve the optimization problem with conflicting objectives. Hence, the MOO approach for control system design problems with different constraints can be developed.
- The multi-objective algorithm based on CFGWO or GWO-ABC can be designed and applied to solve different optimization problems.
- In nutshell, the results reported in this thesis outlined a research direction for application of enhanced soft computing techniques to improve the performance of control system design required for complex systems.

References

[1] G. Goodwin, S. Graebe, and M. Salgado, *Control System Design*. NJ, Englewood Cliffs:Prentice-Hall, 2001.

[2] A. Zilouchian and M. Jamshidi, *Intelligent Control Systems Using Soft Computing Methodologies*. Taylor and Francis CRC Press, 2001.

[3] K. J. Astrom, *PID Controllers: Theory, Design and Tuning*. Instrument Society of America, 1995.

[4] O. Castillo, L. Amador-Angulo, J. R. Castro, and M. Garcia-Valdez, "A comparative study of type-1 fuzzy logic systems, interval type-2 fuzzy logic systems and generalized type-2 fuzzy logic systems in control problems," *Information Sciences*, vol. 354, pp. 257–274, Aug 2016.

[5] O. Castillo and P. Melin, " A review on interval type-2 fuzzy logic applications in intelligent control," *Information Sciences*, vol. 279, pp. 615–631, Sep 2014.

[6] C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller- Part I," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 20, no. 2, pp. 404–418, Mar 1990.

[7] R. K. Mudi and N. R. Pal, "A robust self-tuning scheme for PI- and PD-type fuzzy controllers," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 1, pp. 2–16, Feb 1999.

[8] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multi-objective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr 2002.

[9] P. Gaidhane and M. Nigam, "Multi-objective robust design and performance analysis of two-DOF-FOPID controller for magnetic levitation system," in *Proceedings of 14th India Council International Conference (INDICON)*, Dec 2017, pp. 1–6.

[10] S. Das, S. Saha, S. Das, and A. A Gupta, "On the selection of tuning methodology of FOPID controllers for the control of higher order processes," *ISA Transactions*, vol. 50, no. 3, pp. 376–388, Jul 2011.

[11] P. J. Gaidhane and M. J. Nigam, " A hybrid grey wolf optimizer and artificial bee colony algorithm for enhancing the performance of complex systems," *Journal of Computational Science*, vol. 27, pp. 284–302, Jul 2018.

[12] L. A. Zadeh, "Fuzzy logic, neural networks, and soft computing," *Communications of the ACM*, vol. 37, no. 3, pp. 77–84, Mar 1994.

[13] S. R. Kolla and S. D. Altman, "Artificial neural network based fault identification scheme implementation for a three-phase induction motor," *ISA Transactions*, vol. 46, no. 2, pp. 261–266, Apr 2007.

[14] K. Kumar, N. Sen, S. Azid, and U. Mehta, "A fuzzy decision in smart fire and home security system," *Procedia Computer Science*, vol. 105, pp. 93–98, 2017.

[15] S. Kolla, "Applying fuzzy logic to power system protective relays," *InTech*, vol. 44, no. 6, Jun 1997.

[16] A. Savran and G. Kahraman, "A fuzzy model based adaptive PID controller design for non-linear and uncertain processes," *ISA Transactions*, vol. 53, no. 2, pp. 280–288, Mar 2014.

[17] J. O. Pedro, M. Dangor, O. A. Dahunsi, and M. M. Ali, "Differential evolution-based PID control of non-linear full-car electrohydraulic suspensions," *Mathematical Problems in Engineering*, vol. 2013, pp. 1–13, 2013.

[18] P. J. Gaidhane, A. Kumar, and M. Nigam, "Tuning of two-DOF-FOPID controller for magnetic levitation system : A multi-objective optimization approach," in *Proceeding of 6th International Conference on Computer Applications in Electrical Engineering Recent Advances (CERA)*, Oct 2017, pp. 479–484.

[19] A. K. Awasthi, O. P. Dubey, S. Sharma, and A. Awasthi, "A fuzzy logic model for estimation of groundwater recharge," in *Proceedings of Annual Meeting of the North American Fuzzy Information Processing Society Conference (NAFIPS)*, Jun 2005, pp. 809–813.

[20] R. Sharma, K. P. S. Rana, and V. Kumar, "Performance analysis of fractional order fuzzy PID controllers applied to a robotic manipulator," *Expert Systems with Applications*, vol. 41, no. 9, pp. 4274–4289, Jul 2014.

[21] J. P. Gawande, A. D. Rahulkar, and R. S. Holambe, "A new approach to design triplet half-band filter banks based on balanced-uncertainty optimization," *Digital Signal Processing*, vol. 56, pp. 123–131, Sep 2016.

[22] S. Das, I. Pan, and S. Das, "Performance comparison of optimal fractional order hybrid fuzzy PID controllers for handling oscillatory fractional order processes with dead time," *ISA Transactions*, vol. 52, no. 4, pp. 550–566, Jul 2013.

[23] A. Kumar and V. Kumar, " Evolving an interval type-2 fuzzy PID controller for the redundant robotic manipulator," *Expert Systems with Applications*, vol. 73, pp. 161–77, Mar 2017.

[24] ——, "Hybridized ABC-GA optimized fractional order fuzzy pre-compensated FOPID control design for 2-DOF robot manipulator," *AEU - International Journal of Electronics and Communications*, vol. 79, pp. 219–233, Sep 2017.

[25] T. Kumbasar and H. Hagras, *Interval type-2 fuzzy PID controllers*. Springer Berlin Heidelberg, Jan 2015, pp. 285–294.

[26] A. Kumar, P. J. Gaidhane, and V. Kumar, "A non-linear fractional order PID controller applied to redundant robot manipulator," in *Proceeding of 6th International Conference on Computer Applications in Electrical Engineering Recent Advances (CERA)*, Oct 2017, pp. 527–532.

[27] H. M. Fayek, I. Elamvazuthi, N. Perumal, and B. Venkatesh, "A controller based on optimal type-2 fuzzy logic: Systematic design, optimization and real-time implementation," *ISA Transactions*, vol. 53, no. 5, pp. 1583–1591, Sep 2014.

[28] T. Ganesan, P. Vasant, and I. Elamvazuthi, *Advances in Metaheuristics Applications in Engineering Systems*. CRC Press Taylor & Francis Group, 2017.

[29] N. N. Karnik, J. M. Mendel, and Q. Liang, "Type-2 fuzzy logic systems," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 6, pp. 643–658, Dec 1999.

[30] X. S. Yang, *Nature-Inspired Metaheuristic Algorithms*. Luniver Press, 2008.

[31] K. Shakti, P. Bhalla, and S. Sharma, "Automatic fuzzy rule base generation for intersystem handover using ant colony optimization algorithm," in *Proceedings of International Conference on Intelligent Systems and Networks (IISN-2007)*, Jan 2007, pp. 764–773.

[32] S. Sharma and K. Ogunlana, "Using genetic algorithm and neural network for modeling learning behavior in a multi agent system during emergency evacuation," *International Journal of Computers and their Applications*, vol. 22, pp. 172–182, Dec 2015.

[33] S. K. Dinkar and K. Deep, "Opposition based laplacian ant lion optimizer," *Journal of Computational Science*, vol. 23, pp. 71–90, Nov 2017.

[34] I. Fister, X. S. Yang, J. Brest, and Fister, "A brief review of nature-inspired algorithms for optimization," *Electrotechnical Review*, vol. 80, no. 3, pp. 116–122, Jul 2013.

[35] A. Gotmare, R. Patidar, and N. V. George, "Nonlinear system identification using a cuckoo search optimized adaptive Hammerstein model," *Expert Systems with Applications*, vol. 42, no. 5, pp. 2538–2546, Apr 2015.

[36] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, Apr 1997.

[37] R. Goel and R. Maini, "A hybrid of ant colony and firefly algorithms (HAFA) for solving vehicle routing problems," *Journal of Computational Science*, vol. 25, pp. 28–37, Mar 2018.

[38] S. Gupta and K. Deep, "A novel random walk grey wolf optimizer," *Swarm and Evolutionary Computation*, vol. 44, pp. 101–112, Jan 2019.

[39] M. M. Ali and P. Kaelo, "Improved particle swarm algorithms for global optimization," *Applied Mathematics and Computation*, vol. 196, no. 2, pp. 578–593, Mar 2008.

[40] J. Chen, W. Zhu, and M. M. Ali, "A hybrid simulated annealing algorithm for non-slicing VLSI floorplanning," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 41, no. 4, pp. 544–553, Jul 2011.

[41] N. Kanwar, N. Gupta, K. R. Niazi, A. Swarnkar, and R. C. Bansal, "Simultaneous allocation of distributed energy resource using improved particle swarm optimization," *Applied Energy*, vol. 185, no. 2, pp. 1684–1693, Jan 2017.

[42] S. S. Das, I. Pan, S. S. Das, and A. Gupta, " A novel fractional order fuzzy PID controller and its optimal time domain tuning based on integral performance indices," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 2, pp. 430–442, Mar 2012.

[43] O. P. Verma, G. Manik, and V. K. Jain, "Simulation and control of a complex non-linear dynamic behavior of multi-stage evaporator using PID and Fuzzy-PID controllers," *Journal of Computational Science*, vol. 25, pp. 238 – 251, Mar 2018.

[44] G. Q. Zeng, J. Chen, Y. X. Dai, L. M. Li, C. W. Zheng, and M. R. Chen, "Design of fractional order PID controller for automatic regulator voltage system based on multiobjective extremal optimization," *Neurocomputing*, vol. 160, pp. 173–184, Jul 2015.

[45] U. Mehta and S. Majhi, "On-line identification and control methods for PID controllers," in *Proceedings of 11th International Conference on Control Automation Robotics and Vision*. IEEE, Dec 2010, pp. 1507–1511.

[46] A. Ali and S. Majhi, "PI/PID controller design based on IMC and percentage overshoot specification to controller setpoint change," *ISA Transactions*, vol. 48, no. 1, pp. 10–15, Jan 2009.

[47] M. Ajmeri and A. Ali, "Two degree of freedom control scheme for unstable processes with small time delay," *ISA Transactions*, vol. 56, pp. 308–326, May 2015.

[48] U. Mehta and S. Majhi, "On-line relay test for automatic tuning of PI controllers for stable processes," *Transactions of the Institute of Measurement and Control*, vol. 34, no. 7, pp. 903–913, Nov 2012.

[49] A. Swarup, "Intelligent automatic generation control of two area interconnected power system using hybrid neuro fuzzy controller," *International Journal of Electrical and Computer Engineering*, pp. 1938–1943, 2011.

[50] S. K. Verma, S. Yadav, and S. K. Nagar, "Optimization of fractional order PID controller using grey wolf optimizer," *Journal of Control, Automation and Electrical Systems*, vol. 28, no. 3, pp. 314–322, Jun 2017.

[51] A. Goel and A. Swarup, "Adaptive fuzzy high-order super-twisting sliding mode controller for uncertain robotic manipulator," *Journal of Intelligent Systems*, vol. 26, no. 4, pp. 697–715, Mar 2016.

[52] G. L. Raja and A. Ali, "Modified parallel cascade control strategy for stable, unstable and integrating processes," *ISA Transactions*, vol. 65, pp. 394–406, Nov 2016.

[53] N. V. George and G. Panda, "A robust evolutionary feedforward active noise control system using Wilcoxon norm and particle swarm optimization algorithm," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7574–7580, Jun 2012.

[54] S. Kolla and L. Varatharasa, "Identifying three-phase induction motor faults using artificial neural networks," *ISA Transactions*, vol. 39, no. 4, pp. 433–439, Sep 2000.

[55] M. J. Er and Y. L. Sun, "Hybrid fuzzy proportional-integral plus conventional derivative control of linear and non-linear systems," *IEEE Transactions on Industrial Electronics*, vol. 48, no. 6, pp. 1109–1117, Dec 2001.

[56] J. L. Meza, V. Santibáñez, R. Soto, and M. A. Llama, "Fuzzy self-tuning PID semiglobal regulator for robot manipulators," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 6, pp. 2709–2717, Jun 2012.

[57] S. Alavandar, T. Jain, and M. J. Nigam, "Bacterial foraging optimized hybrid fuzzy precompensated PD control of two link rigid-flexible manipulator," *International Journal of Computational Intelligence Systems*, vol. 2, no. 1, pp. 51–59, 2009.

[58] I. Pan and S. Das, "Fractional order fuzzy control of hybrid power system with renewable generation using chaotic PSO," *ISA Transactions*, vol. 62, pp. 19–29, May 2016.

[59] A. Aggarwal, T. K. Rawat, and D. K. Upadhyay, "Design of optimal digital FIR filters using evolutionary and swarm optimization techniques," *AEU - International Journal of Electronics and Communications*, vol. 70, no. 4, pp. 373–385, Apr 2016.

[60] T. Kumbasar and H. Hagras, "Big Bang-Big Crunch optimization based interval type-2 fuzzy PID cascade controller design strategy," *Information Sciences*, vol. 282, pp. 277–295, Oct 2014.

[61] Y. Arya and N. Kumar, "BFOA-scaled fractional order fuzzy PID controller applied to AGC of multi- area multi-source electric power generating systems," *Swarm and Evolutionary Computation*, vol. 32, pp. 202–218, Feb 2017.

[62] Y. H. Fatihu, Hamza Mand Jen and C. I. Ahmed, "Cuckoo search algorithm based design of interval type-2 fuzzy PID controller for furuta pendulum system," *Engineering Applications of Artificial Intelligence*, vol. 62, pp. 134–151, Jun 2017.

[63] A. Ateş and C. Yeroglu, "Optimal fractional order PID design via Tabu search based algorithm," *ISA Transactions*, vol. 60, pp. 109–118, Jan 2016.

[64] I. Pan, S. Das, and A. Gupta, "Tuning of an optimal fuzzy PID controller with stochastic algorithms for networked control systems with random time delay," *ISA Transactions*, vol. 50, no. 1, pp. 28–36, Jan 2011.

[65] J. M. Mendel, R. I. John, and F. Liu, "Interval type-2 fuzzy logic systems made simple," *IEEE Transactions on Fuzzy Systems*, vol. 14, no. 6, pp. 808–21, Dec 2006.

[66] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning-I," *Information Sciences*, vol. 8, no. 3, pp. 199–249, 1975.

[67] N. N. Karnik and J. M. Mendel, "Introduction to type-2 fuzzy logic systems," in *Proceedings of International Conference on Fuzzy Systems* , vol. 2, May 1998, pp. 915–920.

[68] J. M. Mendel, *Uncertain Rule-based Fuzzy Logic Systems: Introduction and New Directions*. Springer International Publishing, 2017.

[69] O. Castillo, L. Amador-Angulo, J. R. Castro, and M. Garcia-Valdez, "A comparative study of type-1 fuzzy logic systems, interval type-2 fuzzy logic systems and generalized type-2 fuzzy logic systems in control problems," *Information Sciences*, vol. 354, pp. 257 – 274, Aug 2016.

[70] J. H. Kim, K. C. Kim, and E. K. P. Chong, "Fuzzy precompensated PID controllers," *IEEE Transactions on Control Systems and Technology*, vol. 2, no. 4, pp. 406–411, Dec 1994.

[71] J. Kim, J. Park, S. Lee, and E. K. P. Chong, "Fuzzy precompensation of PD controllers for systems with deadzones," *Journal of Intelligent & Fuzzy Systems*, vol. 1, no. 2, pp. 125–133, 1993.

[72] A. Taskin and T. Kumbasar, "An open source Matlab / Simulink Toolbox for interval type-2 fuzzy logic systems," in *Proceedings of 2015 Symposium Series on Computational Intelligence*, Dec 2015, pp. 1561–1568.

[73] D. Wu and J. M. Mendel, "Enhanced Karnik – Mendel algorithms," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 4, pp. 923–934, Aug 2009.

[74] H. Hagras, "A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots," *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 4, pp. 524–539, Aug 2004.

[75] J. M. Mendel, H. Hagras, and R. John, "Standard background material about interval type-2 fuzzy logic systems that can be used by all authors," pp. 1–11, 2010.

[76] A. Kumar and V. Kumar, "Performance analysis of optimal hybrid novel interval type-2 fractional order fuzzy logic controllers for fractional order systems," *Expert Systems with Applications*, vol. 93, pp. 435–455, Mar 2018.

[77] S. Debbarma, L. Chandra, and N. Sinha, "Electrical power and energy systems automatic generation control using two degree of freedom fractional order PID controller," *International Journal of Electrical Power & Energy Systems*, vol. 58, pp. 120–129, Jan 2014.

[78] M. Li, P. Zhou, Z. Zhao, and J. Zhang, "Two-degree-of-freedom fractional order-PID controllers design for fractional order processes with dead-time," *ISA Transactions*, vol. 61, pp. 147–154, Mar 2016.

[79] A. Dumlu and K. Erenturk, "Trajectory tracking control for a 3-DOF parallel manipulator using fractional-order $PI^{\lambda}D^{\mu}$ control," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 7, pp. 3417–3426, Jul 2014.

[80] R. Sharma, P. Gaur, and A. P. Mittal, "Performance analysis of two-degree of freedom fractional order PID controllers for robotic manipulator with payload," *ISA Transactions*, vol. 58, pp. 279–291, Sep 2015.

[81] Y. Tang, M. Cui, C. Hua, L. Yang, and Y. Li, " Optimum design of fractional order $PI^{\lambda}D^{\mu}$ controller for AVR system using chaotic ant swarm," *Expert Systems with Applications*, vol. 39, pp. 6887–6896, Feb 2012.

[82] I. Pan and S. Das, " Chaotic multi-objective optimization based design of fractional order PILDM controller in AVR System," *International Journal of Electrical Power & Energy Systems*, vol. 43, no. 1, pp. 393–407, Dec 2012.

[83] R. Sharma, P. Gaur, and A. P. Mittal, "Design of two-layered fractional order fuzzy logic controllers applied to robotic manipulator with variable payload," *Applied Soft Computing*, vol. 47, pp. 565–76, Oct 2016.

[84] A. Oustaloup, F. Levron, B. Mathieu, and F. M. Nanot, "Frequency-band complex non-integer differentiator: Characterization and synthesis," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 47, no. 1, pp. 25–39, Jan 2000.

[85] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. Cambridge, MA, USA: MIT Press, 1992.

[86] R. Storn and K. Price, "Differential evolution : A simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, Dec 1997.

[87] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, Nov 1995, pp. 1942–1948.

[88] N. Birla and A. Swarup, "PSO approach to preview tracking control systems," in *Proceedings of Region 10 Conference, (TENCON-2009)*, Jan 2009, pp. 1–6.

[89] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[90] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.

[91] S. Mirjalili, "The ant lion optimizer," *Advances in Engineering Software*, vol. 83, pp. 80–98, May 2015.

[92] D. Chitara, A. Swarnkar, N. Gupta, K. R. Niazi, and R. C. Bansal, "Optimal tuning of multimachine power system stabilizer using cuckoo search algorithm," *IFAC-PapersOnLine*, vol. 48, no. 30, pp. 143–148, 2015.

[93] K. S. Singh, I. Elamvazuthi, K. Z. Shaari, and F. V. Lima, "PID tuning control strategy using cuckoo search algorithm," in *Proceedings of Student Conference on Research and Development (SCOReD)*, Dec 2015, pp. 129–133.

[94] A. P. Patwardhan, R. Patidar, and N. V. George, "On a cuckoo search optimization approach towards feedback system identification," *Digital Signal Processing*, vol. 32, pp. 156–163, Sep 2014.

[95] A. Baykasoglu and F. B. Ozsoydan, "Adaptive firefly algorithm with chaos for mechanical design optimization problems," *Applied Soft Computing*, vol. 36, pp. 152–164, Nov 2015.

[96] A. L. aro Bolaji, M. A. Al-Betar, M. A. Awadallah, A. T. Khader, and L. M. Abualigah, "A comprehensive review: Krill herd algorithm (KH) and its applications," *Applied Soft Computing Journal*, vol. 49, pp. 437–446, Dec 2016.

[97] R. A. Formato, "Central force optimization: A new metaheuristic with applications in applied electromagnetics," *Progress In Electromagnetics Research*, vol. 77, pp. 425–491, 2007.

[98] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: A gravitational search algorithm," *Information Sciences*, vol. 179, no. 13, pp. 2232–2248, Jun 2009.

[99] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: Harmony search," *Simulation*, vol. 76, no. 2, pp. 60–68, Feb 2001.

[100] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems," *Computer-Aided Design*, vol. 43, no. 3, pp. 303–315, Mar 2011.

[101] A. Husseinzadeh Kashan, "League Championship Algorithm (LCA): An algorithm for global optimization inspired by sport championships," *Applied Soft Computing*, vol. 16, pp. 171–200, Mar 2014.

[102] J. Oliveira, P. M. Oliveira, J. Boaventura-Cunha, and T. Pinho, "Chaos-based grey wolf optimizer for higher order sliding mode position control of a robotic manipulator," *Nonlinear Dynamics*, vol. 90, no. 2, pp. 1353–1362, Oct 2017.

[103] H. Yu, Y. Yu, Y. Liu, Y. Wang, and S. Gao, "Chaotic grey wolf optimization," in *Proceedings of 2016 International Conference on Progress in Informatics and Computing (PIC)*, Dec 2016, pp. 103–113.

[104] G.-G. Wang, S. Deb, A. H. Gandomi, Z. Zhang, and A. H. Alavi, "Chaotic cuckoo search," *Soft Computing*, vol. 20, no. 9, pp. 3349–3362, Sep 2016.

[105] A. A. Heidari and P. Pahlavani, "An efficient modified grey wolf optimizer with Lévy flight for optimization tasks," *Applied Soft Computing Journal*, vol. 60, pp. 115–134, Nov 2017.

[106] M. Niu, Y. Wang, S. Sun, and Y. Li, "A novel hybrid decomposition-and-ensemble model based on CEEMD and GWO for short-term $PM_{2.5}$ concentration forecasting," *Atmospheric Environment*, vol. 134, pp. 168–180, Jun 2016.

[107] T. Jayabarathi, B. Raghunathan, T Adarsh, and P. N. Suganthan, "Economic dispatch using hybrid grey wolf optimizer," *Energy*, vol. 111, pp. 630–641, Sep 2016.

[108] M. A. Ahandani and H. Alavi-Rad, "Opposition-based learning in the shuffled differential evolution algorithm," *Soft Computing*, vol. 16, no. 8, pp. 1303–1337, Aug 2012.

[109] S. Rahnamayan, H. R. Tizhoosh, and M. M. Salama, "Opposition versus randomness in soft computing techniques," *Applied Soft Computing*, vol. 8, no. 2, pp. 906–918, Mar 2008.

[110] M. Spong and M. Vidyasagar, *Robot Dynamic and Control*, 1st ed. John Wiley & Sons, Inc., 1989.

[111] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, 2nd ed. Addison-Wesley Longman Publishing Company, 1989.

[112] A. Ghosh, T. Rakesh Krishnan, P. Tejaswy, A. Mandal, J. K. Pradhan, and S. Ranasingh, "Design and implementation of a 2-DOF PID compensation for magnetic levitation systems," *ISA Transactions*, vol. 53, no. 4, pp. 1216–1222, Jul 2014.

[113] S. Yadav, S. K. Verma, and S. K. Nagar, "Optimized PID controller for magnetic levitation system," *IFAC-PapersOnLine*, vol. 49, no. 1, pp. 778–782, 2016.

[114] S. Yadav, J. P. Tiwari, and S. K. Nagar, "Digital control of magnetic levitation system using fuzzy logic controller," *International Journal of Computer Applications*, vol. 41, no. 21, pp. 27–31, Mar 2012.

[115] C. Lin, M. Lin, and C. Chen, "SoPC-based adaptive PID control system design for magnetic levitation system," *IEEE Systems Journal*, vol. 5, no. 2, pp. 278–287, Jun 2011.

[116] Z. Yang, K. Kunitoshi, and S. Kanae, "Adaptive robust output-feedback control of a magnetic levitation system by K-filter approach," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 1, pp. 390–399, Jan 2008.

[117] G. M. Komaki and V. Kayvanfar, "Grey Wolf Optimizer algorithm for the two-stage assembly flow shop scheduling problem with release time," *Journal of Computational Science*, vol. 8, pp. 109–120, May 2015.

[118] N. Jayakumar, S. Subramanian, S. Ganesan, and E. Elanchezhian, "Grey wolf optimization for combined heat and power dispatch with co-generation systems," *International Journal of Electrical Power & Energy System*, vol. 74, pp. 252–264, Jan 2016.

[119] S. Medjahed, T. A. Saadi, A. Benyettou, and M. Ouali, "Gray Wolf Optimizer for hyperspectral band selection," *Applied Soft Computing*, vol. 40, pp. 178–186, Mar 2016.

[120] S. Mirjalili, S. Saremi, S. M. Mirjalili, and L. D. S. Coelho, "Multi-objective grey wolf optimizer: A novel algorithm for multi-criterion optimization," *Expert Systems with Appllications*, vol. 47, pp. 106–119, Apr 2016.

[121] X. Song, L. Tang, S. Zhao, X. Zhang, L. L. Huang, and J. W. Cai, "Grey Wolf Optimizer for parameter estimation in surface waves," *Soil Dynamics and Earthquake Engineering*, vol. 75, pp. 147–157, Aug 2015.

[122] M. A. Tawhid and A. F. Ali, "A hybrid grey wolf optimizer and genetic algorithm for minimizing potential energy function," *Memetic Computing*, vol. 9, no. 4, pp. 347–359, Dec 2017.

[123] A. Kishor and P. K. Singh, "Empirical study of grey wolf optimizer," in *Proceedings of Fifth International Conference on Soft Computing for Problem Solving*, Aug 2016, pp. 1037–1049.

[124] J. J. Liang, B. Y. Qu, and P. N. Suganthan, "Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization," *Technical Report 11, Computational Intelligence Laboratory, Zhengzhou University, Singapore.*, no. 34, pp. 281–295, Jan 2013.

[125] S. Mirjalili, "SCA: A sine cosine algorithm for solving optimization problems," *Knowledge-Based Systems*, vol. 96, pp. 120–133, Mar 2016.

[126] J. Derrac, S. Garca, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, Mar 2011.

[127] C. J. Torney, A. Berdahl, and I. D. Couzin, "Signalling and the evolution of cooperative foraging in dynamic environments," *PLOS Computational Biology*, vol. 7, no. 9, pp. 1–10, Sep 2011.

[128] I. Bailey, J. P. Myatt, and A. M. Wilson, "Group hunting within the carnivora: Physiological, cognitive and environmental influences on strategy and cooperation," *Behavioral Ecology and Sociobiology*, vol. 67, pp. 1–17, Jan 2013.

[129] J. Ruch, M. E. Herberstein, and J. M. Schneider, "Families hunt more successfully: Effect of group composition on hunting and communal feeding," *Animal Behaviour*, vol. 91, pp. 170–177, May 2014.

[130] R. Peterson and P. Ciucci, "The wolf as carnivore," *Wolves: Behavior, Ecology, and Conservation*, pp. 104–130, Jan 2003.

[131] C. W. Clark and M. Mangel, "Foraging and flocking strategies : Information in an uncertain environment," *The American Naturalist*, vol. 123, no. 5, pp. 626–641, May 1984.

[132] O. Castillo and P. Melin, *Genetic Optimization of Interval Type-2 Fuzzy Systems for Hardware Implementation on FPGAs*. Springer Berlin Heidelberg, 2012, pp. 63–84.

[133] D. Kim, "An Implementation of fuzzy logic controller on the reconfigurable FPGA system," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 3, pp. 703–715, Jun 2000.

[134] P. Das, P. J. Edavoor, S. Raveendran, S. Rathore, and A. D. Rahulkar, "Design and implementation of PID controller based on orthogonal wavelet filter-banks in FPGA," in *Proceedings of Seventh International Symposium on Embedded Computing and System Design (ISED)*, Dec 2017.

[135] P. Das, P. J. Edavoor, S. Raveendran, and A. D. Rahulkar, "Design and implementation of computationally efficient architecture of PID based motion controller for robotic land navigation system in FPGA," in *Proceedings of Conference on Information and Communication Technology, CICT*, Apr 2018.

[136] G. Bosque, I. Del Campo, and J. Echanobe, "Fuzzy systems, neural networks and neuro-fuzzy systems: A vision on their hardware implementation and platforms over two decades," *Engineering Applications of Artificial Intelligence*, vol. 32, pp. 283–331, Jun 2014.

[137] M. Khosla, R. Sarin, and M. Uddin, "Design of an analog CMOS based interval type-2 fuzzy logic controller chip," *International Journal of Artificial Intelligence and Expert Systems*, vol. 2, no. 4, pp. 167–183, 2011.

[138] A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, and C. A. C. Coello, "Survey of multiobjective evolutionary algorithms for data mining: Part-II," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 1, pp. 25–35, Feb 2014.

[139] A. Arias-Montano, C. a. Coello, and E. Montes-Mezura, "Multiobjective evolutionary algorithms in aeronautical and aerospace engineering," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 5, pp. 662–694, Oct 2012.

[140] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, Part II: Handling constraints and extending to an adaptive approach," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 602–622, Aug 2014.

[141] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, part I: Solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, Aug 2014.

[142] K. Deb, A. Sinha, and S. Kukkonen, "Multi-objective test problems, linkages, and evolutionary methodologies," in *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation - GECCO '06*, Jan 2006, pp. 1141–1148.

[143] N. Gupta, A. Swarnkar, K. Niazi, and R. Bansal, "Multi-objective reconfiguration of distribution systems using adaptive genetic algorithm in fuzzy framework," *IET Generation, Transmission & Distribution*, vol. 4, no. 12, pp. 1288–1298, Dec 2010.

[144] E. Zitzler, D. Kalyanmoy, and Thie, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evolutionary Computation*, vol. 8, no. 2, pp. 173–195, Jun 2000.

[145] G. Reynoso-Meza, X. Blasco, J. Sanchis, and M. Martínez, "Controller tuning using evolutionary multi-objective optimisation: Current trends and applications," *Control Engineering Practice*, vol. 28, pp. 58–73, Jul 2014.

[146] G. Reynoso-Meza, S. Garcia-Nieto, J. Sanchis, and F. X. X. Blasco, "Controller tuning by means of multi-objective optimization algorithms: A global tuning framework," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 2, pp. 445–458, Mar 2013.

[147] A. Kumar and V. Kumar, "Performance analysis of Interval Type-2 FSM controller applied to a magnetic levitation system," in *Proceedings of 2015 International Conference on Soft Computing Techniques and Implementations (ICSCTI)*, Oct 2015, pp. 107–112.

[148] "GML series magnetic levitation system, User manual and experimental manual," 2007.

[149] H. S. Sánchez, F. Padula, A. Visioli, and R. Vilanova, "Tuning rules for robust FOPID controllers based on multi-objective optimization with FOPDT models," *ISA Transactions*, vol. 66, pp. 344–361, Jan 2017.

[150] R. J. Aumann and S. Hart, *Handbook of Game Theory with Economic Applications*. Elsevier, 1994.

Appendix A

Robotic Manipulator and Benchmark Functions

A.1 2-link Robotic Manipulator with Payload at Tip

The schematic diagram of 2-link robotic manipulator with variable payload at the end-effector (on the tip of the second link) is illustrated in Fig. 2.6 [80, 83]. The details of robotic parameters considered in this thesis are reported in Table A.1. The mass of a payload m_{pl} applied at the end-effector is varied arbitrarily from $0.5kg$ to $0.2kg$ in the duration of $4s$ as illustrated in Fig. 5.3. Following specifications are used to carry out all the simulations and experimental analysis in this thesis.

Table A.1: Description and values of different robot parameters used in this study.

Variables (unit)	Description	Link 1	Link 2
m_1, m_2 (kg)	Mass of link	1	1
l_1, l_2 (m)	Length of link	1	1
I_1, I_2 (kgm^2)	Lengthwise centroid inertia of link	0.2	0.2
v_1, v_2	Coefficient of viscous friction	0.1	0.1
d_1, d_2	Coefficient of dynamic friction	0.1	0.1
g (m/s^2)	Acceleration due to gravity	9.81	9.81
l_{c1}, l_{c2} (m)	Distance between joint of link and center of gravity	0.5	0.5
m_{pl} (kg)	The payload mass at the end of Link 2	$--$	0.5 at $t = 0$

Mathematical Modelling : The modelling of torque (τ_1, τ_2) applied to each link and corresponding link positions (θ_1, θ_2) can be expressed mathematically as

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} -b\dot{\theta}_2 & -b(\dot{\theta}_1 + \dot{\theta}_2) \\ b\dot{\theta}_1 & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} + \begin{bmatrix} v_1\dot{\theta}_1 \\ v_2\dot{\theta}_2 \end{bmatrix}$$

$$+\begin{bmatrix} d_1 sgn(\dot{\theta}_1) \\ d_2 sgn(\dot{\theta}_2) \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{A.1}$$

where

$$M_{11} = I_1 + I_2 + m_1 l_{c1}^2 + m_2(l_1^2 + l_{c2}^2 + 2l_1 l_{c2} c_2) + m_{pl}(l_1^2 + l_2^2 + 2l_1 l_2 c_2)$$

$$M_{12} = I_2 + m_2(l_{c2}^2 + 2l_1 l_{c_2} c_2) + m_{pl}(l_1^2 + l_1 l_2 c_2)$$

$$M_{21} = M_{12}$$

$$M_{22} = I_2 + m_2 l_{c2}^2 + m_{pl} l_2^2$$

$$b = m_2 l_1 l_{c2} s_2$$

$$G_1 = m_1 l_{c1} g\, c_1 + m_2 g(l_{c2} C_{12} + l_1 c_1)$$

$$G_2 = m_2 l_{c2} g\, c_{12}$$

In above equations, θ_1 and θ_2 symbolize the positions of the links. Other parameters are $c_1 = cos\theta_1$, $c_2 = cos\theta_2$, $s_1 = sin\theta_1$, $s_2 = sin\theta_2$, $c_{12} = cos(\theta_1 + \theta_2)$. The specification of remaining parameters can be seen in Table A.1.

The Desired Trajectories : The main aim in robotic control system is to design a controller to determine proper actuator signal providing motion to the manipulator links and the end-effector. The desired motion of a manipulator in a multidimensional space is defined in terms of trajectory or path. It is essential to know the details of trajectory for desired goal position and orientation of end-effector. In this thesis, the trajectory equations defined in form of cubic polynomial are used [23, 80]. The trajectory is defined as

$$\theta_{d_i}(t_{sp}) = b_0 + b_1(t_{sp}) + b_2(t_{sp})^2 + b_3(t_{sp})^3 \tag{A.2}$$

along with this, the equations have to satisfy following constraints

$$\dot{\theta_{d_i}}(t_{sp}) = b_1 + 2b_2(t_{sp}) + 3b_3(t_{sp})^2 \tag{A.3}$$

$$\ddot{\theta_{d_i}}(t_{sp}) = 2b_2 + 6b_3(t_{sp}) \tag{A.4}$$

here θ_{d_i} (i = 1, 2) states the prescribed reference positions for respective links.

In this work, the positions of the links for defining the trajectory path are given below in terms of the desired via points at particular time and velocities.

Desired Trajectory for manipulator used in Chapter 4, Section 4.4. :

(i) at $t_{sp} = 2s \Rightarrow \theta_{d_1}(t_{sp}) = 1$ rad, $\theta_{d_2}(t_{sp}) = 2$ rad, $\dot{\theta_{d_i}}(t_{sp}) = 0$

(ii) at $t_{sp} = 4s \Rightarrow \theta_{d_1}(t_{sp}) = 0.5$ rad, $\theta_{d_2}(t_{sp}) = 4$ rad, $\dot{\theta_{d_i}}(t_{sp}) = 0$

Desired Trajectory for manipulator used in Chapter 5, Sections 5.3 and 5.4. :
(i) at $t_{sp} = 2s \Rightarrow \theta_{d_1}(t_{sp}) = 1$ rad, $\theta_{d_2}(t_{sp}) = 3$ rad, $\dot{\theta_{d_i}}(t_{sp}) = 0$
(ii) at $t_{sp} = 4s \Rightarrow \theta_{d_1}(t_{sp}) = 0.5$ rad, $\theta_{d_2}(t_{sp}) = 5$ rad, $\dot{\theta_{d_i}}(t_{sp}) = 0$

A.2 Benchmark Test Functions

The performance of new algorithm is validated through several performance metrics evaluated on standard test bed functions against other state-of-the-art algorithms. This test bed is comprised of 7 unimodal (f_1 to f_7), 6 multimodal (f_8 to f_{13}), and 8 fixed dimension multimodal benchmark functions (f_{14} to f_{21}), and a detailed description of these are given in appendix Tables A.3 and A.4. For further experimentation, 6 composite functions from CEC 2014 [124] test bed (f_{22} to f_{27}), as given in Table A.2 are also investigated. All these functions are to be minimized and corresponding optimal solutions are given in tables. The function specifications such as mathematical formula, dimension (n), upper and lower limits of variables (Range $[Lb, Ub]$), and optimum solution (f_{min}) are detailed in following tables.

Table A.2: Descriptions of the composite benchmark functions.

Function		Dimension (n)	Range $[Lb, Ub]$	f_{min}
Composite functions				
$f_{24}(x)$	$CF1$	10	$[-100, 100]$	2300
$f_{25}(x)$	$CF2$	10	$[-100, 100]$	2400
$f_{26}(x)$	$CF3$	10	$[-100, 100]$	2500
$f_{27}(x)$	$CF4$	10	$[-100, 100]$	2600
$f_{28}(x)$	$CF5$	10	$[-100, 100]$	2700
$f_{29}(x)$	$CF6$	10	$[-100, 100]$	2800
$f_{30}(x)$	$CF7$	10	$[-100, 100]$	2900
$f_{31}(x)$	$CF8$	10	$[-100, 100]$	3000

Table A.3: Descriptions of the unimodal and multimodal benchmark functions.

Function	Dimension (n)	Range $[Lb, Ub]$	f_{min}
Unimodal functions			
$f_1(x) = \sum_{i=1}^{n} x_i^2$	30	$[-100, 100]$	0
$f_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30	$[-10, 10]$	0
$f_3(x) = \sum_{i=1}^{n} (\sum_{j-1}^{i} x_j)^2$	30	$[-100, 100]$	0
$f_4(x) = max_i\{\lvert x_i \rvert, 1 \leqslant i \leqslant n\}$	30	$[-100, 100]$	0
$f_5(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 2)^2]$	30	$[-30, 30]$	0
$f_6(x) = \sum_{i=1}^{n} ([x_i + 0.5]^2$	30	$[-100, 100]$	0
$f_7(x) = \sum_{i=1}^{n} i x_i^4 + random[0, 1]$	30	$[-1.28, 1.28]$	0
Multimodal functions			
$f_8(x) = \sum_{i=1}^{n} -x_i sin(\sqrt{\lvert x_i \rvert})$	30	$[-500, 500]$	$-418.98 \times n$
$f_9(x) = \sum_{i=1}^{n} [x_i^2 - 10cos(2\pi x_i) + 10]$	30	$[-5.12, 5.12]$	0
$f_{10}(x) = -20exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2})$ - $exp(\frac{1}{n}\sum_{i=1}^{n} cos(2\pi x_i)) + 20 + e$	30	$[-32, 32]$	0
$f_{11}(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} cos(\frac{x_i}{\sqrt{i}}) + 1$	30	$[-600, 600]$	0
$f_{12}(x) = \frac{\pi}{n}[10sin(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2[1 + 10sin^2(\pi y_{i+1})] + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_i+1}{4}$ $u(x_i, a, k, m) = k(x_i - a)^m : x_i > a; 0 : -a < x_i < a; k(-x_i - a)^m : x_i < -a$	30	$[-50, 50]$	0
$f_{13}(x) = 0.1\{sin^2(3\pi x_i) + \sum_{i=1}^{n}(x_i - 1)^2[1 + sin^2(3\pi x_i + 1)] + (x_n - 1)^2[1 + sin^2(2\pi x_n)]\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	30	$[-50, 50]$	0

Table A.4: Descriptions of the fixed dimension multimodal benchmark functions.

Function	Dimension (n)	Range $[Lb, Ub]$	f_{min}
Fixed dimensional multimodal functions			
$f_{14}(x) = (\frac{1}{500} + \sum_{i=1}^{25} \frac{1}{j+\sum_{i=1}^{2}(x_i)-a_{ij}})^{-1}$	2	$[-65, 65]$	1
$f_{15}(x) = \sum_{i=1}^{11}[a_i - \frac{x_1(b_i^2+b_i x_2)}{(b_i^2+b_i x_3+x_4)})]^2$	4	$[-5, 5]$	0.00030
$f_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	$[-5, 5]$	0
$f_{17}(x) = (x_2 - \frac{5.1}{4\pi^2}x_i^2 + \frac{5}{\pi}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi})cosx_1 + 10$	2	$[-5, 5]$	0.398
$f_{18}(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)$	2	$[-2, 2]$	3
$f_{19}(x) = -\sum_{i=1}^{4} c_i exp(-\sum_{i=1}^{3} a_{ij}(x_j - p_{ij})^2)$	3	$[1, 3]$	-3.86
$f_{20}(x) = -\sum_{i=1}^{6} c_i exp(-\sum_{i=1}^{3} a_{ij}(x_j - p_{ij})^2)$	6	$[0, 1]$	-3.32
$f_{21}(x) = -\sum_{i=1}^{5}[(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	$[0, 10]$	-10.1532
$f_{22}(x) = -\sum_{i=1}^{7}[(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	$[0, 10]$	-10.4028
$f_{23}(x) = -\sum_{i=1}^{10}[(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	$[0, 10]$	-10.5363