DESIGN OF INTELLIGENT OPTIMAL FOPID/PID CONTROLLER FOR INDUSTRIAL SYSTEMS AND INTEGRATING OF IOT APPLICATION

Submitted in partial fulfillment of the requirements

for the award of the degree of

DOCTOR OF PHILOSOPHY

by

JAILSINGH BHOOKYA

(Roll No: 716146)

Supervisor:

Dr. J. RAVI KUMAR
Associate Professor

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY WARANGAL
TELANGANA STATE-506004, INDIA

April 2021

Approval Sheet

This thesis entitled "**Design of Intelligent Optimal FOPID/PID Controller for Industrial Systems and Integrating of IoT application**" by **Jailsingh Bhookya (716146)** is approved for the degree of Doctor of Philosophy.

Examiners

Supervisor(s)

Dr. J. Ravi Kumar
Associate Professor

Chairman

Prof. L. Anjaneyelu
Professor and Head

Date: ______________

DECLARATION

This is to certify that the work presented in the thesis entitled "**Design of Intelligent Optimal FOPID/PID Controller for Industrial Systems and Integrating of IoT application**" is a bonafide work done by me under the supervision of **Dr. J. Ravi Kumar**, Associate Professor, Department of Electronics and Communication Engineering, National Institute of Technology Warangal, India and was not submitted elsewhere for the award of any degree.

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea / data / fact / source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Jailsingh Bhookya

(Roll No: 716146)

Date:......................

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY WARANGAL
TELANGANA STATE-506004, INDIA

CERTIFICATE

This is to certify that the thesis entitled "**Design of Intelligent Optimal FOPID/PID Controller for Industrial Systems and Integrating of IoT application**", which is being submitted by **Mr. Jailsingh Bhookya (Roll No: 716146)**, in partial fulfillment for the award of the degree of Doctor of Philosophy to the Department of Electronics and Communication Engineering of National Institute of Technology Warangal, is a record of bonafide research work carried out by him under my supervision and has not been submitted elsewhere for any degree.

Dr. J. Ravi Kumar
(Research Supervisor)
Associate Professor
Department of E.C.E.
N.I.T. Warangal
Warangal - 506004, India

<u>Dedicated with Gratitude</u>

to

My family & Teachers

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my supervisor **Dr. J. Ravi Kumar** for the continuous support of my Ph.D. study and related research, for his patience, motivation, and guidance with his moral values. My sincere thanks to him for providing me an opportunity to join the institute as a Ph.D. research scholar and giving me access to the research facilities. I find words inadequate to thank him for enabling me to complete this work in spite of all obstacles. The thesis would not have seen the light of the day without his insistent support and cooperation. Also, I think that he deserve an extra note of thanks for all of his ongoing leadership and support. Finally, I just wanted to thank him for being such a great advisor in my personal day to day life, I have enjoyed working with him so much over the past few years.

My special words of thanks should also go to **Prof. L. Anjaneyelu**, Head of the department, Electronics and Communication Engineering Department, NIT Warangal for his valuable suggestions and support that he shared during my research tenure.

Here, I also would like to take this privilege to thankful my DSC members, Prof. T. Kishore Kumar, Department of Electronics and Communication Engineering, NIT Warangal, Prof. C. B. Rama Rao, Department of Electronics and Communication Engineering, NIT Warangal and Dr. A. Seshagiri Rao, Associate Professor, Department of Chemical Engineering, NIT Warangal for their continuous support, suggestions and advice during my research period whenever required.

I also appreciate the encouragement from teaching staff, non-teaching members and fraternity of Dept. of E.C.E. in N.I.T. Warangal. They have always been encouraging and supportive towards my research.

I thank my fellow labmates (Dr. Hathiram Nenavath, Mr. Kishor Ingle, Mr. Prathap Soma, Mr. A. B. Ahadit, Mr. M. Vijaya Kumar) for the stimulating discussions, for the sleepless nights we were working together before deadlines, and for all the fun we have had in the last few years.

It is my pleasure to show my indebtedness to my co-scholars at NITW like Dr. R. Shashank, Dr. V. Santhosh Kumar, Mr. B. Roshan, Dr. M. A. Mushahhid Majeed, Mr. T. Sunil Kumar and friends for their help during the course of this work. Their timely help and friendship shall always be remembered during my stay.

I owe my deepest gratitude towards my better half (**Saritha Bhookya**) for her continuous support, love and understanding of my goals and aspirations. Her patience and sacrifice will remain my inspiration throughout my life. Without her help, I would not have been able to complete much of what I have done and become who I am. Also, I love my daughter (**Khyati Priya Bhookya**) for giving me happiness and support during the complete research. I appreciate my daughter for abiding my ignorance and the patience she showed during my research. I consider myself the luckiest in the world to have such a lovely and caring family, standing beside me with their love and unconditional support.

My heartfelt thanks goes to my family members (parents **Sathamma-Rajanna**, brothers family **Aruna-Balsingh**) for supporting me spiritually throughout my life in general. Their unconditional love, support, self-sacrifices and patience has always been the source of my strength. I shall be grateful forever because of their kindness and love.

Moreover, I take this opportunity to express special gratitude to my sister's family **Gugulothu Sujatha-Bhaskar**. I find no words inadequate to express any form of special acknowledgment to them for their love, constant inspiration, encouragement, financial help and moral support in any situation. In my life I shall be grateful forever to them and they are my best friends.

Last but not least, I want to mention my sincere thanks to my friends Mr. R. Prakash Kumar for his financial support and love, Mr. G. Bhaskar, Mr. G. Santhosh Kumar has supported me during this research.

Finally, I thank God, for filling me every day with new hopes, strength, purpose and faith.

ABSTRACT

This thesis focuses on developing novel intelligent Fractional-Order (FO) Proportional-Integral-Derivative (PID) controller designs for industrial systems and the integration of the Internet of Things (IoT) applications in the control system. To make further credences in this and its profound existence in the recent state-of-the-art design of optimal FOPID/PID controller, develop a novel optimization algorithm and experimental validation of IoT application integration.

Most of the standard systems are the Single Input Single Output (SISO) systems. The AVR system is chosen for the study of the SISO system. The controller in the AVR system acts as a significant power controller to maintain terminal voltage characteristics. Hence, the AVR system utilizes the PID/FOPID controller to improve performance. The FOPID/PID controllers are model-based controller design method of a physical system to get better performance. The recently developed sine-cosine-algorithm (SCA) is used for PID/FOPID controller parameter tuning. The FOPID/PID controller is designed by an SCA optimization technique using the AVR system's time-domain objective function. A novel Improved Jaya Algorithm (IJA) for optimization is proposed to develop a PID/FOPID controller in the AVR system. The AVR system has good regulation of terminal voltage characteristics at the output to meet the desired performance.

The industrial systems are generally the Multi-Input-Multi-Output (MIMO) systems. The design of a controller for the MIMO system is also equally important. Hence, the PID/- FOPID controller design is a challenging task because of MIMO systems' loop interactions. A standard example of the MIMO system is the Two-Input-Two-Output (TITO) system to study the multivariable control system's dynamical behavior. The simplified decoupling method is used to have less/minimum interactions of the loop. A new Modified Differential Evolution (MDE) algorithm is proposed to design a PID controller for a TITO system. The same algorithm is then used to design an optimal PID controller for the TITO system with the desired closed-loop response. The MDE based PID controller design obtains good time-domain specifications selected response, and the TITO system's stability. Here, Wood-Berry (WB) and Industrial Scale Polymerize (ISP), these two benchmark cases have demonstrated the proposed technique's effectiveness for the PID controller design in the TITO system. These PID controllers in each loop have a better improvement of the setpoint tracking and disturbance rejection.

A FOPID controller plays an essential role in the stability of industrial systems. The multivariable control systems are difficult to control because of the interactions. A decoupler system is used to eliminates interactions. However, these MIMO system loop interactions can be eliminated using the FOPID controller using optimization algorithms in each loop. Hence, the proposed FOPID control method is designed using bio-inspired algorithms like Teaching Learning-based optimization (TLBO) & SCA to get the desired objectives in the industrial systems. This proposed method aims to (i) eliminate interaction between the control loops and (ii) Reference tracking along the disturbance in each loop. Moreover, a novel FOPID controller design without decoupler for the MIMO/TITO system using an optimization algorithm is proposed in the MIMO/TITO system. The proposed method eliminates the interaction of loops with the optimal tuning of FOPID controller parameters. The result demonstrates the superiority over other controller designs for the MIMO/TITO system.

Further, Nature-inspired optimization algorithms have an advantage in optimization problem-solving. Hence, a novel optimization algorithm called the hybrid MFO-WOA algorithm is proposed to solve global optimization problems and design an optimal FOPID/PID controller. The proposed algorithm has a fast convergence capability than the MFO and WOA algorithms by considering the advantage of both. The algorithm's performance is evaluated using the essential 23 benchmark functions. Then, the results proved that the proposed algorithm is competitive enough with state-of-the-art algorithms. The proposed algorithm is used for the FOPID/PID controller design in case studies of different industrial systems. A hybrid MFO-WOA algorithm does the controller parameters tuning for efficiency with other controller design is presented.

The IoT plays a critical role in industrial automation, and industrial automation is an always evolving research area that involves multi-disciplinary applications. IoT helps streamline and build effective, inexpensive, and responsive device architectures. Hence, the IoT-based controller designs are made to overcome the hazardous industrial systems. Here, an IoT based real-time monitoring and control of liquid level in a single tank system is proposed. A PID controller design using a novel modified Grey Wolf Optimization (mGWO) algorithm to control the liquid level. Later, the PID controller is implemented on the ESP32 module, and an android based application is developed using the Blynk platform to monitor and control plants using IoT. The results show that the mGWO based single tank liquid level control system performs better than the classical tuning and SIMC tuning methods.

This thesis shows that the proposed intelligent controller methods are more efficient. However, the proposed controller designs can be integrated into the IoT application to control better and monitor remote access to industrial systems.

Contents

List of Figures

List of Tables

List of Abbreviations

1-D	One Dimensional
2-D	Two Dimensional
3-D	Three Dimensional
ABC	Artificial Bee Colony
ACO	Ant Colony Optimization
ADC	Analog to Digital Conversion
ALO	Ant Lion Optimization
AVR	Automatic Voltage Regulator
BLT	Biggest Log Modulus
CRONE	Commande Robuste d'Ordre Non Entier
CS	Cuckoo Search
CSA	Crow Search Algorithm
DAC	Digital to Analog Conversion
DDC	Direct Digital Control
DE	Differential Evolution
EC	Evolutionary Computation
FC	Fractional Controller
FFE	Fitness Function Evolutions
FIR	Finite Impulse Response
FO	Fractional Order
FOPID	Fractional Order Proportional Integral Derivative
FOPTD	First Order Plus Time Delay
GA	Genetic Algorithm
GM	Gain Margin
GPIO	General Purpose Input/Output
GWO	Grey Wolf Optimization
HTTP	Hypertext Transfer Protocol

IIR	Infinite Impulse Response
IJA	Improved Jaya Algorithm
IKIA	Improved Kidney Inspired Algorithm
IMC	Internal Model Control
IoT	Internet of Things
JA	Jaya Algorithm
KIA	Kidney Inspired Algorithm
MDE	Modified Differential Evolution
MFO	Moth-Flame Optimization
mGWO	modified Grey Wolf Optimization
MIMO	Multiple-Input Multiple-Output
MIPS	Microprocessor without Interlocked Pipelined Stages
ML	Mitag-Leffler function
MPC	Model Predictive Control
MVO	Multi-Verse Optimizer
NDT	Non-dimensional Tuning
NFL	No Free Lunch
P	Proportional Control
PI	Proportional Integral Control
PID	Proportional Integral Derivative Control
PLC	Programmable Logic Control
PM	Phase Margin
PSO	Particle Swarm Optimization
PWM	Pulse Width Modulation
RC	Resistance Capacitance
REST	Representational State Transfer
RGA	Relative Gain Array
RL	Resistance Inductance
SCA	Sine-Cosine-Algorithm
SCADA	Supervisory Control and Data Acquisition
SIMC	Skogestad Internal Model Control
SISO	Single-Input Single-Output
TITO	Two-Input Two-Output
TLBO	Teaching Learning Based Optimization
WB	Wood-Berry system
ZN	Ziegler-Nichols

List of Symbols

t	Time
σ	Magnitude of real-part
w	Frequency
L	Laplace Transform
z_i	Zeros of the polynomial
p_i	Poles of the polynomial
T	System dynamic operator
C	Celsius
$\mathbb{R}$	
Γ	Gamma function
β	Beta function
T	Total number of iterations
I	Identity Matrix
arg	Argument
eig	Eigenvalue of a matrix
w_b	Low frequency
w_h	High frequency
K_P	Proportional gain
K_I	Integral gain
K_D	Derivative gain
T_i	Integrative time constant
T_d	Derivative time constant
λ	Fractional order of integration
μ	Fractional order of derivation
$PI^{\lambda}D^{\mu}$	Fractional order PID controller
M_P	Peak overshoot
T_S	Settling time
T_R	Rise time
E_{ss}	Steady state error
$\mathbb{R}^n$	
$e^{-\beta}$	Weight factor
Λ	Relative gain array
$e(t)$	Error function

Chapter 1

Introduction

This chapter describes about the brief introduction to the topics covered in this thesis, highlighting its contribution.

The engineering division of Control System Engineering [1] is deals with the principles of control theory to design a system that provides the desired behavior in a controlled mode. Control system engineers often worry about performance and controlling aspects of their environment. The objectives of comprehension and maintenance are complementary, since it is important to understand and model effective structures. Furthermore, control engineers are often considered inadequate control systems like chemical process systems [2]. The current difficulty is for engineers to model and control modern, sophisticated, interconnected systems, like motor speed control, chemical processes and robotic systems. The engineer can simultaneously manage multiple beneficial and exciting industrial automation, the all characteristics of control engineering for regulating system function and industrial-economic methods for social benefit.

In the last few decades, optimization has been spreading at an unusual change in all directions. New algorithmic and theoretical techniques have been established, diffusion has taken place at a rapid change in other disciplines, and our understanding of all aspects of the field has grown much more in-depth. At the same time, the increasingly growing focus on the interdisciplinary. The nature of the discipline is one of the most striking trends in optimization. In applied mathematics, engineering, medicine, economics and other sciences, optimization has become a fundamental method [3].

The focused research work in this thesis is aimed at providing the application of control engineering. Advanced engineering design synthesis techniques in improving novel approaches are essential for the execution of intelligent systems, aiming to improve the methods' efficiency and stability by the efficient optimal smart control system in the Internet of Things (IoT) [4] application practice.

In recent, several embedded or cyber-physical systems [5] are included many control applications as multidisciplinary. The majority of control system applications are applied as a software task on microcontroller/microprocessor. Hence, these control applications are responsible for controlling the physical plants/systems are associated. The physical plants' interconnection to the cyber system roots presents physical time in today's IoT systems. Special care is needed to implement such applications to ensure high performance and guaranteed safety by integrating IoT applications [6].

1.1 Overview

Incredible progress has been made in case of multi-disciplinary structures over the last few decades. Particularly in industrial control and automation, significant technological advances in computing, sensing and wireless communication technologies [7]. Combining these inventions made it possible to embedded electronic devices of pocket size, designed to sense, compute, and communicate fascinating information. Sensors and microprocessors are now universally installed in physical plants, remote systems, buildings, and the environment. In order to perform effective control acts, they will process measurement data in real-time and transmit this data. In IoT applications [8], a control device has sensors, microcontrollers, and actuators linked through a communication network. The automatic control is an integral and unavoidable scheme for space vehicle systems, robotic systems, modern manufacturing systems, and industrial processes involving temperature, pressure, humidity, flow, etc.

Many variables can be sensed simultaneously in industrial control, like temperature, flow, and pressure and distance. In a single-phase, all of these may be interdependent variables that require a microcontroller for complete control. Tools used today may be obsolete tomorrow because of the rapid developments in technology, as new and more effective measurement methods are continuously implemented. There can be greater precision, efficiency, accuracy, and performance driving these changes.

The idea of feedback is deceptively simple. Feedback can reduce the effects of disturbances. It can make a system insensitive to process variations and create a plan that follows commands faithfully. Feedback has also had a profound influence on technology. The Proportional Integral Derivative (PID) control is a simple implementation of feedback [9] control. In PID control, by integral action, it can kill steady-state offsets, and by a derivative action, it can predict the future error. The PID controller alternatives are PI, PD, and P controllers, which are sufficient for many control problems, mainly when process dynamics are known and reasonable performance requirements. PID controllers are discovered in considerable measures in most of the industries. The controllers appear in numerous variant forms, and the PID controller is regularly connected with logic, sequential functions, and simple functional blocks to control complex automation systems applied for power generation, transport, and manufacture. Recent advanced control methods are usually arranged hierarchically, such as model predictive control. PID control is practiced at an inferior level. Many technological changes, changing from pneumatics to microprocessors via electronic tubes, transistors, and integrated circuits, have continued with the PID controllers. The PID controller was affected significantly by the microprocessor. Virtually every PID controller created today is based on microprocessors. PID control has designed other features like automatic tuning, gain scheduling, continuous adaptation, and diagnostics.

By incorporating a non-integer (fractional-order) derivative/integration concept, fractional-order mathematics presents a new way towards the system with exceptional dynamic properties. Fractional-order calculus has been a reasonably successful discussion over the last 300 years [10, 11]. In [12, 13], a well-established mathematical theory and offers additional modeling possibilities [14].

Fractional mathematics exists in terms of applications, made its plan towards complicated mathematical and physical problems. Particular examples of fractional calculus formations are radiation conduction by a semi-infinite material [15], and infinite communication lines [16]. In precise, it can be beneficial to model any system with memory, and hereditary properties [17], taking into a description of fractional calculus dynamics. Also, fractional calculus is a generalization of traditional calculus, and fractional models are usually assumed to present a higher accuracy description of the system's dynamics are based on standard differential equations [18]. In many industrial and scientific areas, fractional calculus is also easily used, for example, in the study of electrical circuits [19], signal processing [20], chemical processes [21] and bioengineering [22]. Hence, in system theory and automatic control, fractional calculus has been especially useful. Fractional differential equations are employed to get a highly accurate dynamic systems model, build the latest control methods, and improve the control loop's stability.

Electrical engineering's general field is continually dependent on its growth on a suitable combination of operational experience in engineering and the responsible endorsement of mathematical advances and discoveries. By inspecting published literature work over the past few decades, the optimization field can partially evaluate the presence of newly developed optimization computational methods in the area.

In the 1950s [23], Bellman's introduced the theory of optimality, and dynamic programming gave quite an impetus to problems involving discrete and discontinuous variables. The invention of the Kuhn-Tucker theorem in 1951 and its introduction to optimization literature led to advances in problem formulation, including inequality constraints. Indeed, progress in electric power systems' design continues to be affected by specific robust nonlinear programming procedures. The maximum theory in the late 1950s and early 1960s concluded in many electrical power systems issues in this approach. Comparatively, the case with dynamic, linear, and nonlinear programming, one of the essential optimization instruments used in our types of problems, is the minimum concept.

1.2 Advanced Control system

A control system is a synthesis of elements that functions together so that a complete system works automatically in a pre-specified desired mode. The characteristic feature of control implies that it decreases personal assistance in denoting technical activities as much as it is feasible. Furthermore, the control system overcomes hazardous workplaces during working hours, consequently allowing people to a better quality of life.

1.2.1 Control System Basis

A system is an aggregate of elements (properly connected) that accomplish an appropriate job. An approach to perform a particular task, a usual input signal $u(t)$ must excite it. Concurrently with scientific terms and symbols are shown in figure 1.1, which provides a clear view of this concept. Note that the $y(t)$ response is often called action or output of systems.

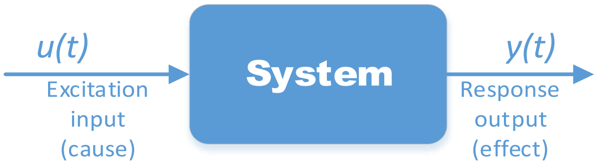


Figure 1.1: Block diagram of a system with its input and output.

Symbolically, the output $y(t)$ is related to the input $u(t)$ by the following Eq.(1.1) [24]

$$y(t) = Tu(t) \tag{1.1}$$

where T is a dynamic system operation. There exist three factors associated in Eq.(1.1): the input $u(t)$, the system operator T , and the output $y(t)$. In engineering problems, two of these three elements to attain the third. As a consequence, the ensuing these three fundamental engineering problems to appear:

1. To study the issue, by the given input $u(t)$, and the system T, then measure the output $y(t)$.
2. To synthesis issue, by the given input $u(t)$, and the output $y(t)$, then required to design the system T.
3. To estimation issue, by the given system T, and the output $y(t)$, then expected to measure the input $u(t)$.

PID Controller

The controller $G_c(s)$ measures an output signal to the control component based on tuning parameters and the controller error $e(t)$. The ideal, continuous and position form of the controller is expressed in Eq.(1.2) [25, 26]

$$u(t) = u_{bias}(t) + G_c(t) \tag{1.2}$$

As before, $u(t)$ is the controller output, $u_{bias}(t)$ is the controller bias. The additional tuning parameters of controller, the Laplace transfer function form for completeness of controller is the ratio of controller output and the error function as the input of controller. The Laplace form of the controller is given in Eq.(1.3)

$$G_c(s) = \frac{U(s)}{E(s)} \tag{1.3}$$

A PID controller is generally given to the control of three words. The first letters of the individual terms that present the regular three-term controller apply to the prompt PID. 'P' is the proportional gain term, 'I' is the integral gain term, and 'D' is the derivative gain term. This PID controller has three gain terms [25]: the proportional term designated as K_P, the integral term specified as $\frac{K_I}{s}$, and the derivative term defined as $K_D s$. Thus, the transfer function of a PID controller has the general form given in Eq.(1.4) [26]

$$G_c(s) = K_P + \frac{K_I}{s} + K_D s = \frac{K_P s + K_I + K_D s^2}{s} = \frac{K_D(s^2 + \frac{K_P}{K_D}s + \frac{K_I}{K_D})}{s} \tag{1.4}$$

where K_P, K_I, and K_D are the proportional, integral, and derivative gains, respectively. This PID controller is also expressed in other form as follows in Eq.(1.5)

$$G_c(s) = K_P(1 + \frac{1}{T_i s} + T_d s); \quad where\ T_i = \frac{K_P}{K_I},\ T_d = \frac{K_D}{K_P} \tag{1.5}$$

where K_P is the proportional gain, T_i is the integration gain time constant, and T_d is the derivative gain time constant.

Unique models of PID controllers are the P, PI, and PD control forms. To promote PID controllers' study, first consider the PD and PI controllers [26].

A typical closed-loop system involving a PID controller is given in Figure 1.2. The PID controller transfer function $G_c(s)$ defined in Eq.(1.5) may further be written as in Eq.(1.6)

$$G_c(s) = K_P(1 + \frac{1}{T_i s} + T_d s) = K_P\frac{(as+1)(bs+1)}{s} \tag{1.6}$$

where $a + b = T_i$ and $ab = T_i T_d$

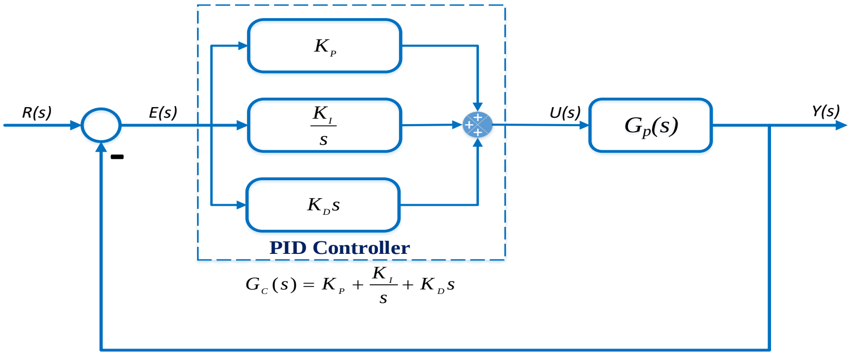


Figure 1.2: Closed-loop system with PID controller.

Hence, the PID controller increases the number of zeros by two and the number of poles by one, where the two zeros are located at $s = \frac{-1}{a}$ and $s = \frac{-1}{b}$ and the pole is located at $s = 0$

FOPID Controller

The general conventional arrangement of a fractional-order PID (FOPID) controller is the $PI^{\lambda}D^{\mu}$ controller [27], proposing integration of order λ and differentiation of order μ, where λ and μ can be any real number. Different alternatives to fractional-order controllers exist, and all have been practiced depends on the application-specific criteria in many applications. The most popular is the family of the FOPID controllers. This FOPID controller is an extension of a conventional PID controller, with the advantage of fractional calculus [28].

The description of Integer Order PID (IO-PID) controller transfer function in the parallel structure is given Eq.(1.4) as shown in Figure 1.2 and the identical generalized fractional order PID controller as presented in Figure 1.3 and the Laplace domain is given [27] in Eq.(1.7) is

$$G_c(s) = \frac{U(s)}{E(s)} = K_P + \frac{K_I}{s^{\lambda}} + K_D s^{\mu} \tag{1.7}$$

where K_P-proportional gain, K_I-integral gain, λ-fractional order of integration, K_D-derivative gain, μ-fractional order of derivation.

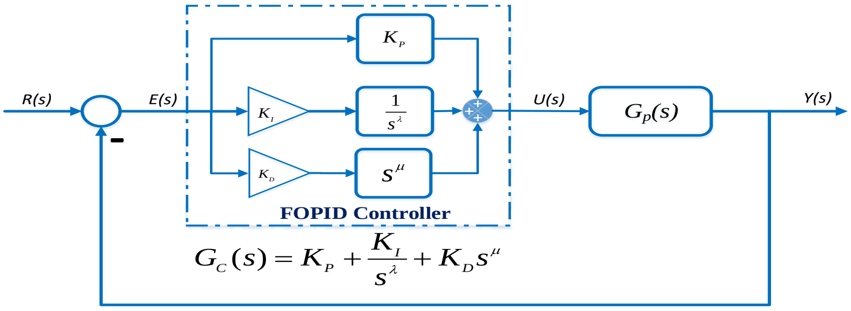


Figure 1.3: Block diagram of FOPID controller in closed-loop.

The FOPID controller system can defeat the parallel structure of the traditional PID controller. With the use of fractional derivative or integral expressions, additional alternatives, including FO-PI and FO-PD controller forms, are possible. The transfer function in Eq.(1.8) writes to the time domain fractional differential equation [29].

$$u(t) = K_P e(t) + K_{I0}\mathrm{D}_t^{-\lambda} e(t) + K_{D0}\mathrm{D}_t^{\mu} e(t) \tag{1.8}$$

The identical transfer function in the discrete-time domain is given in Eq.(1.9)

$$G_c(z) = \frac{U(z)}{E(z)} = K_P + \frac{K_I}{(w(z^{-1}))^{\lambda}} + K_D(w(z^{-1}))^{\mu} \tag{1.9}$$

The FOPID can exist as a PID controller of point to a plane from the Figure 1.4 to present the PID illustration, and the FOPID controllers in the λ-μ axis plane [30]. The point to plan gives P, PI, PD, and PID controllers at the four points on the λ-μ axis plane, and the FOPID controller can hold any point in the plane to derive a controller. Hence, the control designer typically has a more considerable degrees of freedom and can practice specific extra controls to fine-tune the controller design for particular objectives.

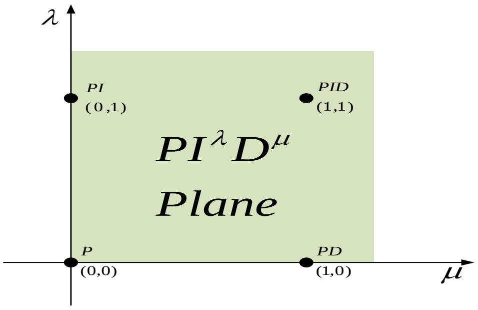


Figure 1.4: The PID Point to plane representation of FOPID controller.

During the work, typically think that the control system is designed by the negative feedback of the pattern in Eq.(1.10)

$$G_{cl}(s) = \frac{G_c(s)G_p(s)}{1 + G_c(s)G_p(s)} \tag{1.10}$$

where $G_c(s)$ is the controller and $G_p(s)$ is the plant under control.

Performance metrics

The development of control system execution in the time-domain is proportional to depreciating error ($e(t)$) value. The various performance measure [31] objectives are estimated to design a controller:

- Integral absolute error (IAE)

$$IAE = \int_0^t |e(t)|\, dt \tag{1.11}$$

- Integral square error (ISE)

$$ISE = \int_0^t e^2(t)dt \tag{1.12}$$

- Integral time-square error (ITSE)

$$ITSE = \int_0^t te^2(t)dt \tag{1.13}$$

- Integral time-absolute error (ITAE)

$$ITAE = \int_0^t t\,|e(t)|\,dt \tag{1.14}$$

- The time domain performance objective function [32]

$$J_{Min}(K) = (1 - e^{-\beta})(M_P + E_{SS}) + e^{-\beta}(T_S - T_R) \tag{1.15}$$

1.2.2 Industrial Controller

The typical industrial control loop configuration is shown in Figure 1.5, which is simple to accommodate higher than four elements. The key features can be group according to the subsequent loop works [33].

Process: This is the actual mechanism for which it is appropriate to control those particular physical variables. Boilers, kilns, furnaces, and distillation columns are standard processes of industry examples.

Actuation: An actuator is a processing unit that provides the process with raw materials or a power supply. It is necessary to view the actuator is working by amplification. For example, a tiny action on a valve stem regulating a great flow of usable gas into an industrial boiler may be the control signal.

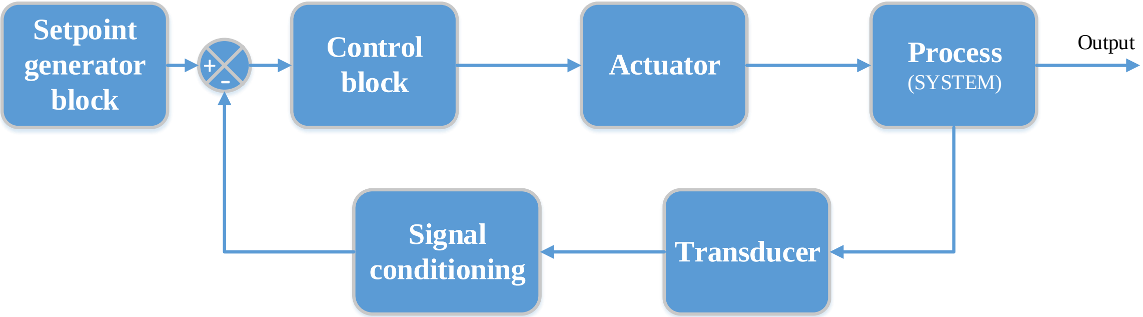


Figure 1.5: Block diagram of an industrial control loop.

Measurement: The familiar motto is that there is no control without a measurement. Usually, a transducer and correlated signal processing elements are used in the measurement process. The transducer will include a sensor to detect a particular physical property (such as temperature) and generate a property description in another physical form (like voltage). A noisy signal will likely be the measured output, and some of the noise may pass into the control loop via the signal-conditioning of the measuring system.

Control: The controller is designed to build a stable closed-loop system and fulfill specific pre-specified dynamic and static process output requirements. An error signal based on the difference between the target set-point or reference signal and the actual calculated output is typically the input to the controller device.

Communications: The earlier parts and components are all connected in the control loop. The control mechanism is usually hardwired in small local loops, but control loops work with control rooms in spatially dispersed processes. Computer communication components (networks, transmitters, and receivers) may be required. Not often is this aspect of control engineering addressed. Communication delays in the loop can be an important barrier to the excellent performance of the control system.

In the construction of the first control design, it is likely to use explicit process modeling. Practically all physical processes are nonlinear in operations, but providentially, various industrial processes are runs to maximize productivity [33]. This industrial system allows linearizing the nonlinear process dynamics of steady-state system operation a feasible research path for several systems. Subsequently, nonlinearity can be resolved by benefit planning or change in management under various operating conditions. While linearity is the route to a simple classification procedure, even a straight-forward loop can commence to a problematic design if all the segments are estimated. A block diagram model as shown in Figure 1.6.

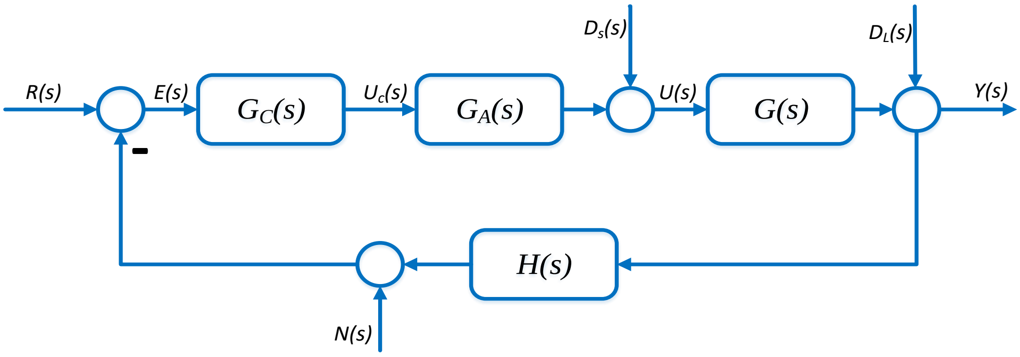


Figure 1.6: Components in a typical industrial control loop.

Table 1.1: Key terms of components model in control loop.

Components		**Loop signals**	
$G(s)$	Process Model	$Y(s)$	Process output
		$D_L(s)$	Load disturbance signal
$H(s)$	Process measurement model	$N(s)$	Measurement noise
$G_c(s)$	Controller unit	$R(s)$	Reference input
		$E(s)$	Error input to controller
		$U_c(s)$	Controller output
$G_A(s)$	Actuator unit model	$U(s)$	Actuator output to process
		$D_s(s)$	Supply disturbance signal

1.2.3 Optimal Control

An important object of control engineering is the design of optimal control systems. The purpose of the plan is to execute a strategy that provides the desired performance with functional components. The predicted performance can be readily specified in terms of time-domain performance indices, such as the integral performance indices. A system's architecture may be based on performance minimization.

As in Eq.(1.16) [34], the output of a control system, written in terms of a system's state variables, can be expressed.

$$J = \int_{t_0}^{\infty} g(\mathbf{x}, \mathbf{u}, t)dt \tag{1.16}$$

where $x(t)$ is the state vector, and $u(t)$ is the control vector. This section considers the idea of optimal control systems utilizing states of system and error performance indices.

1.3 Fractional Calculus

1.3.1 Background

The theory of the differentiation operator $D = \frac{d}{dx}$ is near to everything that has analyzed fundamental calculus. As proper uses, the n^{th} derivative of $f(x)$, specifically $\frac{d^n}{dx^n}f(x)$, is completely determined when n is a real integer. In 1695 LHospital queried Leibniz what sense could be known to $\frac{d^n}{dx^n}f(x)$ if n is a fractional number [21]. After that, fractional calculus has attracted many great mathematicians attention, such as Euler, Laplace, Fourier, Abel, Liouville, Riemann, and Laurent. The approach was then extended to carry the D^m operator, where m

could be rational or irrational, positive or negative, real or complex. Differentiation and integration to an arbitrary order may be a better definition. However, stick to tradition and refer to this principle as fractional calculus [35].

Fractional integrals and derivatives also appear in the theory of control of dynamical systems [35], when a fractional differential equation describes the controlled system or/and the controller. By defining their characteristics in terms of fractional derivatives, the analytical modeling and simulation of structures and methods inevitably lead to differential equations of fractional order and the need to resolve such equations.

Most of the physical systems can be modeled more precisely by fractional differential integrals. The best example is the charging and discharging lossy capacitors, for instance, confirming experimentally to have essential fractional-order dynamics. The fractional-order description, as seen in Figure 1.7 [29], differs from the number line's integration. In integer order calculus, an integer number can be separated or successfully integrated by a function f. This notion of separation reflects the number line's solid dotted points, and integration can be extended to hold any point on the number line between the integer cases. Thus a generalization of the calculus of the integer-order is called the calculus of the fractional order.

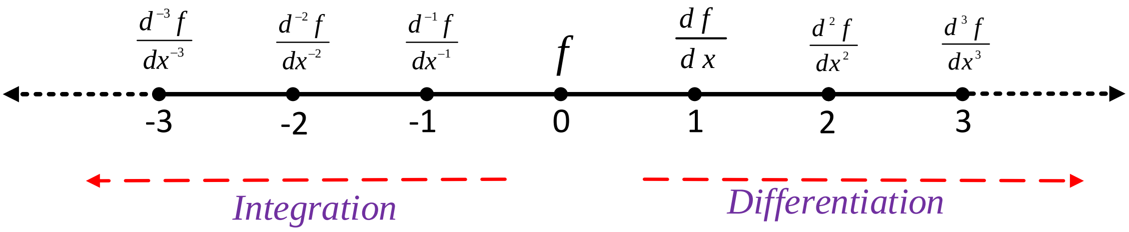


Figure 1.7: Number line and its interpolation for differ-integrals of fractional calculus.

Fractional calculus is a deriving of integration and differentiation to fractional or non-integer order as key operator ${}_a\mathrm{D}_t^n$, where a and t are the limits of the operation and $n \in \mathbb{R}$. The continuous differ-integral operator is defined [35] as in Eq.(1.17)

$${}_a\mathrm{D}_t^n = \begin{cases} \frac{d^n}{dx^n} & n > 0 \\ 1 & n = 0 \\ \int_a^t dx^n & n < 0 \end{cases} \tag{1.17}$$

Fractional differ-integral definitions

1. Grünwald-Letnikov (G-L) Definition [35]: This formula is a continuation of the backward finite difference formula. This formula is used commonly for the numerical solution of a function's fractional differentiation or integration. The alpha order differ-integration of the function $f(t)$ is defined by the Grunwald-Letnikov method as:

$$\frac{d^n}{dx^n}f(x) =_a \mathrm{D}_t^n f(x) = \lim_{h\to 0}\frac{1}{h^\alpha}\sum_{j=0}^{\alpha}(-1)^j\binom{\alpha}{j}f(t-jh) \tag{1.18}$$

where $\binom{\alpha}{j} = \frac{\alpha!}{j!(\alpha-j)!}$
Now the Grünwald-Letnikov definition of Eq. (4.21) extends for non-integer orders and the corresponding n^{th} order differ-integration (for any real number n) of a function $f(x)$ defined as in Eq. (1.19)

$${}_a\mathrm{D}_t^n f(x) = \lim_{h\to 0}\frac{1}{h^\alpha}\sum_{j=0}^{\alpha}(-1)^j\binom{\alpha}{j}f(t-jh) \tag{1.19}$$

Instead of an integer value n, for a non-integer value n, the factorial expression $\binom{\alpha}{j}$ can be replace by the Euler's Gamma function, i.e.

$$\binom{\alpha}{j} = \frac{\Gamma(\alpha+1)}{\Gamma(j+1)\Gamma(\alpha-j+1)} \tag{1.20}$$

Considering, $\alpha = \frac{t-a}{h}$ where a is a real constant, which expresses a real value, the differ-integral can be expressed as

$${}_a\mathrm{D}_t^n f(x) = \lim_{h\to 0}\frac{1}{h^\alpha}\sum_{j=0}^{\frac{t-a}{h}}(-1)^j\binom{\alpha}{j}f(t-jh) \tag{1.21}$$

2. Riemann-Liouville (R-L) Definition [35]: This definition is an extension of n-fold successive integration and is widely used for analytically finding fractional differ-integrals. By the Riemann-Liouville formula the n^{th} order integration of a function $f(x)$ is defined as in Eq. (1.22)

$${}_a\mathrm{I}_t^n f(x) =_a \mathrm{D}_t^{-n} f(x) = \frac{1}{(n-1)!}\int_a^t f(\tau)(x-\tau)^{n-1}d\tau \tag{1.22}$$

when $0 < n < 1$

Assume that $n1 < \beta \leq n$, and denote $n = \beta$. The RiemannLiouville fractional-order derivative can be expressed as in Eq. (1.23)

$$_a\mathrm{D}_t^{\beta} f(x) = \frac{1}{\Gamma(\alpha - \beta)} \frac{d^{\alpha}}{dx^{\alpha}} \int_a^t \frac{f(\tau)}{(t-\tau)^{1+\beta-\alpha}} d\tau \tag{1.23}$$

In such a definition, the power of the term $t\tau$ in the integrand is ensured to be not less than 1.

3. Caputo Definition [35]: In the fractional order systems and control related literatures mostly the Caputos fractional differentiation formula is referred. This typical definition of fractional derivative is generally used to derive fractional order transfer function models from fractional order ordinary differential equations with zero initial conditions. According to Caputos definition the n^{th} order derivative of a function $f(x)$ with respect to time is given in Eq. (1.24)

$$_a\mathrm{D}_t^{n} f(x) = \frac{1}{\Gamma(\alpha - n)} \int_a^t \frac{f^n(\tau)}{(x-\tau)^{n-\alpha+1}} d\tau \tag{1.24}$$

for $\alpha - 1 \leq n \leq \alpha$

Properties of Fractional Differ-Integrals

Some of the major properties of fractional differentiation and integration are given as follows [36]:

- If $f(x)$ is an analytical function of 'x' then its fractional derivatives $_0\mathrm{D}_t^n f(x)$ is an analytical function of x and n.
- When $n = \alpha$, where 'α' is an integer, the operation $_0\mathrm{D}_t^n f(x)$ gives the same result as the classical differentiation or integration.
- When $n = 0$, the operation $_0\mathrm{D}_t^n f(x)$ is Identity operator. i.e. $_0\mathrm{D}_t^n f(x) = f(x)$.
- Fractional differentiation and integration are linear operators as follows
$_0\mathrm{D}_t^n[af(x) + bf(x)] = a\ {}_0\mathrm{D}_t^n f(x) + b\ {}_0\mathrm{D}_t^n f(x)$
- Fractional order differentiation and integration operates satisfy the additive index law, and also satisfy
$_0\mathrm{D}_t^{\alpha}[{}_0\mathrm{D}_t^{\beta} f(x)] =_0 \mathrm{D}_t^{\beta}[{}_0\mathrm{D}_t^{\alpha} f(x)] =_0 \mathrm{D}_t^{\alpha+\beta} f(x)$

Fractional Differ-Integrals Laplace Transform

Laplace transformation holds an efficient and effective method for solving fractional differential equations that often occur in practical science and engineering challenging issues. The following sections describe the traditional Laplace transform for the integer-order instances and their extensions to the fractional-order case.

The Laplace transform of the R-L fractional integration of order $n > 0$ as define in Eq.(1.22) [35] can be stated as a convolution of two functions $g(x) = t^{q1}$ and $f(t)$.

$$
{}_0\mathrm{D}_t^{-n} f(x) = \frac{1}{\Gamma n} \int_0^t (x-\tau)^{n-1} f(\tau) d\tau = t^{n-1} * f(x) \tag{1.25}
$$

Now, the Laplace transform of the function $g(t)$ is given in Eq.(1.26)

$$
G(s) = L\{t^{n-1}\} = \Gamma(n) s^{-n} \tag{1.26}
$$

Using the convolution property, the Laplace transform of the R-L fractional integral can be obtain as in Eq.(1.27)

$$
L\{{}_0\mathrm{D}_t^{-n}\} = s^{-n} F(s) \tag{1.27}
$$

1.3.2 Fractional order system model

The conventional transfer function scheme for integer order systems is expanded in fractional-order systems. The Laplace transform can attain a fractional-order transfer function demonstration of the system described by a fractional order ordinary differential equation. The fractional-order systems can be classified [35], as shown in figure 1.8.

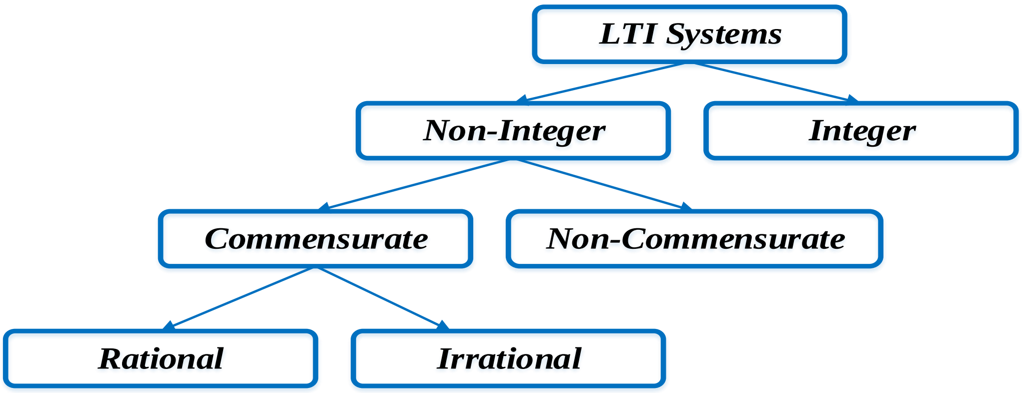


Figure 1.8: Classification of LTI fractional order systems

The system is assumed to be of commensurate order if all the derivation orders are integer multiples of a base order n such that $q \in \mathbb{R}+$. The system can then be expressed [35] as in Eq.(1.28)

$$\sum_{k=0}^{q} a_k D^{kq} y(t) = \sum_{k=0}^{p} b_k D^{kq} u(t) \tag{1.28}$$

If in Eq.(1.28), the order is $q = \frac{1}{r}$, $r \in \mathbb{Z}+$, the system will be of rational order.

Applying the Laplace transform to Eq.(1.28) with zero initial conditions, a fractional-order system can be obtained in the form of a transfer function by its input-output representation.

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^{\beta_m} + b_{m-1} s^{\beta_{m-1}} + ... + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + a_{n-1} s^{\alpha_{n-1}} + ... + a_0 s^{\alpha_0}} \tag{1.29}$$

Deal with the issues of process management in the sense of this work. A process model's fractional-order transfer function representation consists of Eq.(1.29) and an input delay term given as $u(t) = ud(t - L)$ in the time domain. Thus the general form in Eq.(1.30)

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^{\beta_m} + b_{m-1} s^{\beta_{m-1}} + ... + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + a_{n-1} s^{\alpha_{n-1}} + ... + a_0 s^{\alpha_0}} e^{-Ls} \tag{1.30}$$

The following state-space model can also represent the fractional-order linear time-invariant (LTI) system [35, 37].

$$\begin{aligned} {}_0\mathrm{D}_t^{-n} x(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{1.31}$$

A state-space representation is the first equation called the fractional-order state equation in the above fractional-order system, and another one is the output equation. The fractional state-space form may practice the resulting relationship to translate to the fractional transfer function form.

$$G(s) = C(s^q I - A)^{-1} B + D \tag{1.32}$$

If $q_1 = q_2 = ... = q_n$, the system in Eq.(1.31) called a commensurate order , otherwise it is an incommensurate-order system.

Here, I represents the dimension identity matrix $n \times n$ and $G(s)$ represents the dimension fractional order transfer function matrix $p \times l$.

Time Domain Analysis

The other solution includes the mathematical calculation of fractional-order derivatives, which is considered to utilizing an improved Grünwald-Letnikov definition in Eq.(4.23) rewritten [38] as

$$ {}_{a}\mathrm{D}_{t}^{n} f(x) = \lim_{h\to 0} \frac{1}{h^{\alpha}} \sum_{j=0}^{\frac{t-a}{h}} w_{j}^{(\alpha)} f(t-jh) \tag{1.33}$$

where h is the computation step-size and $w_j^{(\alpha)} = (-1)^j \binom{\alpha}{j}$ can be evaluated recursively from given Eq.(1.34)

$$w_0^{(\alpha)} = 1,\ w_j^{(\alpha)} = \left(1 - \frac{\alpha+1}{j}\right) w_{j-1}^{(\alpha)},\ j = 1,\ 2,\ 3..... \tag{1.34}$$

To obtain a numerical solution for the Eq.(1.30) or Eq.(1.31) the signal $\hat{u(t)}$ should be obtained first, using the Eq.(1.33) as in Eq.(1.35)

$$\hat{u(t)} = b_m D^{\beta_m} u(x) + b_{m-1} D^{\beta_{m-1}} u(x) + ... + b_0 D^{\beta_0} u(x) \tag{1.35}$$

The system's time-domain responses can be achieved by applying the following Eq.(1.36).

$$y(t) = \frac{1}{\sum_{i=0}^{n} \frac{a_i}{h^{\alpha_i}}} \left[\hat{u(t)} - \sum_{i=0}^{n} \frac{a_i}{h^{\alpha_i}} \sum_{j=1}^{\frac{t-a}{h}} w_j^{(\alpha)} y(t-jh) \right] \tag{1.36}$$

If the system in Eq.(1.30) has a input-output delay L, the resulting delay responses $y_d(t)$ with $y_d(0) = 0$ is obtained such that

$$y_d(t) = \begin{cases} y(t-L), & t > L \\ 0 & Otherwise \end{cases} \tag{1.37}$$

Approximation of Fractional Operators

Oustaloup introduced in [39] and presented in [40, 41], the recursive filter gives the perfect approximation of the fractional operators in the defined frequency range. It is a well-established approach and is mostly used to implement fractional-order systems and controllers realistically.

The approximation of fractional order differentiation or fractional order integration of $(-\alpha)$ by a general transfer function to estimate the zeros and poles of the transfer function using the

following Eq.(1.38)

$$s^{\alpha} = K \prod_{k=1}^{N} \frac{s + w_k^{'}}{s + w_k} \tag{1.38}$$

where

$$w_k^{'} = w_b \cdot w_u^{\frac{(2k-1-\alpha)}{N}} \tag{1.39a}$$

$$w_k = w_b \cdot w_u^{\frac{(2k-1+\alpha)}{N}} \tag{1.39b}$$

$$K = w_h^{\alpha} \tag{1.39c}$$

$$w_u = \sqrt{\frac{w_h}{w_b}} \tag{1.39d}$$

and N is the order of approximation in the valid frequency range (w_b, w_h).

Due to the property of The fractional-order derivative reduces with integer-order derivations for fractional order $\alpha \geq 1$ it holds

$$s^{\alpha} = s^{n} s^{\gamma} \tag{1.40}$$

where $n = \alpha - \gamma$ signifies the integer division of α and s^{γ} is achieved by the Oustaloup approximate approach by using Eq.(1.38). Thus, every operator in Eq.(1.30) may be approximated using Eq.(1.40) and exchanged by the approximation, by allowing a traditional integer-order transfer function. The obtained approximation may be converted to its discrete-time equivalent in the digital implementations using a suitable method.

1.3.3 Stability Analysis

The conventional algebraic approaches, such as the Routh-Hurwitz criterion, cannot be used for stability analysis directly in the fractional-order case [21, 35, 37], except under some special conditions. This is because the fractional-order case does not have a characteristic polynomial. A pseudo-polynomial with rational power is what exists (a multivalued function). Below are such stability requirements for FO systems. The proof, for brevity, is omitted.

Consider the following theorems to evaluate the stability of a fractional system given in Eq.(1.30) or Eq.(1.31).

Theorem 1: (Matignon 1996) A commensurate order system expressed as a rational transfer function in $G(s) = K\frac{\sum_{k=0}^{m} b_k(s^{\alpha})^k}{\sum_{k=0}^{n} a_k(s^{\alpha})^k} = K\frac{Q(s^{\alpha})}{P(s^{\alpha})}$ is stable if and only if the following condition is satisfied in σ-plane:

$$|arg(\lambda_i)| > \alpha\frac{\pi}{2},\ \forall\sigma \in C. \tag{1.41}$$

where $0 < n \leq 1$ and $\sigma := s^n$. When $\sigma = 0$ is a single root of $P(s)$, the system cannot be stable. For $n = 1$, this is the classical theorem of pole location in the complex plane: no pole is in the closed right plane of the first Riemann sheet.

A fractional-order system's stability regions are shown in Figure 1.9. Notice that there are currently no polynomial techniques to evaluate the stability of fractional-order structures [29], either Routh or Jury type.

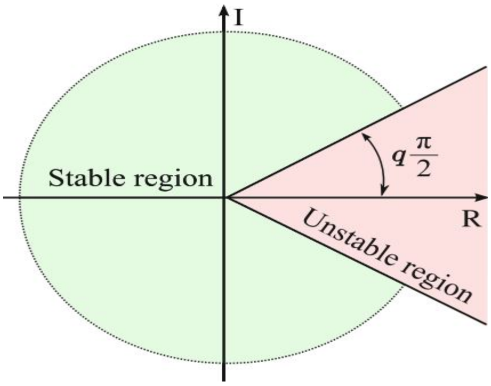


Figure 1.9: fractional-order system's stability region for $0 < n \leq 1$.

Theorem 2: (Matignon 1998) For the commensurate case of the state space representation of FO LTI systems, the system is stable if the following conditions are satisfied

$$|arg(eig(A))| > \alpha\frac{\pi}{2},\ \forall\sigma \in C. \tag{1.42}$$

where $n \in (0,\ 2)$ and $eig(A)$ represents the eigenvalue of matrix A.

In the FO systems, the stability region is shown as shaded in Figure 1.9. The fractional-order systems can be stable even if some of the poles are on the right half-plane inside the shaded region as shown in Figure 1.10, compared to the integer-order case where every pole on the right half-plane indicates an unstable structure.

Theorem 4: The discrete time fractional order state space model is asymptotically stable if the $\left|\tilde{A}\right| < 1$ is satisfied. Here, || denotes the matrix norm defined as max$|\lambda_i|$ and $|\lambda_i|$ being the i^{th} eigenvalue of extended matrix $\tilde{A}$.

The stability criteria of the nonlinear fractional order systems for the commensurate and the incommensurate order systems are stated next.

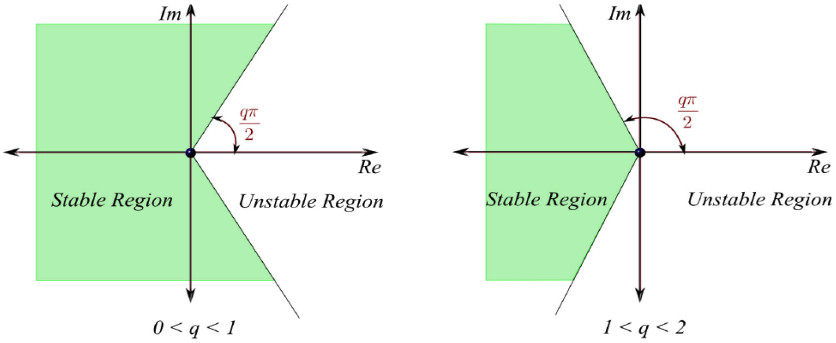


Figure 1.10: fractional-order system stability region for $0 < n < 1$ and $1 < n < 2$.

Theorem 5: The equilibrium points are asymptotically stable for $n_1 = n_2 = ... = n_q \equiv n$ if all the eigenvalues $|\lambda_i|, \forall i = 1,\ 2,\ 3....., q$ of the Jacobian matrix $J = \frac{\partial f}{\partial x}$, where $f = [f_1, f_2, ..., f_n]^T$, evaluated at the equilibrium point, to satisfy the condition as follows

$$|arg(eig(J))| = |(\lambda_i)| > q\frac{\pi}{2}, i = 1, 2, ...q \tag{1.43}$$

Theorem 6: For the incommensurate fractional order system where $n_1 \neq n_2 \neq ..... \neq n_q$. Let $n_i = \frac{v_i}{u_i}$ where $v_i,\ u_i \in \mathbb{Z}^+\quad \forall i = 1, 2, 3, ..., q$ let m be the lowest common multiple of the denominators u_i. Let $\gamma = \frac{1}{m}$ then the fractional order nonlinear system is asymptotically stable if

$$|(\lambda)| > \gamma\frac{\pi}{2} \tag{1.44}$$

for all roots λ of Eq.(1.45)

$$det\,(diag\,([\lambda^{n_1}\lambda^{n_2}.....\lambda^{n_q}])) \tag{1.45}$$

1.4 Evolutionary optimization

In specific ways, an evolutionary optimization algorithm is distinct from classical optimization methods [42]. It depends on random sampling; that is, the technique is non-deterministic. So the strategy of finding an optimal solution has no theoretical guarantee. Second, with a population of candidate solutions, an evolutionary algorithm works. In the meantime, traditional methods of optimization generally retain a single best solution found so far. Using population sets helps avoid being stuck in a local solution [43] by the evolutionary algorithm.

Also, since we already know the global minimum value of the issue beforehand, we will never know whether we have found an actual global minimizer. In general, the upper limit for the number of function evaluations, [43, 44], is typically used to terminate the evolutionary algorithm. The algorithm stops once the number of function evaluations exceeds this upper limit, and the best solution found so far is regarded as a global minimum. In Figure 1.11, the basic scheme of an evolutionary algorithm is given.

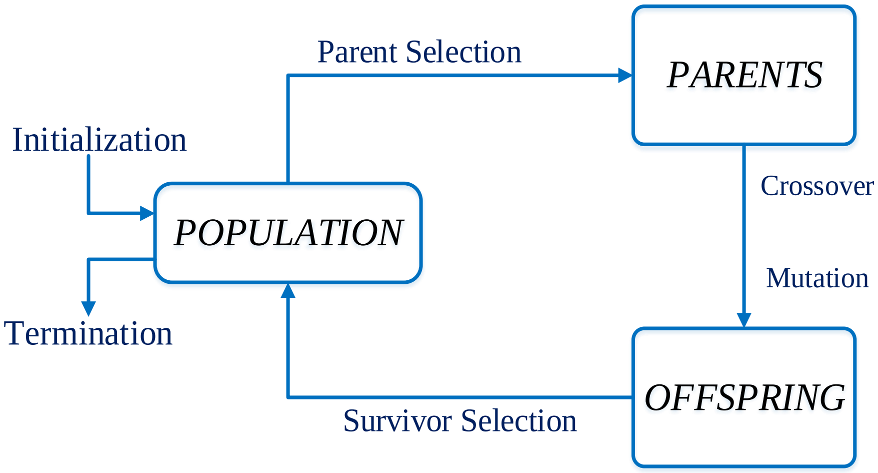


Figure 1.11: Basic scheme of an evolutionary algorithm.

An evolutionary optimization loop used as shown in Figure 1.12, an evolutionary algorithm directly to check for several global solutions, it is possible that the iteration process wanders around the solutions already found without further advancement. We need to stop this hindrance and go further for other solutions because we are looking for all possible solutions. To suggest the fitness function modification procedure here, which helps to go after the other solutions. The modification utilizes the technique of the tunneling function. Once a local or global solution is found during the computation, a new position is built in the second search [45] to escape from this solution region.

It is usually not possible to predict the efficacy of EC implementations in advance. The need for realistic execution and machine experimentation is to justify. Besides, no single optimization algorithm is the best for solving all possible (optimization) problems according to the No Free Lunch (NFL) theorem [46]. From the NFL theorem, two points can be derived. Firstly, EC should not be applied to any situation blindly. Second, for a particular problem, each specific EC algorithm design and execution should be careful.

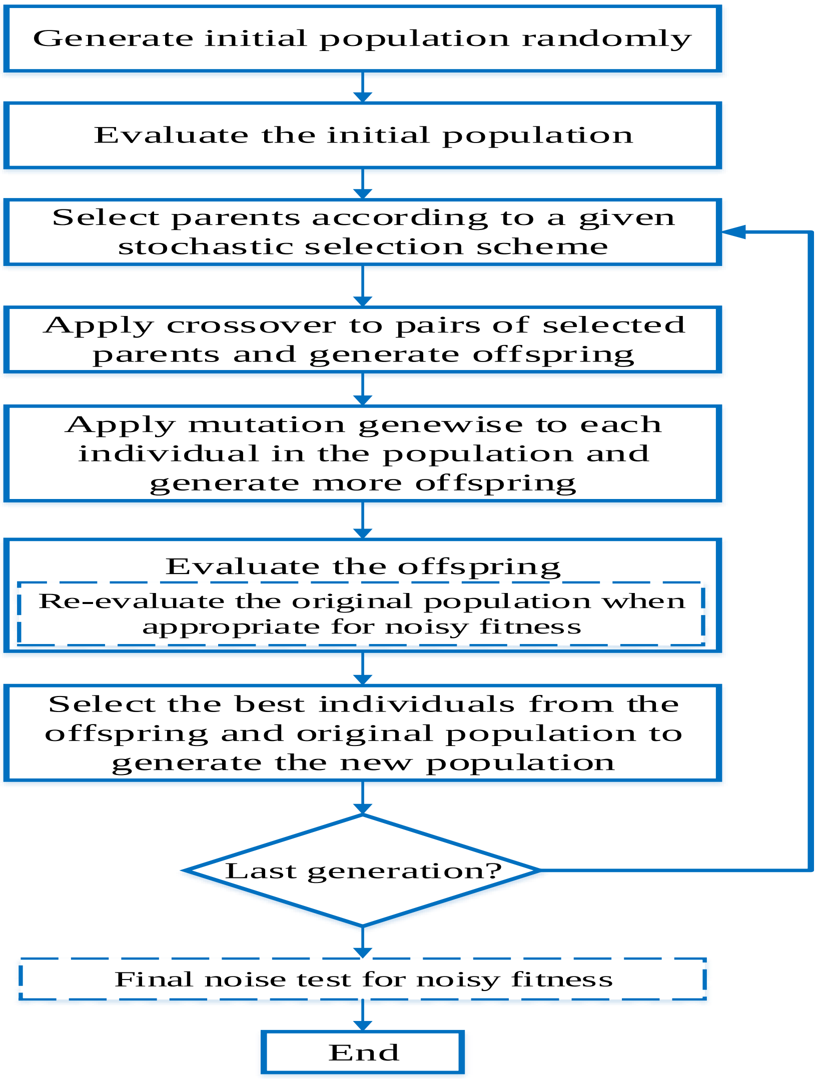

Generate initial population randomly
Evaluate the initial population
Select parents according to a given stochastic selection scheme
Apply crossover to pairs of selected parents and generate offspring
Apply mutation genewise to each individual in the population and generate more offspring
Evaluate the offspring
Re-evaluate the original population when appropriate for noisy fitness
Select the best individuals from the offspring and original population to generate the new population
Last generation?
Final noise test for noisy fitness
End

Figure 1.12: The evolutionary optimization used in the engineering design synthesis process.

1.5 Internet of Things

The Internet of Things (IoT) is a knowledge network in the combination of devices and sensors to allow for new and autonomous capabilities [47]. From the watches we wear to our cars, IoT devices are integrated within our daily lives. Many networks, containing Industrial Control Systems, Control Systems, Building Management Systems, Business, and Staff Networks, will locate them.

Progressively, IoT offers the information technology world with digital skin, allowing it to feel and monitor its operating environment. For this purpose, attempting to protect these devices from those with destructive intent is essential. Increasing the knowledge of the liabilities an IoT device may familiarize into a system is also necessary.

1.5.1 IoT in Control Systems

Nowadays, the Internet is used to communicate with more than 2-billion people per day. The IoT is understood as the expected growth of the Internet containing not only the communication among human but also with any object [48]. An analysis of the convergence of this field and the control system is offered.

The entire Internet was built with human-processed data: websites, blogs , social networks. The problem is that people have little time , energy and precision, which suggests that they are not very good at gathering knowledge about problems in the real world. The primary reason for the control system 's existence is to ensure that all the essential data from a thousand devices required for the functioning of a large scientific facility is collected.

The "Sensor" or "Actuator" is also used to explain things related to the physical world. Sensors play a significant role in extracting knowledge from the real environment that is processed. Traditionally, many realms also use sensors, but to access their information from a computer network, normally located on a local network, a processing unit (computer) is required. The sensors identified by the IoTs are directly linked, similar to computers, to a global Internet network. This concept came with the emergence of new RFID-based technologies that provide passive objects with communication skills. Privacy is a big challenge to overcome, as the system does not have the same sensitivity to monitor the circulation of data as humans. An actuator is a system capable of operating on the real environment. One instance of this group is the IP bulb. Often they are only linked to an intra-net because security is a crucial constraint.

1.5.2 IoT Architecture

The widespread IoT pattern has provided recent technological development for several applications, like smart cities, smart grids, smart buildings, smart homes, industry, transportation, health care, and surveillance. This has allowed the presence of an increasing number of devices connected to interoperable communication protocols and software in the network infrastructure [49]. The new IoT paradigm brings technological fields into a context in which almost everything in the virtual world can be linked and managed, allowing a different aspect of information and communication technology, considering connectivity for anyone, anytime and anywhere [50].

A layer-based architecture was developed to meet the CS needs in an acceptable manner, in view of the technical advancement forced by the IoT. As shown in Figure 1.13 an example to reflect the IoT-CS architecture , two control cores are used. The proposed solution specifies that five hierarchically ordered layers communicate with each of the control cores for problem formulation, called the Application Layer, Network Control Layer, Network Layer, System Management Layer, and Device Layer.

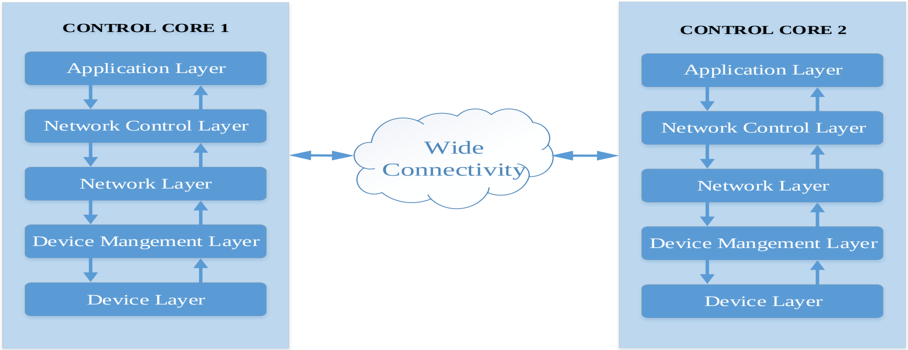


Figure 1.13: IoT-based Control Systems architecture.

The basis for each IoT architecture and its overall process flow of data is similar. Initially, it contains Internet-connected devices that can sense the surroundings around and capture information that is passed on to IoT accesses through their embedded sensors and actuators. Figure 1.14, the subsequent stage involves IoT; these are data acquisition systems and gateways that accumulate and transform the defeating form of unrefined data into digital streams, and pre-process is arranged for examination. In the third layer, edge devices are accountable for additional processing, and enhanced data analysis are quantified in this layer, where visualization and machine learning technology can also move in. Later, the data is transferred to data centers that can be either cloud-based or installed locally. Here the information is stored, treated, and evaluated in detail with actionable insights.

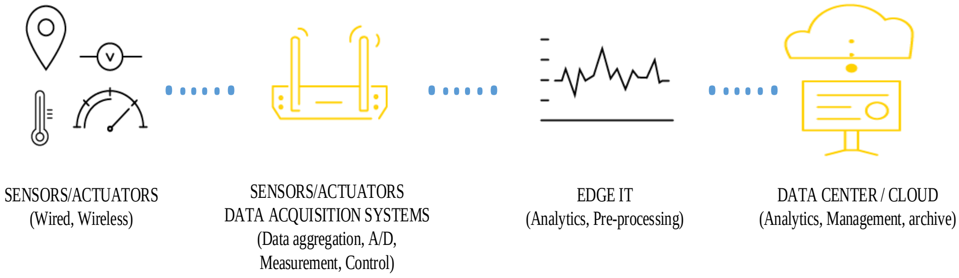


Figure 1.14: Four layers of IoT architecture.

1.6 Literature Survey

The design of control systems is a primary instance of engineering development. The control engineering method aims to achieve the structure, specifications, and identification of a proposed control system's key parameters. In reality, the control system's design requires cyclical effort in which one iterate among [51] modeling, design, simulation, testing, and execution. The control system's design frequently numerous forms and involves a slightly different method to each one.

1.6.1 Control Systems

In control system applications, an effective and economical controller design is a significant challenge for control engineers. Thus, researchers are still concerned with controller architectures, and design methodologies [52]. The proportional integral derivative (PID) controller is still the most important controller used in the industry because of its low cost and specific tuning parameters, Astrom and Hagglund (1994) [53, 54], considering the ability of modern control techniques. Many tuning rules for PID controllers have been suggested in the literature. The most common design technique is that of Ziegler-Nichols, which relies exclusively on parameters obtained from the Ziegler and Nichols (1993) [55] plant phase response. Third-order systems model the processes of many applications in feedback control systems. Unfortunately, for these kinds of scenarios, PID controllers are not acceptable. Thus, for third-order systems, a controller structure different from the PID controller structure is required.

In [56] some different methods for optimizing the dynamic response are discussed in a review of recent work on the AVR system. The previous literature shows better performance in optimal determination of control system parameters compared to other modern control approaches. In the 1990s [57], the AVR system was introduced with self-tuning control. Afterward, at the same time, Finch used a global predictive control technique such as a self-tuning control algorithm [58]. Since the early twentieth century, control engineers have proposed an algorithm for optimization based on adaptive controllers. It is therefore, an excellent option to use artificial intelligence algorithms to evaluate the PID controller parameters in the AVR method [58]. The parameters of a PID controller in the AVR framework are calculated using artificial intelligence algorithms consisting of PSO [59] and GA have compared the results.

The controller design for the multivariable system of multiloop is a challenging task because of the interaction effect. The understanding or measuring of the loop interaction dynamics is complicated. Still, to control MIMO systems, the maximum number of industries are accepted proportional-integral-derivative (PID) controllers because of its simple structure and robustness, the convenience of implementation [60]. Many industrial process systems are using a decentralized controller as an alternative to a multivariable centralized controller because of more complexity than decentralized controllers. These decentralized controller designs are simple to set up, easy to tune, and implementation.

Due to interactions in control loops, the controller parameters' design and tuning are further complexes compared with such single loop controllers. As the controller interacts with each other in the loops, the controller tuning of an independent loop is challenging. The PID controller design using many single-input-single-output (SISO) controllers in each loop is regularly used to control MIMO schemes [61]. The intention of applying SISO PID controllers in multiloop control is to have minimum interaction and a capability to achieve the maximum of control goals. In addition to extending the controller design schemes for SISO systems to MIMO systems, the systems' performance and stability are regularly affected. Several approaches have been suggested like the detuning method [62], relay auto-tuning method [63], sequential loop closing method [64] and independent loop method [65] to exercise loops synergies into account in the multiloop controller design.

1.6.2 Soft Computing in Control

Optimization refers to minimizing or maximizing its performance to discover optimal values for a specified problem. In all fields of study, optimization problems can be found, making optimization techniques essential and a fascinating research path for researchers. In the last two decades, due to the disadvantages of traditional optimization paradigms, local optima stagnation, and the need to derive the search space [66], increasing attention has been seen in stochastic optimization methodologies [67]. Stochastic optimization procedures reflect its issues as black boxes [68]. The mathematical model derivation is unnecessary because these paradigms of optimization adjust the inputs and track the system outputs to optimize or minimize their results. The high versatility means stochastic algorithms are an additional advantage of allowing for problems as black boxes.

Nature-based algorithms begin as a population with an initial collection of variables and then reach the objective function's global minimum or maximum [56]. One of the most common methods in this outline is a genetic algorithm (GA) [69]. Genetic algorithms, as defined by Charles Darwin, are driven by the evolution of living beings. They hire natural genetic variation and natural selection-imitated operators [70]. Particle swarm optimization (PSO) [71] is another instance of nature-based algorithms. The teamwork of groups, such as birds and fish, encourages PSOs. Several particles are put in the problem parameter space in PSOs, and each particle assesses the fitness at its current position. Particles then set their movement through the parameter space by combining some aspect of the past between their cost values with those of the swarm members and then moving between the parameter. In addition to these algorithms inspired by nature, some other algorithms have recently been built based on social humanity. In more applications of nature-based algorithms, these algorithms demonstrate better performance. One of these algorithms is the cultural evolutionary algorithm developed by Jin and Reynolds in 1999 [72]. In cultural algorithms, the primary concept is to nominate values from the derived population and enforce the information to direct the search in return. Another social-based evolutionary algorithm, recently proposed by Atashpaz et al. is the Imperialist competitive algorithm (ICA) [73]. Recently, this algorithm was proposed for the use of various optimization problems.

1.6.3 Optimal Control Systems

Optimal control is a control technique that seeks to find a control policy for a given complex system that optimizes a given output index. The theory of classical optimal control is based on the optimum principle of Pontryagin's [74] or the focus of dynamic programming. The full direction of the Pontryagin's provides a necessary condition for optimality. Also, it gives an open-loop control law, while the principle of dynamic programming offers an essential requirement, which is a partial differential equation. For active systems, its analytical solution is difficult to derive. The optimal control theory for linear dynamical systems is well developed with a strong focus on the linear quadratic regulator and linear-quadratic monitoring.

Over the last few years, many advanced methods have been investigated in the field of control methods for industrial systems or process control [75, 76]. A well-known Proportional Integral Derivative (PID) controller has been broadly utilized in the industries due to its simple structure and significant performance in the comprehensive scope of operational situations. However, successful tuning of PID controller gain parameters is challenging to cope with difficulties in high order, such as time delays and non-linearities. In FOPID controller [77], the tuning parameters are increased to five, but the gain parameters of FOPID controller are still challenging to tune accurately or optimally. For many years, different meta-heuristic algorithms have been applied for tuning FOPID [78] controller gain parameters.

In current years, many evolutionary-based optimization procedures have been projected for PID controller design, rather than using the traditional tuning methods that fail to operate efficiently under different operating conditions and perform poorly. Recently, controller design methods such as bio-geography-based optimization (BBO) [79], gravitational search algorithm (GSA) [80], grasshopper optimization algorithm (GOA) [81], continuous firefly algorithm (CFA) [82], teaching learning-based optimization (TLBO) [83] have been based on heuristic optimization. The various stochastic techniques, such as genetic algorithm (GA) [69], particle swarm optimization (PSO) [84, 85], simulated annealing (SA), and teaching learning-based optimization [86] have been recently gained significant attention for performing high efficient and exploring universal optimum solution of the problem. Due to the increased potential for global optimization of GA, Differential Evolution (DE) and PSO [87] have received considerable attention in control engineering by the exploration for optimum parameters of PID controller.

Many scientists are currently using fractional-order controllers [78] extensively to achieve the most robust efficiency of the [88, 89] method. I. In 1994, Podlubny introduced the fractional PID controller for a fractional-order scheme, [88]. Several additional engineers developed fractional PID controllers using many tuning approaches after this work. This review paper [78] discusses various proposal techniques and software resources existing and modification methods. A fractional PID controller's tuning is difficult since here five parameters to tune, and two more parameters are additional than a traditional PID controller. In this work, various design methods for the fractional-order controller are also established. Besides, the researchers '[90]' have made several contributions. To build design approaches for the FOPID controller constructed on meta-heuristics algorithms in the literature, such as PSO [91], ant colony bee colony and [89] genetic algorithm. In these methods, by minimalizing a given objective function to achieve FOPID controller design, the authors use the principle of optimization. In [92], which is based on PSO, the design method of the FOPID controller for a functional AVR system is proposed, and the effects attained by this method are compared with the traditional PID controller.

1.6.4 IoT Applications

After the Internet, the IoT is considered [93] to be a technological then financial competition in the global information industry. The IoT is a smart network that connects all entities to the Internet in compliance with approved protocols to share information and communicate via data detecting devices. It attains the objective of finding, locating, tracking, controlling, and handling items in an intelligent way [94]. It is an extension and progress of the Internet-based network, which extends the connectivity of objects or things and things from person and human to human. In the IoT model, several artifacts around will be linked into networks in one way or another. Many applications can embed RF Identification (RFID), sensor technology, and other smart technologies.

The evolution towards IoT in home-based networks is previously obvious among the developing approach of Internet-based systems for smart appliances, smart sensors, fitness, or health monitors. These systems use home network WiFi to implement isolated health surveillance and energy-efficient smart home management services and maintain safety and security at home [95]. In general, intermittent (low-rate) or event-driven traffic [96] is manifested by IoT-based home WiFi systems. Body area sensors regularly send symptom measurements and signs to a remote eHealth platform may be an example of intermittent traffic. In particular, the largest IoT situations require sensors and devices transmitting information from home, so these IoT applications flow battle for WiFi's limited uplink capability and frequently struggle with the bulk downloads for massive file transfer of conventional Internet applications [97].

1.6.5 Findings of literature

The control literature articles offering the classical and modern methodology established design and implementation of the PI/PID controllers for a systems class. In this thesis, the PID/FOPID controller scheme for the TITO system is discussed. The present controller design technique for the TITO system can be classified as 1. A centralized method, and 2. A Decentralized method. In a centralized control system, the controller design is reasonably challenging compared to the decentralized control system due to the loops' interaction. In this thesis, the heuristic algorithm assisted the centralized FOPID controller in designing discussed for this thesis's TITO systems.

The Internet revolution is the latest-generation creation of software structures for control systems used by scientific facilities. These frameworks interconnect all devices with the client applications on the primary network. In reality, scientific facilities' direct control systems are primarily tailored for communication from machine to human (monitoring sensors), representing the data for action from human to machine (steering actuators). But it is more and more popular to build virtual controllers with primary Machine-to-Machine communication. In reality, any application that only requires “real-time" can be created in the direct control system. The controllers are completely abstracted in some control system implementations to only represent the functional entity.

1.7 Motivation and Problem

1.7.1 State of the art

An interest in applying fractional calculus in control theory and feedback control occurs throughout the latest decade of the 20^{th} century. In precise, the French CRONE (Commande Robuste d'Ordre Non-Entier) investigation group led by Oustaloup [98], has established tools to recognize and monitor fractional-order systems. In specific, fractional structures, the CRONE community is noteworthy for evolving time-domain system identification techniques. A scheme was proposed in [99] to suggest a fractional differential equation founded upon Grünwald fractional derivation adopting a least-squares error function. The identification of a method for the fractional dynamics of a system in [100] was suggested based on a rational model's approximation of a fractional integrator. For fractional structures, optimal instrumental variable methods have recently expanded. The CRONE research community also established some CRONE controllers, typically based on the frequency domain study of the control systems [98]. These CRONE controls are the well-established technique of Oustaloup approximation that emerged for fractional order operations.

In [30, 101], Podlubny implemented a fractional-order PID (FOPID) controller. Compared with traditional PID controllers in [102, 103], it is reported that FOPID controllers provide superior performance. [78] offers an analysis of tuning methods for the FOPID controllers. Depending on the tuning problem's approach, these approaches can be classified into analytic, rule-based, and numerical. General methods of analytics are scarce. Several rule-based methods were suggested in [104], partly based on traditional tuning rules.Otherwise, the numerical approach has been successfully practiced on common problems with FOPID parameters tuning [98]. In [91], the global optimization based meta-heuristic algorithms are used in controller design methods.

1.7.2 Motivation

Most of the process sector feedback loops are operating on common observation in control loops with self-induced oscillations. Most certainly, all of these loops were tuned by someone with little to no controller tuning expertise. It's not easy, however, even with a background in process control tuning. Typically, there is no time to define a comprehensive model or conduct simulations to find the most suitable tuning when conducting tuning in practice. Typically, such as the open-loop phase test, one uses a simple engineering identification process. An engineering decision is intended behavior, but it generally depends on the scale of disturbances, the variance limits, and the interaction with other controller loops.

The fractional calculus helps the drawbacks of standard differential equations be improved wherever only the integer operator is used. The fractional calculus affords to improve a system's dynamics, studying characteristics such as self-similarity and its state history. In [103], the existing industrial control systems are of great difficulty. Also, fractional dynamics remained found in the time-domain [14] of an almost simple. Therefore, in terms of revised execution, efficiency, and cost reduction, the application of fractional-order identification and control methods to real industrial control problems is expected to affect the specific industrial process positively.

Previously a valid model of a process is developed, a model-based control design can be maintained. With fractional controllers [18, 19], fractional dynamics are better compensated. However, relative to traditional controls, the tuning thereof is more involved. To solve this problem, the numerical methods of optimization are commonly used. The optimization problem must be set up correctly due to the complexities of fractional models.

1.7.3 Problem statement

As mentioned above, the use of fractional models and controllers in industrial applications is of particular significance. The studies show that about 90% of industrial control loops are PI/PID type; also, about 80% of these current control loops are inadequately designed [78, 101, 102]. Therefore, the FOPID controllers provide more tuning freedom and stabilizing capabilities. Industrial integration of these controllers is expected to benefit significantly. Accordingly, to review fractional process models and provide the implementation of an electronic controller, a design method of meta-heuristic algorithms to obtain a collection of FOPID controllers over several operating conditions. The stability of fractional-order systems is to explore the possibilities of incorporating newly developed fractional controllers.

Statement: **“Developing a framework for the optimal tuning parameters of FOPID/PID controller design for industrial (SISO/MIMO) systems using meta-heuristic algorithms and implementation of real-time integration of IoT applications."**

1.7.4 Objectives

The PID controller is mostly used in industries because of the ease of design and implementation. The PID controller design has a limited degree of freedom specifications; to improve the degree of freedom Fractional-order controller is introduced. The fractional controller is called Fractional Order PID (FOPID) controller. Moreover, the FOPID controller has five design parameters to get more specifications. These FOPID controller parameters are in the trade-off between the optimization of five parameters. Then, this FOPID controller is considered an optimization problem to design an optimal control system. In real-time industrial plants can be operated continuously. It is essential to monitor and control the industrial process systems. Therefore, to control and monitor continuously, there is a need for Internet of Things (IoT) based applications.

- Design the optimal FOPID/PID controller for the SISO system using optimization algorithms.
- Design the optimal FOPID/PID controller for the multivariable control system using optimization algorithms.
- To develop a novel Hybrid MFO-WOA algorithm for global optimization and optimal FOPID/PID controller designs.
- To implement the PID controller for a single tank liquid level system using optimization algorithms and experimental validation in the integration of IoT applications.

1.8 Thesis organization

This thesis is done in six chapters and brief outline of the idea is summarized as follows:

Chapter 1 presents the introduction to the research topic. It includes a background of the study, motivation towards the problem statement, and scopes of the research by illustrating the latest literature review related to the research of controller design and IoT applications are disclosed.

Chapter 2 discusses the development of a controller for the SISO system to assess the proposed control scheme's performance effectiveness. The FOPID/PID controller designs are based on the advanced meta-heuristic algorithms for AVR system is elaborated. Also, the improved

Jaya algorithm is proposed to design the FOPID controller in the AVR system. This chapter also reveals the impact of the SISO system to have optimization-based FOPID/PID controller designs.

Chapter 3 focuses on developing a controller for the MIMO system to assess the proposed control scheme's performance effectiveness. The FOPID/PID controller designs are based on the advanced meta-heuristic algorithms for TITO system is elaborated. Here, three optimization algorithms are selected, such as MDE, TLBO, and SCA for obtaining the controller designs. Hence, the aspects of the recommended algorithm are also explained in this chapter. A comparative study between the established algorithms in finding the best controller model is illustrated.

Chapter 4 gives a novel MFO-WOA algorithm for global optimization and FOPID/PID controller designs in SISO and MIMO systems. The proposed algorithm is verified with the existing algorithms and proven that the proposed algorithm gives the best results. The intelligent control schemes have been presented in this chapter to define the optimal FOPID/PID controller of various systems. The objective of the advanced controller is to attain high performance and robust controller in the system. Furthermore, the account of the controllers via three types of systems are validated. Thus, the relative performance between the other controllers is also offered.

Chapter 5 validates the real-time implementation of an industrial process control prototype liquid level system with integration of IoT applications. The development of a controller in the real-time liquid level system using mGWO to give efficient control performance. The controller performance is observed and evaluated based on the IoT application of ESP32 to achieve in suppressing the unwanted disturbances in real-time plants. The IoT application is developed using a gateway module of ESP32 with the plant and a controller by developing a microcontroller in the Arduino environment.

Chapter 6 Finally, this chapter summarizes the thesis conclusions from the contributions and provides a brief discussion of future work direction.

References are given at the end of the chapters.

Chapter 2

Optimal FOPID/PID Controller Design for SISO System

This Chapter attempted to design a controller for SISO systems using optimization algorithms. The AVR system is considered a SISO system; two different methods based FOPID controller has been composed for the AVR system. First, the optimal FOPID controller design applying SCA for the AVR system to maintain the terminal output voltage. Secondly, the optimal FOPID controller design using IJA for AVR system.

2.1 Introduction

A system is classified based on the inputs and outputs of the system. A system has one or many input signals and one or many output signals usually. Hence, from how many inputs and outputs of the system, one usual characterization of systems is a single-input and single-output (SISO) system. The other classification is the multi-input and multi-output (MIMO) system. In control engineering, it is a basic single-variable control system with one input and one output. Based on the number of inputs and outputs present, the control system designs can be described as SISO control systems and MIMO control systems [105, 106].

The SISO is a primary control mechanism that has a single input and produces a single output. Their simplicity of input/output is simple to design/implement and is used for most basic applications. As the name implies, the MIMO system is a complex type of control system with multiple inputs triggering the system controller and producing multiple system outputs in response. These are larger and more complex devices often used in industrial applications, involving a set of outputs. These are very flexible in our everyday lives and at the same time very much seen around us.

Analysis of linear SISO configuration feedback systems, as shown in Figure 2.1. The so-called nominal loop, i.e., the controller's effect interacting with the simple feedback model, is initially analyzed. The result of modeling errors occurs when the controller is applied to the existing system rather than the model.

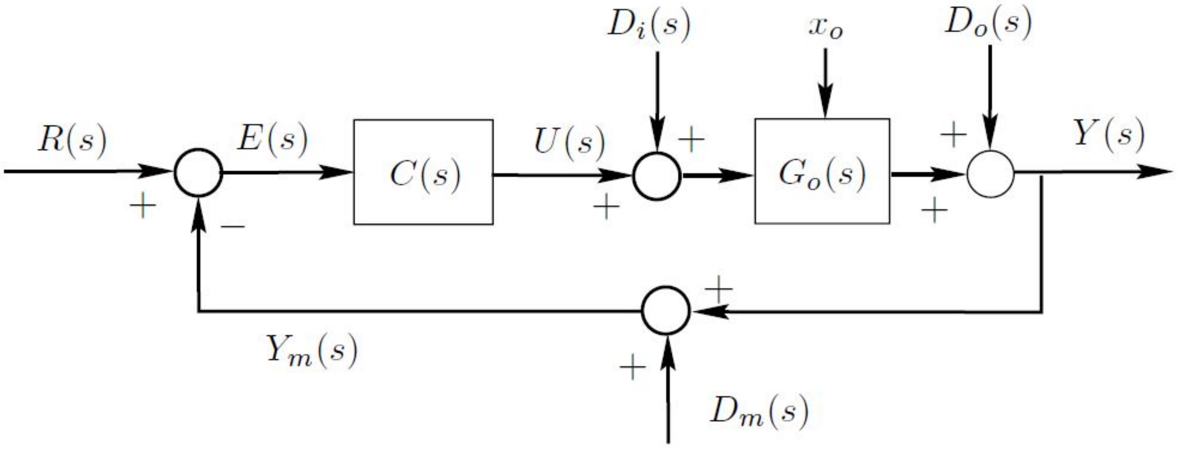


Figure 2.1: Simple feedback control system.

The control loop exposed in Figure 2.1 practices transfer function models of Laplace transforms to describe the relationships between signals. In this case, $C(s)$ and $G_o(s)$ indicate the models of the transfer function of the controller and the actual system, which can be described as follows:

$$C(s) = \frac{U(s)}{E(s)} \tag{2.1}$$

$$G_o(s) = \frac{B_o(s)}{A_o(s)} \tag{2.2}$$

where $B_o(s)$ and $A_o(s)$ are polynomial equations in s, $R(s)$, $U(s)$, and $Y(s)$, these are presents in the Laplace transforms of setpoint, control signal and plant output, $D_i(s)$, $D_o(s)$ and $D_m(s)$ represents the Laplace transforms of input disturbance, output disturbance and measurement noise. Here, x_0 is also use to give the initial condition of the model.

The following relations hold between the variables in the Figure 2.1.

$$Y(s) = G_o(s)U(s) + D_o(s) + G_o(s)D_i(s) + \frac{f(s, x_o)}{A_o(s)} \tag{2.3}$$

$$\begin{aligned} U(s) =& C(s)R(s) - C(s)Y(s) - C(s)D_m(s) \\ =& C(s)\left(R(s) - D_m(s) - G_o(s)U(s) - D_o(s) - G_o(s)D_i(s) - \frac{f(s,x_o)}{A_o(s)}\right) \end{aligned} \tag{2.4}$$

where $f(s,x_o)$ is a linear function of the initial state. The above equations can be solved to yield:

$$U(s) = \frac{C(s)}{1+G_o(s)C(s)}\left(R(s) - D_m(s) - G_o(s)U(s) - D_o(s) - \frac{f(s,x_o)}{A_o(s)}\right) \tag{2.5}$$

and

$$Y(s) = \frac{C(s)}{1+G_o(s)C(s)}\left(G_o(s)C(s)R(s) - D_m(s) - G_o(s)U(s) - D_o(s) - \frac{f(s,x_o)}{A_o(s)}\right) \tag{2.6}$$

The closed loop control system formation shown in Figure 2.1, is called a one degree of freedom (one d.o.f.) structural design. The term redirects the fact that is only one degree of freedom existing to form the two transfer functions from $R(s)$ and $D_m(s)$ to $Y(s)$ and from $D_o(s)$ and $D_i(s)$ to $Y(s)$. Therefore, if the controller transfer function $C(s)$ is designed to provide a exact reference signal response i.e., as follows

$$\frac{Y(s)}{R(s)} = \frac{G_o(s)C(s)}{1+G_o(s)C(s)} \tag{2.7}$$

then this induces a unique output disturbance response

$$\frac{Y(s)}{D_o(s)} = \frac{1}{1+G_o(s)C(s)} \tag{2.8}$$

without any further design freedom.

In advanced control systems, the AVR system can be regarded as a benchmark model of the SISO system [107] with voltage regulation at the output terminal. Moreover, these features are well suited for power systems to give assured control and stability to the mentioned desired output levels in the electrical power system network. Additionally, the purpose of this control is subjected to actual and volatile power. In this case, the difference in the terminal output voltage varies across power with a significant boundary. However, the AVR system controls the output voltage of a synchronized generator at the reference level is a specific controlling problem. Therefore, this work's main objective is to maintain the reference terminal voltage of the AVR system and design an optimal FOPID controller using IJA. The proposed FOPID/PID controller design based on the optimization technique [56] for the AVR system will give better reference tracking and robust performance. The FOPID controller has been tuned optimally using the proposed IJA optimization technique. Moreover, this IJA based FOPID controller in the AVR system transient response is optimal and much better than the current state of the art.

2.2 AVR System Description

The terminal voltage of the AVR system generated by a generator output at the desired level has to improve the quality of electrical power. When the generator output terminal voltage has voltage fluctuations or when a deviation occurs, a sensor detects the terminal voltage and compares the actual measured voltage with a reference voltage for the difference. Then, the difference is used to control the terminal voltage of AVR system. An amplifier magnifies the control signal and it is used to control the generator with excitation from the exciter.

A linear model of the AVR system consists of 4-blocks, the design of a practical and stable automatic voltage regulation is of common concern in electrical power systems and each block of the AVR system [32, 108] with controller block diagram is shown in Figure 2.2.

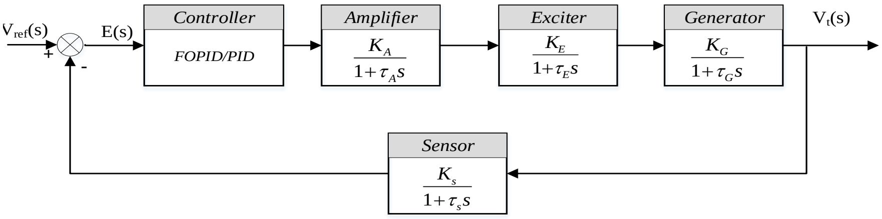


Figure 2.2: Block diagram of the AVR system.

The linear model of each block transfer function [32] representations are as follows.

Amplifier. The amplifier section model considering amplifier gain (K_A), with the amplifier time constant (τ_A), for the model is given as in Eq.(2.9)

$$\frac{V_R(s)}{V_e(s)} = \frac{K_A}{1 + \tau_A s} \tag{2.9}$$

The physical parameter values of amplifier gain (K_A) in the range of 10 to 400, while the time constant (τ_A) in the range of 0.02 to 0.1s.

Exciter. The exciter section is modelled with an exciter gain of K_E and exciter time constant of τ_E, as given in Eq.(2.10)

$$\frac{V_F(s)}{V_R(s)} = \frac{K_E}{1+\tau_E s} \tag{2.10}$$

The values of exciter gain (K_E) in the range of 10 to 400 and the time constant (τ_E) in the range of 0.5 to 1.0s.

Generator. The generator section modeling of the terminal voltage is consider as generator gain (K_G) and time constant (τ_G) is given in Eq.(2.11)

$$\frac{V_t(s)}{V_F(s)}=\frac{K_G}{1+\tau_G s} \tag{2.11}$$

The coefficients of K_G is in the range of 0.7 to 1.0, while the generator time constant τ_G is either 1 or 2s.

Sensor. The sensor section is modeled by sensor gain of K_R and time constant (τ_R) as given in Eq.(2.12)

$$\frac{V_s(s)}{V_t(s)}=\frac{K_R}{1+\tau_R s} \tag{2.12}$$

The values of τ_R remain very small, in the range of 0.01 to 0.06s.

2.3 Methodology

2.3.1 Sine Cosine Algorithm

S Mirjalili proposed an advanced population prescribed optimization method is Sine Cosine Algorithm (SCA) [68] for global optimization in solving engineering problems. This SCA forms many arbitrary candidates as a solution then allows to reach near to the best (global) optimal result by accurate idea made on trigonometric functions are sine and cosine. The following location apprising equations are given for all phases [109]

$$X_i^{g+1} = X_i^g + r_1 * sin(r_2) * |r_3(P_i^g - X_i^g)| \tag{2.13}$$

$$X_i^{g+1} = X_i^g + r_1 * cos(r_2) * |r_3(P_i^g - X_i^g)| \tag{2.14}$$

In generally above two equations Eq.(2.13) and Eq.(2.14) can be combined as one equation as follows in Eq.(2.15)

$$X_i^{g+1} = \begin{cases} X_i^g + r_1 * sin(r_2) * |r_3(P_i^g - X_i^g)| & \text{if } r_4 < 0.5 \\ X_i^g + r_1 * cos(r_2) * |r_3(P_i^g - X_i^g)| & \text{if } r_4 > 0.5 \end{cases} \tag{2.15}$$

where X_i^g is current solution location in the i^{th} dimension at the t^{th} iteration, and r_1, r_2 and r_3 are random numbers, P_i^g is target point location in the i^{th} dimension, and | | describes the absolute value. The r_4 is a random value between [0, 1].

The unique properties of the sine function and cosine functions of Eq.(2.13), Eq.(2.14) are as given in Figure 2.3. It demonstrates that exactly how the equations are designated within dual results in a search space. In demand, to balance utilization and consideration, the range of a sine function and cosine functions in Eq.(2.13) and Eq.(2.14) are an adaptive improvement [68] by next Eq.(2.16)

$$r_1 = a - a\frac{t}{T} \tag{2.16}$$

where t is current iteration, T is maximum number of iterations and a is a constant value. An abstract idea of sine functions and cosine functions by the dimension of in the limits of [-2, 2] is presented in Figure 2.4.

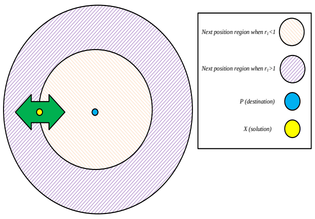

Next position region when r1<1
Next position region when r1>1
P (destination)
X (solution)

Figure 2.3: The effects of sine and cosine functions of Eq.(2.13), Eq.(2.14) in the next position [100].

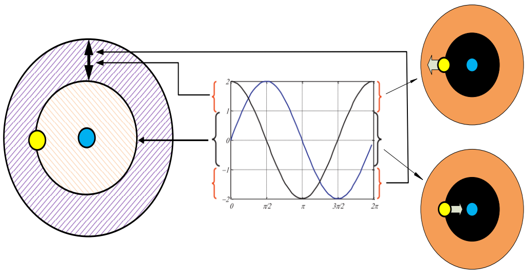


Figure 2.4: Sine and cosine with the range in [-2, 2] allow a solution to go around or beyond the destination [100].

2.3.2 Improved Jaya Algorithm

Jaya Algorithm

Venkata Rao [110] proposed an algorithm with no algorithm specific parameter named Jaya algorithm. This algorithm can be employed for solving variously unconstrained as well as constrained problems.

Let the objective function is to minimize, At any iteration j, there are h number of design variables and i is the number of candidate solutions (i.e. population). If $X_{h,i,j}$ is the value of the h^{th} variable for the i^{th} candidate during the j^{th} iteration, then this value is modified as per the

following Eq.(2.17)

$$X'_{h,i,j} = X_{h,i,j} + r_{1,h,j}(X_{h,best,j} - |X_{h,i,j}|) - r_{2,j,G}(X_{h,worst,j} - |X_{h,i,j}|) \tag{2.17}$$

where $X_{h,best,j}$ is the value of variable h for the best value of objective function and $X_{h,worst,j}$ is the value of variable h for the worst value of objective. $X'_{h,i,j}$ is a new value of $X_{h,i,j}$. $r_{1,h,j}$ and $r_{2,h,j}$ are two random numbers in the space of [0,1] of the h^{th} variable during the j^{th} iteration.

Only general control parameters such as the population size and the number of generations are required in the Jaya algorithm, thus it is fairly convenient for use in practical and the complete flow chart of the algorithm is given in Figure 2.5.

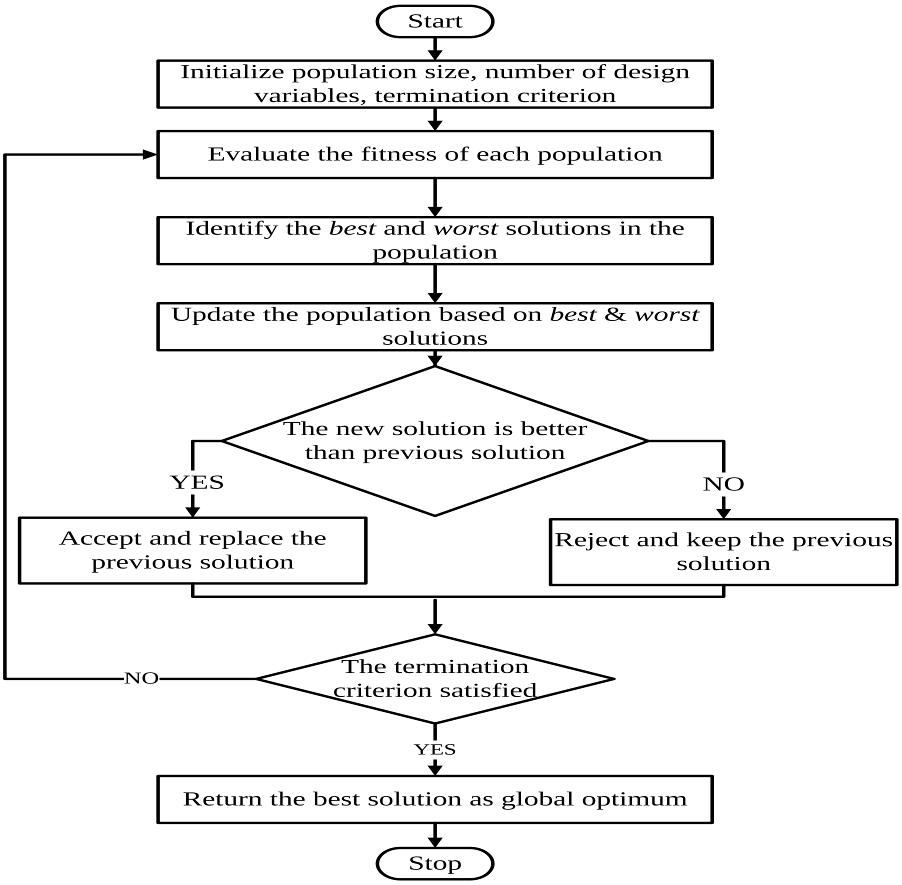

Start
Initialize population size, number of design variables, termination criterion
Evaluate the fitness of each population
Identify the *best* and *worst* solutions in the population
Update the population based on *best* & *worst* solutions
The new solution is better than previous solution
YES
NO
Accept and replace the previous solution
Reject and keep the previous solution
The termination criterion satisfied
NO
YES
Return the best solution as global optimum
Stop

Figure 2.5: Flow-Chart of Jaya algorithm

Improvement in Jaya Algorithm

The Jaya algorithm [111, 112] is attempts to enhance successes by achieving the best solution, the population is at the beging ranked to ascending order as from the *best* to the *worst* based on the fitness function evoluation of each solution, then this can be divided in two groups of population. Note, the first as best group G_b comprise n_b elements and the second is worst group G_w comprise n_w elements. n_b and n_w convince the following Eq.(2.18)

$$n_b + n_w = n \tag{2.18}$$

where n_b is the best group population, n_w is the worst group population and n is total number of population.

Moreover, the better group mean is $X_{h,mbest,j}$ and the worst group mean is $X_{h,mworst,j}$, and these are calculated using the following equations:

$$X_{h,mbest,j} = \frac{\sum_{i=1}^{n_b} X_{h,i,j}}{n_b} \tag{2.19}$$

$$X_{h,mworst,j} = \frac{\sum_{i=1}^{n_w} X_{h,i,j}}{n_w} \tag{2.20}$$

where $X_{h,i,j}$ is the population matrix.

Ultimately, the improved Jaya algorithm (IJA) with these two groups population of solution is updated as in Eq.(2.21), the pseudo code for the improved jaya algorithm is given in Algorithm 1.

$$X'_{h,i,j} = X_{h,i,j} + r_{1,h,j}(X_{h,mbest,j} - |X_{h,i,j}|) - r_{2,j,G}(X_{h,mworst,j} - |X_{h,i,j}|) \tag{2.21}$$

Algorithm 1 : Pseudo code for Improved Jaya algorithm

```
Initialize a set of population 'i' of 'h' individuals
$X_{i,h} = X_{i,1}, X_{i,2}, X_{i,3}, .., X_{i,h}$
while Current iteration < Maximum iterations do
    Evaluate fitness of each population
    Sort the fitness in ascending order
    Obtain the best and worst solutions
    choose $n_b$ and $n_w$ based on Equation (2.18)
    calculate $X_{j,mbest,G}$ and $X_{j,mworst,G}$ based on Equations (2.19) and (2.20)
    Update the population as $X'_{h,i,j} = X_{h,i,j} + r_{1,h,j}(X_{h,mbest,j} - |X_{h,i,j}|) - r_{2,j,G}(X_{h,mworst,j} - |X_{h,i,j}|)$
    if New solution is better then
        accept and replace the previous solution
    else
        reject and keep the previous solution
Result: Return the best/global solution as optimal
```

2.4 Results and Discussions

2.4.1 PID/FOPID Controller Design

The performance indices are integration of absolute errors (IAE), integration of square errors (ISE) and integration of time multiplication of square errors (ITSE) [113] in control systems. These performance indices, IAE, ISE and ITSE are given in Eq.(2.22), Eq.(2.23) and Eq.(2.24) respectively [90].

$$IAE = \int_0^{\infty} |r(t) - y(t)|\, dt = \int_0^{\infty} e(t)dt \tag{2.22}$$

$$ISE = \int_0^{\infty} e^2(t)dt \tag{2.23}$$

$$ITSE = \int_0^{\infty} t * e^2(t)dt \tag{2.24}$$

where $r(t)$, $y(t)$ are the desired and measured voltages respectively, of synchronous generator and $e(t)$ is the error function.

$$J_{Min}(K) = (1 - e^{-\beta})(M_P + E_{SS}) + e^{-\beta}(T_S - T_R) \tag{2.25}$$

where K is [K_P, K_i, K_d] for PID control, K is [K_P, K_i, λ, K_d, μ] for FOPID control and β is the weighting factor in the range of 0.7 to 1.4. T_R is the rise time of the system, M_P is the peak overshoot of the response, T_S is the settling time to take the response and E_{SS} is steady state error of the response. Further, in this controller design, a transient response objective is applied, i.e. in Eq.(2.25) [32] and this objective function based controller parameters will gives best control performance.

Usually, in the control systems engineering, the choice of FOPID/PID [114] controller parameters are more difficult in controller design phase. The solution of controller design problems are switch between swarm and evolutionary metaheuristic algorithms. The optimization methods start by looking at the working of optimizing an objective function within multiple minima. The algorithm is then appropriately altered to discover the controller parameters of a FOPID/PID for adequately controlling an AVR system as shown in Figure 2.6.

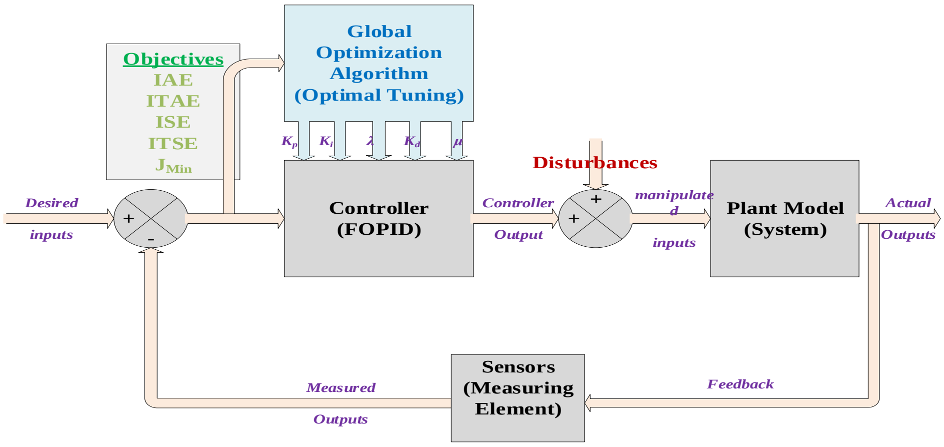


Figure 2.6: Problem formulation of Optimization based control system.

2.4.2 SCA based results

Here, the optimal tuning parameters of FOPID controller are found using a proposed SCA-FOPID method, the effectiveness of a proposed method is exposed on AVR system. The FOPID controller in a control loop of AVR system is given in Figure 2.2. The search space for FOPID parameters are in range as the lower limits [0 0 0.01 0 0.01] and upper limits [2.5 1.5 1.5 0.5 1.5]. The fractional parameter values of λ and μ can be changing with in the range, then the time domain specifications of the system can further better which is advantage of the fractional PID controller. The influence is almost same as beyond the upper limit, almost no influence on the system performance. The AVR system parameters are adopted from [32, 91] and presented in Table 2.1 for simulation verification.

Table 2.1: Parameters of AVR system.

Section	Parameters	
Amplifier	$K_A = 10$	$\tau_A = 0.1$
Exciter	$K_E = 1$	$\tau_E = 0.4$
Generator	$K_G = 1$	$\tau_G = 1$
Sensor	$K_R = 1$	$\tau_R = 0.01$

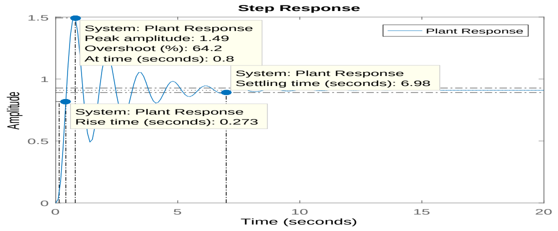


Figure 2.7: Step response of AVR system (No controller).

The unique terminal voltage at AVR system with no control (step response) as shown in Figure 2.7. It is observed that the AVR system has large overshoot, more settling time with many oscillations. Hence, the AVR system needs a controller to control the terminal voltage at a desired level.

The algorithm parameters of SCA method are dimensions of *5*, solution cadidates are set to *50*, $a = 2$, $r_2 = 2\pi \times rand$, $r_3 = 2rand$, $r_4 = rand$ and maximum number of iterations set to *100*. The value of r_1 is decreasing linearly according to Eq.(2.16). Then SCA is takes the objective function of Eq.(2.25) with weightage of $\beta = 1$ to minimize the time domain performance indices, which includes the Rise Time (T_R), Maximum Overshoot (M_p), Settling Time (T_s) and Steady State Error (E_{ss}) of the system with an exponential weightage of ($e^{-\beta}$) for performance matrices. Then this optimization algorithm come up with the optimal solution for tuning of FOPID controller parameters.

The SCA based FOPID controller design in Figure 2.6 is simulated, and the results gives the best transient response of AVR system as shown in Figure 2.8. The closed-loop FOPID controller provides the fast response with the desired terminal voltage level at the generator. The controller design objective convergence of SCA based FOPID controller is very fast as shown in Figure 2.9. The convergence characteristics made the controller to fast action to get better performance of the AVR system.

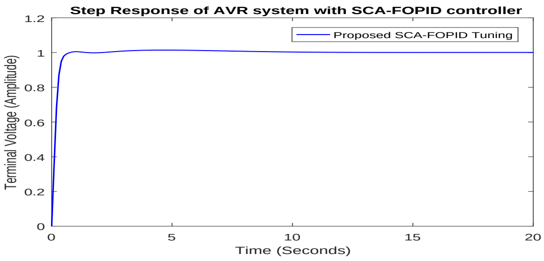


Figure 2.8: AVR system step response with proposed SCA-FOPID controller design

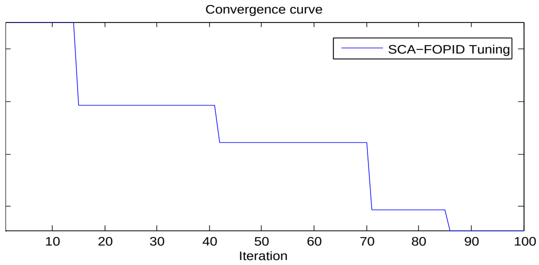


Figure 2.9: AVR system SCA-FOPID controller tuning convergence characteristics

Initially, the AVR system is employed PID controller to control the terminal voltage of the AVR system. This PID controller is tuned using classical methods such as ZN tuning, IMC tuning and optimal tuning methods as Gaing et. al. [32] PSO tuning, Zafer et. al. [115] CSA tuning are as shown in Figure 2.10 along with proposed SCA-PID tuning method.

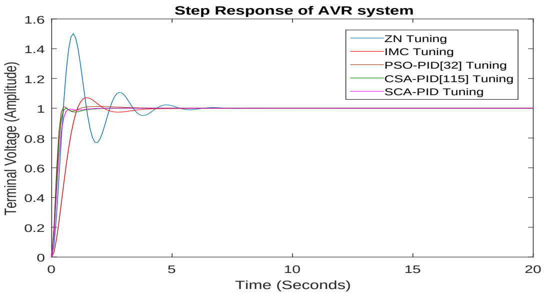


Figure 2.10: Comparision of the AVR system step response with PID controller designs

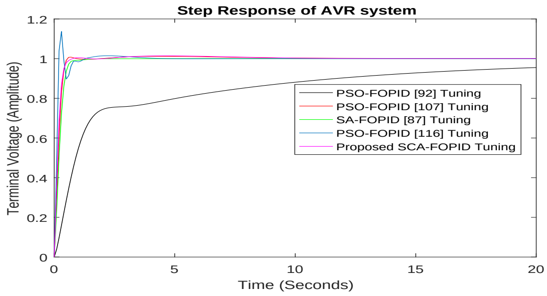


Figure 2.11: Comparision of the AVR system step response with FOPID controller designs

Now, this AVR system is adopted FOPID controller for controlling the terminal voltage of the same system. Then, this FOPID controller is tuned using such as Xiao Li et. al. [87] PSO tuning, Zamani et. al. [92] PSO tuning, Hafsi et. al. [107] SA tuning and Ekinci et. al [116] Improved KIA tuning these methods are as shown in Figure 2.11 along with proposed SCA-FOPID tuning method.

Robust Analysis

Robust analysis is performed to inspect the control system characteristics under parametric variations [117] of AVR system. The SCA optimization algorithm is associated in the FOPID controller to attain the best (optimal) results for the evaluation of objective function.

Tuning parameters of FOPID controller are continual over a time, but the parameters acquired by the SCA optimization based on the reference tracking error performance indices. This is the main feature that stands out for the controllers presented in this paper.

The AVR system dynamics can be change due to load instability, i.e. in τ_A, τ_E, τ_G and τ_R values. Hence, the robustness analysis is essential for the controller in AVR system, to simulate these changes, the parameter values of τ_A, τ_E, τ_G and τ_R are changing randomly. These random uncertainty of AVR system can be responsible by the FOPID controller, that can provide a desired response. Therefore, the FOPID controller is robust in the AVR system for variations in dynamics, the same performances are shown in Figure 2.12 to Figure 2.14.

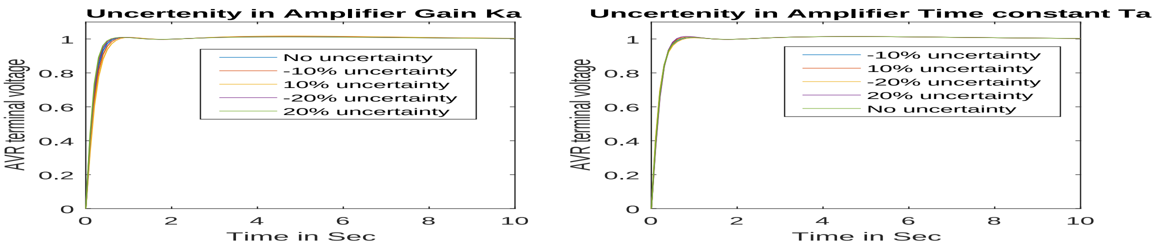


Figure 2.12: Parameter uncertainty in amplifier block

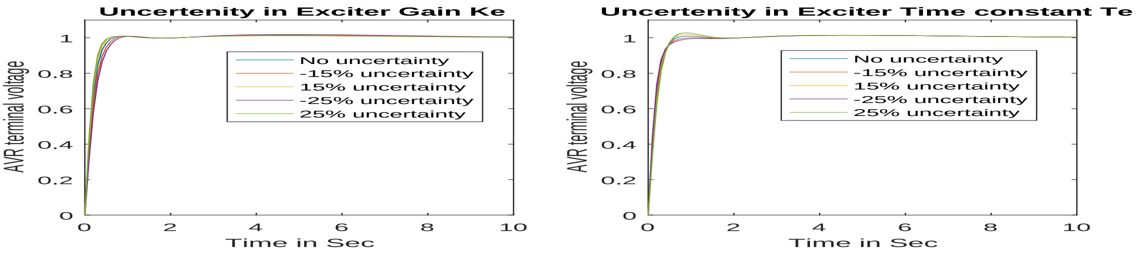


Figure 2.13: Parameter uncertainty in exciter block

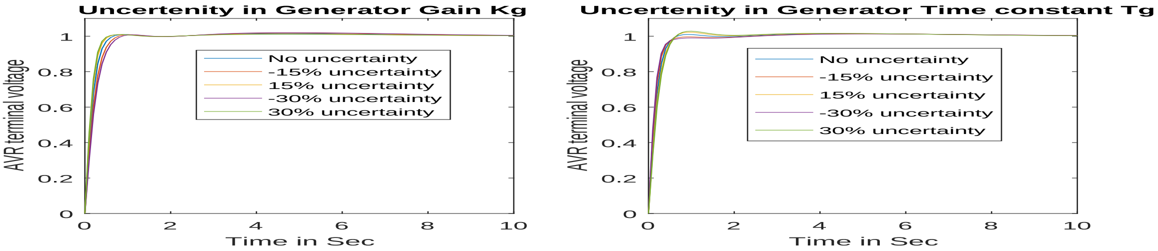


Figure 2.14: Parameter uncertainty in generator block

Discussion

Consider the AVR system is engage with FOPID controller as given in Figure 2.2, to control the nominal terminal voltage of the AVR system at a desired level. Therefore, this FOPID controller parameters are tuned using Gaing [32] PSO tuning, Xiao Li et. al. [87] PSO tuning, Zamani et. al. [92] PSO tuning, Hafsi et. al. [107] SA tuning, Zafer et. al. [115] CSA tuning, and Ekinci et. al [116] Improved KIA tuning these methods are compared with proposed SCA based PID/FOPID controller as shown in Figure 2.15.

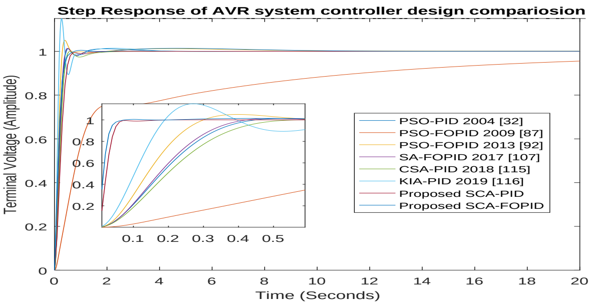


Figure 2.15: Comparison of AVR system step response with FOPID/PID controller designs

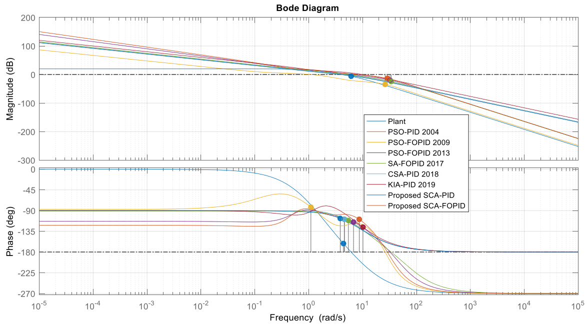


Figure 2.16: Frequency response of AVR system with different controller designs

The proposed FOPID/PID controller is designed for AVR system gives the optimal response. This optimal response is also compared with the all other controller designed for the

AVR system as shown in Figure 2.15. Therefore, it is observed that SCA based FOPID controller gives best performance for AVR system as compared to the other tuning methods of controller. The proposed controller tuning method presents good performance of reference tracking and disturbance rejection. The frequency response of the proposed SCA-FOPID controller along with the other FOPID/PID controllers are shown in Figure 2.16. The Bode plot provides the frequency response of relative stability in terms of GM & PM. The calculation of time domain & frequency domain performances are listed in Table 2.2. Then, the proposed SCA based FOPID controller has better control action and robust performance.

In order to compare the SCA based FOPID/PID controller design for AVR system to regulate the terminal voltage at the output as specified level as shown in Figure 2.17, the proposed SCA based fractional PID controller is better. The proposed SCA based FOPID/PID controllers better in performance by comparing other methods and also the controlling efforts of the controller. The controlling efforts are required for the controller to control AVR system is as shown in Figure 2.18. Also, the transient response numerical performance indices are listed in Table 2.2.

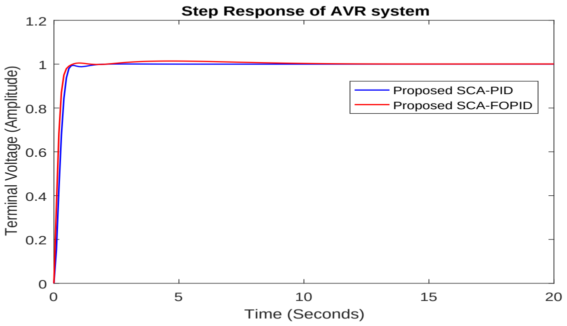


Figure 2.17: AVR system step response with SCA based FOPID/PID controller designs.

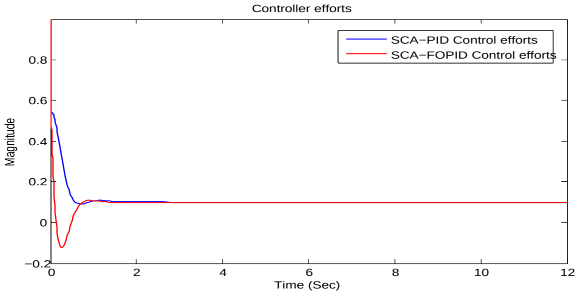


Figure 2.18: AVR system step response with SCA based FOPID/PID controller designs

Table 2.2: Controller parameters and Performance indices of the AVR system

Methods	K_P	K_I	K_D	λ	μ	T_r	$M_p(\%)$	T_s	E_{ss}	GM	PM	Objective(Eq.2.25)
No control	-	-	-	-	-	0.2633	61.96	6.238	0.0908	1.92	18.59	-
ZN-PID	0.5988	1.145	0.07828	-	-	0.3367	47.65	4.17	3.41E-06	∞	27.31	-
IMC-PID	0.2745	0.1913	0.03262	-	-	0.7282	6.42	3.157	5.24E-06	∞	61.51	-
PSO-PID 2004 [32]	0.657	0.5389	0.2458	-	-	0.2884	1.14	0.451	5.71E-09	∞	71.21	1.2303
PSO-FOPID 2009 [92]	0.17	0.03	0.014	0.97	1.38	6.2865	0	11.91	0.0792	51.84	97.46	750
PSO-FOPID 2013 [87]	1.2623	0.5526	0.2381	1.2559	1.1832	0.1603	0.01	0.2651	0	9.55	65.15	0.6248
SA-FOPID 2017 [107]	0.7837	0.5027	0.2307	1.0103	1.0727	0.2653	0.52	0.4057	5.99E-10	16.37	68.59	1.6520
CSA-PID 2018 [115]	0.6198	0.4165	0.2126	-	-	0.308	0.001	0.4261	0	∞	71.26	–
KIA-FOPID 2019 [116]	1.0426	1.0093	0.5999	-	-	0.128	15	0.753	0	∞	54.16	0.00161
Proposed SCA-PID	**0.5084**	**0.3288**	**0.1479**	-	-	**0.3935**	**0.019**	**0.665**	**9.92E-09**	∞	**72.97**	**0.03441**
Proposed SCA-FOPID	**1.5447**	**0.5691**	**0.2389**	**1.3522**	**1.3195**	**0.1549**	**0.01**	**0.2183**	**1.49E-05**	**4.65**	**71.01**	**0.04131**

2.4.3 IJA based results

To evaluate the ability of IJA-FOPID/PID controllers, a practical AVR system model [32] is considered. The physical parameters of AVR system are amplifier ($K_A = 10$ & $\tau_A = 0.1$), exciter ($K_E = 1$ & $\tau_E = 0.4$), generator ($K_G = 1$ & $\tau_G = 1$) and sensor ($K_R = 10$ & $\tau_R = 0.01$) [32, 90] as shown in the block diagram of Figure 2.19. The range of all parameters in PID controller is $[0, 1]$. The minimum and maximum limits of the FOPID controller parameters in the range are $K_p = [0,\ 25], K_i = [0,\ 15], \lambda = [0,\ 2], K_d = [0,\ 20], \mu = [0,\ 2]$. The actual terminal voltage step response of the AVR system without a controller and the classical tuning methods step responses are shown in Figure 2.20. In this case, the AVR system performance metrics without a controller are given by $T_r = 0.2633$, $M_p = 61.95$, $T_s = 6.2382$, and $E_{ss} = 0.092$ and $GM = 5.69dB$ at 6.12 rad/s, $PM = 18.6deg$ at 4.4 rad/s.

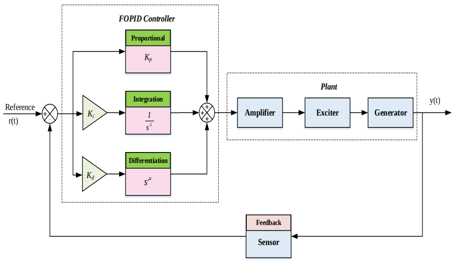


Figure 2.19: AVR system with FOPID.

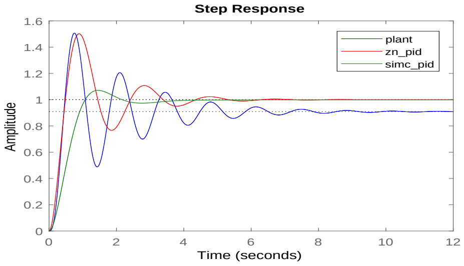
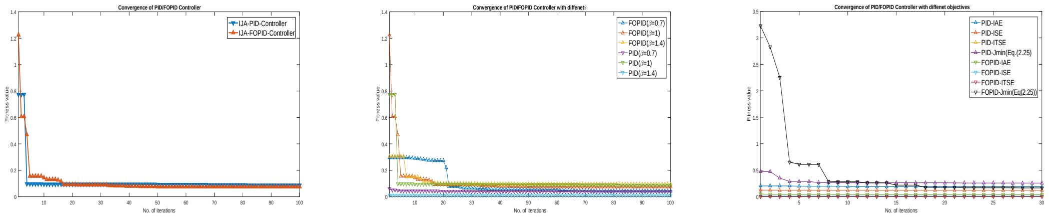


Figure 2.20: Classical PID tunings.

The proposed FOPID controller using IJA controls terminal voltage regulation of AVR system. The parameters of algorithm verifies the performance using Eq.(2.22), Eq.(2.23), Eq.(2.24) and Eq.(2.25) of the IJA-FOPID/PID controllers for the AVR system.

1. The number of tuning parameters [K_p, K_i, K_d,λ,μ] for FOPID and [K_p, K_i, K_d] for PID.
2. Min and Max limit of tuning parameters are mentioned as above section.
3. The number of solution population = 50.
4. Weighting factor (β) = 0.7 < β < 1.4.
5. The maximum number of iterations 30/50/100.

All the controller design objectives shows great convergence as shown in Figure 2.21(a), (b) and (c). The controller design of the AVR system is determined using several objective functions while the performance of FOPID/PID controller design by transient response objective having a weight factor of β=1 is found to be more efficient. The proposed method has better transient and frequency responses at the same time, where these responses are compared with existing FOPID/PID controller methods. The results show that IJA-FOPID controller has the best exploration of optimal parameters.



(a) FOPID/PID controller design

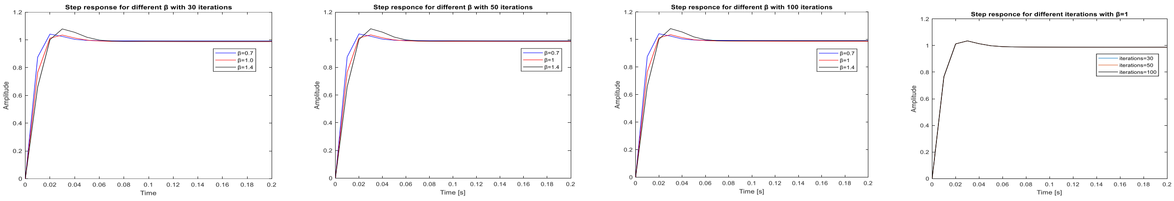
(b) FOPID/PID controller design with $\beta = 0.7, 1, 1.4$

(c) FOPID/PID controller design of different objectives

Figure 2.21: FOPID/PID controller design convergence of AVR system using IJA.

Experimental Analysis Controller Design

Here, there are many alternate arrangements of simulation or experimental setups to estimate the performance indices of each controller. In every simulation of controller tuning, a weight has been considered in the performance criterion (objective function) and different maximum number of iterations (generations) are given in Table 2.3 for IJA-FOPID and IJA-PID controllers. The simulation or experimental effects of PID controllers are obtained from the best solutions as shown in Figure 2.22(a) to Figure 2.22(d) and the simulation results of FOPID controllers are also obtained from the best solution as shown in Figure 2.23(a) to Figure 2.23(d). As observed, the FOPID controller drive to better transient response of each combination of experimental simulation results, to control the regulation voltage of AVR system. In Table 2.3, several performance criteria in the time domain response of each experimental simulation of the proposed controllers for the AVR system are demonstrated. With The controller action being estimated by four performance criteria, IJA-FOPID controller is shown to have the best controlling efforts in comparison with other controllers.



(a) Various β of 30 iterations

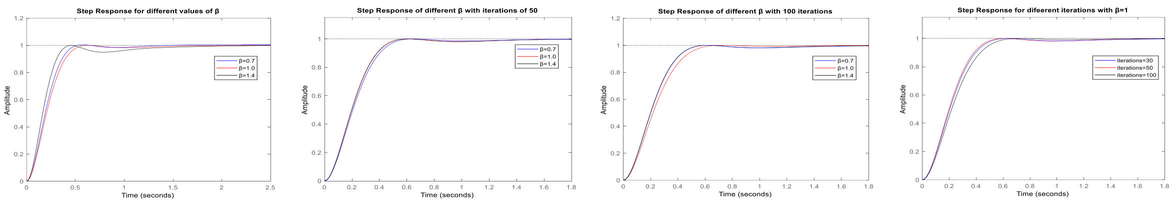
(b) Various β of 50 iterations

(c) Various β of 100 iterations

(d) Various iteration of $\beta = 1$

Figure 2.22: AVR system terminal voltage responses with FOPID controller tuning using IJA

The PID controller can also be built using the same procedure to illustrate the benefits of the proposed FOPID controller. The same estimation method is used and Table 2.3 summarizes the best solution. The terminal voltage regulation of the AVR system with the FOPID/PID controller is also shown in Figure 2.22 & Figure 2.23, respectively. The FOPID controller provides better performance when it comes to the inconsistent reaction time characteristics in



(a) Various β of 30 iterations

(b) Various β of 50 iterations

(c) Various β of 100 iterations

(d) Various iteration of $\beta = 1$

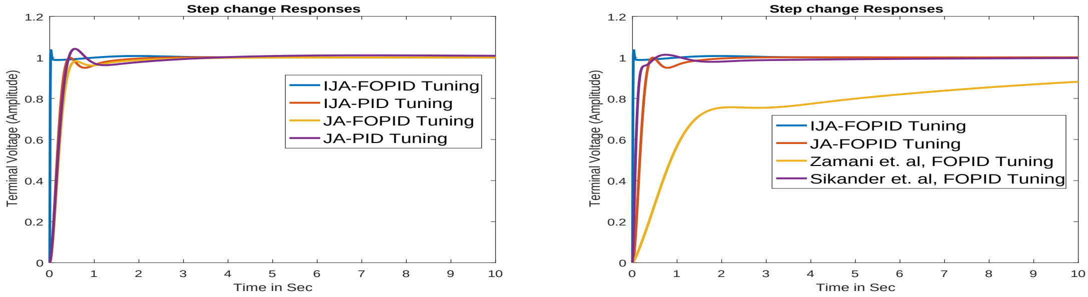
Figure 2.23: AVR system terminal voltage responses with PID controller tuning using IJA

Table 2.3: Performance of FOPID/PID controller with different simulation combinations.

Controller	it	β	K_p	K_i	λ	K_d	μ	T_r	M_p	T_s	E_{ss}	GM	PM	f_{min}
PID	30	0.7	0.6658	0.4926	1	0.2305	1	0.2964	0.4235	0.4557	0.0017	∞	69.5543	0.2923
		1	0.6364	0.4271	1	0.2059	1	0.325	0	0.5066	0.0037	∞	70.1511	0.0668
		1.4	0.7323	0.4725	1	0.2925	1	0.249	0	1.291	0.0016	∞	67.746	0.2569
	50	0.7	0.6205	0.4276	1	0.2016	1	0.3309	0	0.5171	0.0024	∞	70.369	0.0923
		1	0.6439	0.4416	1	0.2157	1	0.314	0	0.4888	0.0047	∞	69.97	0.0643
		1.4	0.6453	0.4416	1	0.2169	1	0.3179	0.0066	0.4945	0.0055	∞	69.9043	0.0430
	100	0.7	0.6409	0.4292	1	0.2109	1	0.5607	0	0.4945	0.0055	∞	69.9043	0.0877
		1	0.5845	0.4151	1	0.1852	1	0.3568	0	0.5607	0.0018	∞	71.0856	0.0749
		1.4	0.6385	0.4357	1	0.2131	1	0.3185	0	0.4959	0.0031	∞	70.0284	0.0437
FOPID	30	0.7	24.98	12.51	1.52	12.91	1.59	0.0104	4.1314	0.032	7.70E-04	∞	55.6037	0.0386
		1	24.99	14.17	1.55	6.94	1.6	0.0141	3.5047	0.0371	1.27E-04	∞	58.4195	0.0385
		1.4	24.78	11.36	0.94	9.45	1.45	0.0155	3.9369	0.0498	9.58E-05	∞	45.5797	0.0385
	50	0.7	22.81	13.24	1.25	18.17	1.72	0.0081	1.3276	0.0099	6.37E-05	∞	55.6037	0.011
		1	21.81	11.31	1.21	16.11	1.76	0.008	1.3481	0.0098	1.72E-05	∞	58.4195	0.011
		1.4	22.78	14.35	1.13	19.17	1.69	0.0081	2.016	0.0201	5.98E-05	∞	45.5797	0.011
	100	0.7	23.72	11.26	1.49	7.42	1.49	0.016	3.4431	0.0497	8.93E-07	∞	55.6037	1.60E-04
		1	22.79	12.67	1.53	6.94	1.62	0.0137	2.971	0.0348	1.59E-07	∞	58.4195	1.60E-04
		1.4	24.78	12.97	1.16	9.42	1.66	0.0108	2.0546	0.0292	7.84E-07	∞	45.5797	3.60E-05

the AVR system. However, as observed and confirmed by subsequent simulations, the FOPID controller in AVR system has a much stronger stability than the PID controller in AVR system.

IJA-FOPID controller is found to give good reference tracking, better performance and is observed to be robust. The proposed method of IJA-FOPID/PID controller design for the AVR system has been employed and this controller indicates ability of reference tracking as shown in Figure 2.24(a). To estimate the performance and controller efforts of the IJA-FOPID/PID controller, different methods are used to compare is shown in Figure 2.24(b).



(a) FOPID/PID controller tuning using IJA & JA

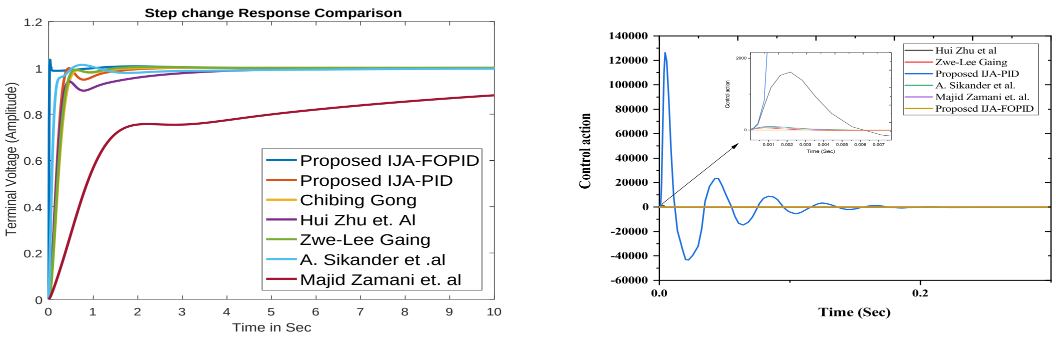
(b) FOPID tuning using IJA and other methods

Figure 2.24: Step change responses of AVR system with FOPID/PID controllers

The performance of prposed IJA-FOPID controller in AVR system has much improvement in rise time, phase margin and objective value. The basic difference between the two methods result gives a quick steady state response within a less rise time by the proposed IJA-FOPID controller in AVR system. Moreover, if the reference or steady state value is changed to other level immediatly it will get back to the steady sate value with a reported rise time of the system.

Comparison of Proposed method

The proposed JA-FOPID/PID controller is compared with other controller designs from recent state of the art of the AVR system to emphasize the superiority of the proposed method. The proposed method is compared with other controller designs for PID controller such as Zwe-Lee Gaing [32], Hui Z et.al. [75], Chibing G [117], and for FOPID controller A. Sikander et.al [90] and Majid Zamani et. al [92] as shown in Figure 2.25a. The comparison is made using the same evaluation function and algorithm parameters as mentioned above. The simulation results shows that the proposed IJA-FOPID controller is better at regulating the terminal voltage of AVR system as shown in Figure 2.25b. The proposed controller technique has reasonable controller efforts needed to maintain the desired terminal voltage of the AVR system. The transient response performance of proposed controller with comparison is also tabulated in Table 2.4.



(a) Step responses

(b) Controller actions

Figure 2.25: AVR system performances comparison

Table 2.4: Proposed controller tuning parameters value (time in Sec)

Methods	K_p	K_i	λ	K_d	μ	T_r	T_s	M_p	E_{ss}	PM	GM	f_{min}
IJA-FOPID	**21.81**	**11.31**	**1.21**	**16.11**	**1.76**	**0.008**	**1.348**	**0.009**	**1.70E-05**	**74.37**	∞	**0.3371**
IJA-PID	**0.732**	**0.472**	**1**	**0.292**	**1**	**0.249**	**0.925**	**0**	**0**	**67.74**	∞	**0.1304**
Chibing G [117]	0.60051	0.41386	1	0.20138	1	0.32259	0.5013	0.0016	0	62.16	∞	0.1786
Zhu et al.[75]	0.608	0.27	1	0.278	1	0.293	3.222	0	0	72.43	∞	0.4117
Gaing [32]	0.625	0.457	1	0.218	1	0.317	0.508	0.383	0	71.14	∞	1.4581
Sikander [90]	2.515	0.612	0.97	0.388	1.38	0.104	0.451	0	0	74.18	∞	1.1915
Zamani [92]	0.17	0.03	0.97	0.014	1.38	27	52	0	0	97.63	∞	1.5801

Computational Complexity of Algorithm

The computational complexity of controller tuning by the proposed method while evaluating the objective function to get optimal tuning parameters of FOPID/PID controller is investigated. The improved Jaya algorithm evaluates different objective functions in Eq.(2.22) to Eq.(2.25), over 10000 times evaluations, with 50 number of solution candidates and 100 number of iterations of two independent runs. The time taken for each objective function computations are listed in Table 2.5. The improved Jaya algorithm is faster to computation of optimal value of objective function as compared with GA, DE and PSO techniques. The Jaya algorithm is simple and does not have selection, crossover, mutation and other complex operations as in other evolutionary algorithms. Therefore, IJA based FOPID/PID controllers are found to be more effective and efficient than other controllers like PSO-FOPID, PSO-PID and GA-PID etc.

Table 2.5: Computational time in Sec's.

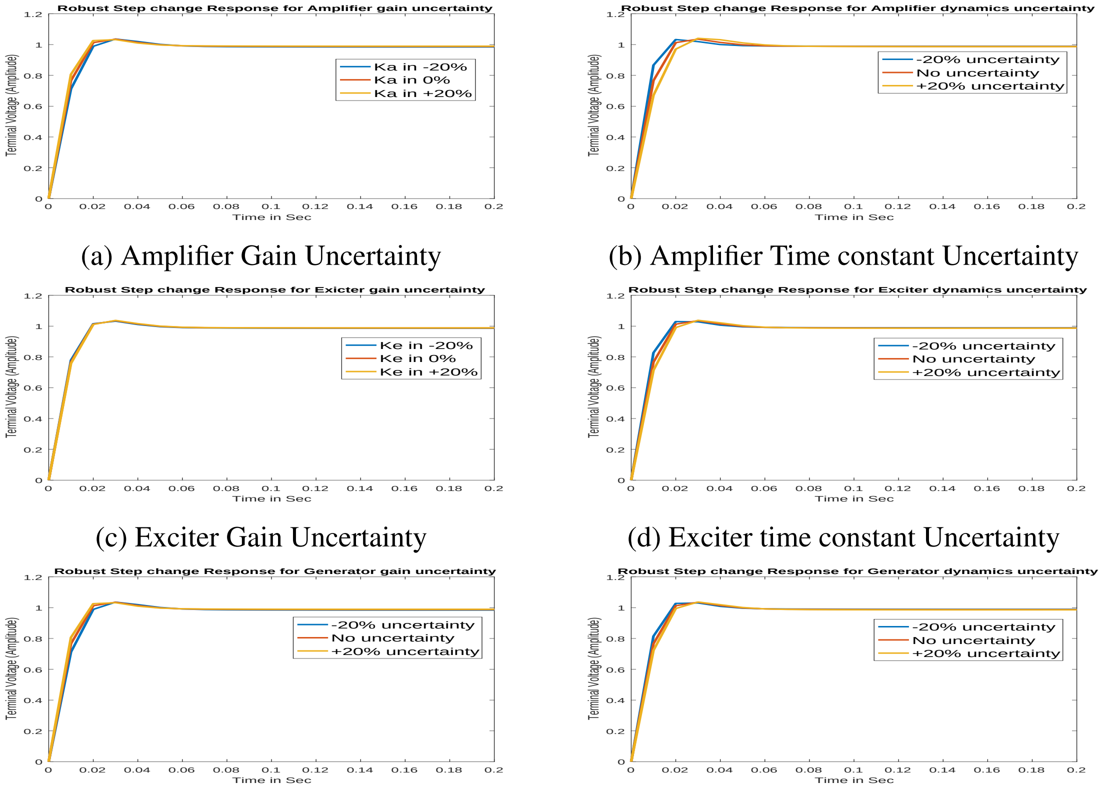
Method	IAE	ISE	ITSE	Eq.(2.25)
Fitness evolution	10000	10000	10000	10000
IJA-FOPID	117	121	119	128
IJA-PID	92	96	87	93
JA-FOPID	86	89	85	87
JA-PID	83	86	82	85

Robust Analysis

Robust analysis is performed to inspect the control system characteristics under parametric variations [117] of the AVR system. The IJA optimization algorithm is associated in the FOPID controller to attain the best (optimal) results for the evaluation of objective function.

Tuning parameters of FOPID controller are continual over a time, but the parameters acquired by the IJA optimization based on the reference tracking using error performance indices.

The AVR system dynamics can be change due to load instability, i.e. in K_A, K_E, K_G, and τ_A, τ_E, τ_G values. Hence, the robustness analysis is essential for the controller in AVR system, to simulate these changes, the parameter values of K_A, K_E, K_G, and τ_A, τ_E, τ_G are changing randomly from the -20% to +20%. These random uncertainty of AVR system can be responsible by the FOPID controller, that can provide a desired response. Therfore, the FOPID controller is robust in the AVR system for variations in dynamics, the same performances are shown in Figure 2.26.



(a) Amplifier Gain Uncertainty

(b) Amplifier Time constant Uncertainty

(c) Exciter Gain Uncertainty

(d) Exciter time constant Uncertainty

(e) Generator Gain Uncertainty

(f) Generator time constant Uncertainty

Figure 2.26: FOPID controller Robust/Uncertainty performance for AVR system

2.5 Summary

In this chapter, the FOPID/PID controller design for the SISO system is proposed based on the meta-heuristic algorithms called SCA and IJA. The proposed FOPID/PID controller algorithm is used to control the terminal voltage of the AVR system. The performance of the AVR system has been improved.

Initially, the SCA based FOPID/PID controller method is derived to improve the AVR system's performance. The FOPID/PID controller parameters are optimized using the controller design objective. Then, the proposed method's performance is validated and concluded as better in the performance of the AVR system. Further, the IJA based FOPID/PID controller is recommended for the AVR system to control the constant output terminal voltage. Here, the basic Jaya algorithm has been modified to improve the optimization performance to optimize the FOPID/PID controller parameters in the AVR system. Finally, the proposed meta-heuristic algorithm-based controller design for the AVR system of SISO system performance has been improved.

Chapter 3

Optimal FOPID/PID Controller Design for Multivariable System

In this chapter of the thesis, the systems are having many inputs and/or many outputs. Most of the systems are satisfied in the general practice of this type. It is usually the fact that the input and output can be arranged into sets and handled as if all are separated in single-input single-output (SISO) problems. Still, the unavoidable interactions can be present among the multiple inputs and outputs. Here, no option to avoid interactions but to design a controller to the specific multiple-input multi-output (MIMO) problem. This control system is called multivariable control systems, and these multivariable controls are described with standard two-input two-output (TITO) systems. This TITO system is controlled by the design of the FOPID/PID controller using different optimization algorithms.

3.1 Introduction

The processes encountered in the real world are usually multivariable (Mult-input-Multi-output (MIMO)) systems that often exhibit undesirable couplings (interactions) between the variables controlled and manipulated. A multivariable system that has strong interactions can be much harder to control. In this case, the control of decoupling using a decoupler is an effective control strategy to address the interactions between processes. However, the ideal decouplers are not always physically realizable, and the main problem with this decoupling approach is that the complexity of decoupler elements and apparent decoupling processes increases for multivariable high-dimensional systems.

The linear MIMO system can be presented using the state-space model [118]. In the MIMO state-space model, the modification adds the dimensions of inputs and outputs as a vector. In precise, the system has input vector $u(t) \in \mathbb{R}^m$ and output vector $y(t) \in \mathbb{R}^m$, then its state-space model is as following Eq.(3.1).

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{3.1a}$$

$$y(t) = Cx(t) + Du(t) \tag{3.1b}$$

where $x \in \mathbb{R}^n$ is the state vector, $x_o \in \mathbb{R}^n$ is the state vector at time t and where $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{m \times n}$ and $D \in \mathbb{R}^{m \times m}$.

As it is straightforward to convert a state-space model to a transfer function model. The Laplace transforming of Eq.(3.1) leads to as follows equations.

$$sX(s) - x(o) = AX(s) + BU(s) \tag{3.2a}$$

$$Y(s) = CX(s) + DU(s) \tag{3.2b}$$

Here, on taking $x(0) = 0$, we have that

$$Y(s) = \left(C(sI - A)^{-1}B + D\right) U(s) \tag{3.3}$$

The matrix transfer function $G(s)$ is then defined by

$$G(s) \cong C(sI - A)^{-1}B + D \tag{3.4}$$

Mention that if the unobservable and/or uncontrollable dimension is non-zero, then the $G(s)$ matrix would be an incomplete system description, the degree describes input-output features with zero initial conditions.

3.2 Multivariable System Descriptions

The general arrangement of multivariable feedback control system including decoupling is as given in Figure 3.1.

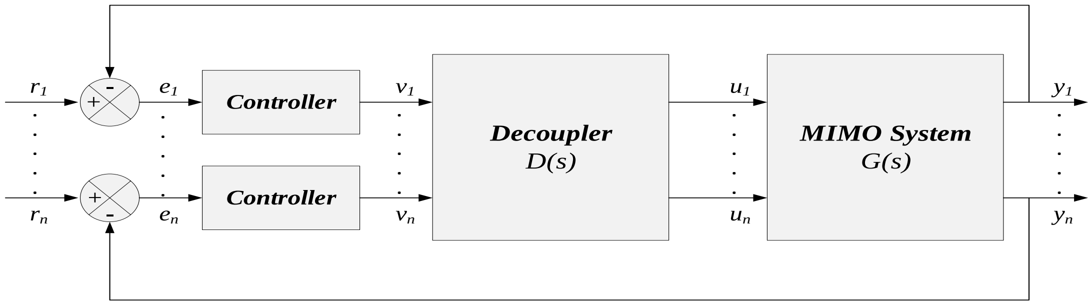


Figure 3.1: Block diagram of Multivariable closed loop system.

Description of multivariable or multiloop system plants having inputs and outputs structure is given in Figure 3.1. Here, $r = [r_1, r_2, ..., r_n]$ is the vector of setpoint inputs, $y = [y_1, y_2, ..., y_m]$ is the vector of process actual outputs, $e = [e_1, e_2, ..., e_n] = [r_1 - y_1, r_2 - y_2, ..., r_n - y_m]$ is the vector of error signals input to FOPPID controller $u = [u_1, u_2, ..., u_n]$ is vector of manipulated input signals through the FOPID controllers. Alternatively, the MIMO process $G_p(s)$ is a multivariable transfer function matrix is expressed as [60, 119] in Eq.(3.5). The multiloop control system designing is based on model of typical MIMO system. The decoupler system is use to attain SISO subsystems [120, 121]. In addition, the configuration of MIMO control system [122] with PID controllers with $n \times n$ diagonal matrix and the structure of multivariable feedback control system with decoupler as given in Figure 3.1 is defined in [123, 124] as following Eq.(3.5).

$$G_p(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) & .... & g_{1n}(s) \\ g_{21}(s) & g_{22}(s) & .... & g_{2n}(s) \\ . & . & .... & . \\ . & . & .... & . \\ g_{m1}(s) & g_{m2}(s) & .... & g_{mn}(s) \end{bmatrix} \tag{3.5}$$

where $g_{ij}(s)$ is the transfer function of a sub controlled system involving to the i^{th} process outputs and the j^{th} control inputs and also for $i = 1, 2, ..., m$ and $j = 1, 2, ..., n$.

$$D(s) = \begin{bmatrix} D_{11}(s) & D_{12}(s) & ..... & D_{1n}(s) \\ D_{21}(s) & D_{22}(s) & ..... & D_{2n}(s) \\ ..... & ..... & ..... & ..... \\ ..... & ..... & ..... & ..... \\ D_{m1}(s) & D_{m2}(s) & ..... & D_{mn}(s) \end{bmatrix} \tag{3.6}$$

To reduce such interaction properties of multivariable control in the situation of TITO system, the decoupler is used to minimize the interaction in control loops is in practice. Alternatively, in multivariable control system optimization technique is employed to reduce the interaction. Here, TITO system is considered to evaluate as said no decoupling in the feed forward direction of the control loop uses it in respective loops. In this r_1, r_2 are setpoint (reference inputs), v_1, v_2 are controller outputs or manipulated inputs to the process and y_1, y_2 are actual process outputs of the TITO process.

3.2.1 TITO System

Consider a TITO system is as given in Figure 3.2, $G_p(s)$ is the process transfer function model [119]. These multivariable systems are having interaction in between the loops (input to output also), interaction is main challenging task in the MIMO systems.

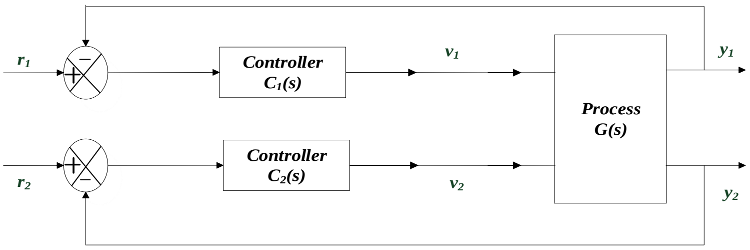


Figure 3.2: TITO system without decoupler control design.

The generalized TITO [60, 119] system model is given in Eq.(3.7)

$$G_p(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix} \tag{3.7}$$

where $g_{11}(s) = \frac{y_1(s)}{r_1(s)}e^{-\tau_{11}s}$, $g_{12}(s) = \frac{y_1(s)}{r_2(s)}e^{-\tau_{12}s}$, $g_{21}(s) = \frac{y_2(s)}{r_1(s)}e^{-\tau_{21}s}$, $g_{22}(s) = \frac{y_2(s)}{r_2(s)}e^{-\tau_{22}s}$

The choice of control loop pairing is crucial in the multiloop controller design. The Relative Gain Array (RGA) is a standardized form of the gain matrix, which estimates interactions within the inputs and outputs at steady state control loop pairing. Niderlinksi Index is also employed to decide the control loop pairing stability [125].

3.2.2 Decoupler Design

A multivariable system is demonstrated in Figure 3.1, where $G_p(s)$ is the process model transfer function matrix. The MIMO system loop pairing is based on RGA matrix is obtain and the recommended pairing was considered as $1-1/2-2$ pairing as shown in Figure 3.3. The selection of active decoupler as simplified decoupler method such that the resultant system of the combined process becomes a diagonal system as in Eq.(3.8).

$$P(s) = G_p(s) \times D(s) \tag{3.8}$$

where $P(s)$ is the desired interaction less model transfer function matrix (diagonal matrix) of the decoupled MIMO system.

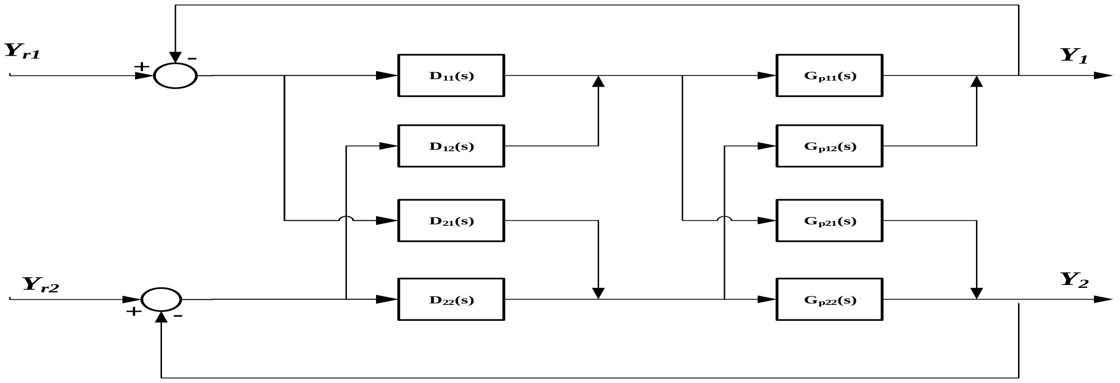


Figure 3.3: Decoupler for TITO process with 1-1/2-2 pairing.

Here, the simplified decoupling method is to minimize/eliminate the interaction between the pairing of each loop. This simplified decoupling methode is described as follows in Eq.(3.9)

$$D(s) = \begin{bmatrix} d_{11}(s) & d_{12}(s) \\ d_{21}(s) & d_{22}(s) \end{bmatrix} = \begin{bmatrix} 1 & d_{12}(s) \\ d_{21}(s) & 1 \end{bmatrix} \tag{3.9}$$

where $d_{12}(s) = \frac{-g_{12}(s)}{g_{22}(s)}$, $d_{21}(s) = \frac{-g_{21}(s)}{g_{11}(s)}$
On solving the above Eq.3.8, the resultant diagonal process system is obtained as in Eq.(3.10)

$$P(s) = \begin{bmatrix} p_{11}(s) & p_{12}(s) \\ p_{21}(s) & p_{22}(s) \end{bmatrix} = \begin{bmatrix} g_{11}(s) - \frac{g_{12}(s)g_{21}(s)}{g_{22}(s)} & 0 \\ 0 & g_{22}(s) - \frac{g_{12}(s)g_{21}(s)}{g_{11}(s)} \end{bmatrix} \tag{3.10}$$

For this resultant process system, the controller is designed as diagonal control matrix is given as in Eq.(3.11)

$$G_c(s) = \begin{bmatrix} g_{c11}(s) & 0 \\ 0 & g_{c22}(s) \end{bmatrix} \tag{3.11}$$

where $g_{c11}(s) = PID_1$, $g_{c22}(s) = PID_2$.

3.3 Methodology

Optimization problems exist everywhere in the academic and real-world applications such as in science, finance, and scientific areas. Optimization is to minimize a systems essential properties by simultaneously maximizing its desired characteristics. In general, optimization problems start whenever the resource (materials, time and cost) are limited. Specialists and experts can deal with an efficient and robust optimization is came closer to solve the problems that are essential to daily work.

3.3.1 MDE Algorithm

R. Storn and K. V. Price introduced differential Evolution (DE) in 1995 [126]. DE is a simple population-based, derivative-free function optimizer, which is enough powerful to minimize most of the stochastic functions. The major clue behind DE is a technique for creating of trial vector. The initial parental population $X_{i,0} = X_{1,i,0}, X_{2,i,0}, ..., X_{D,i,0}, i = 1, 2, ..., NP$ is arbitrarily created using standard or uniform distribution for $x_j^{low} \leq x_{j,i,0} \leq x_j^{up}$. Here, NP is the size of an initial population, D is the number of dimensions of the objective function to the problem, x_j^{low} and x_j^{up} are lower boundary & upper boundaries (search space) of the j^{th} element of the vector. After the initialization of population & parameters, DE enters the loop of mutation, crossover and selection [127] to evaluate objective function as shown in the flowchart of DE in Figure 3.4.

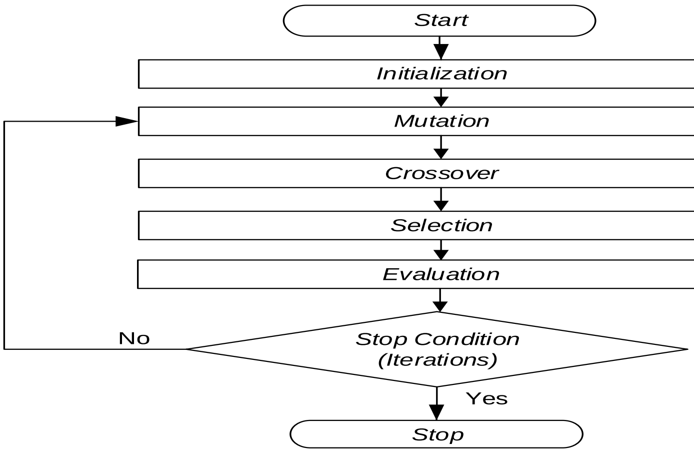


Figure 3.4: Flow chart of DE algorithm.

Mutation: In MDE, the mutant vector $v_{i,g}$ is constructed based on the present parental population for i^{th} population in generation (g). The alternative mutation schemes often used in the literature are as follows

Mutation Strategies	Donor Vector
'DE/rand/1'	$v_i = x_{r1} + F_1(x_{r2} - x_{r3})$
'DE/best/1'	$v_i = x_{best} + F_1(x_{r2} - x_{r3})$
'DE/rand-to-best/1'	$v_i = x_{r1} + F_1(x_{r2} - x_{r3}) + F_2(x_{best} - x_{r1})$
'DE/current-to-best/1'	$v_i = x_{r1} + F_1(x_{r2} - x_{r3}) + F_2(x_{best} - x_i)$
'DE/rand/2'	$v_i = x_{r1} + F_1(x_{r2} - x_{r3} + x_{r4} - x_{r5})$
'DE/best/2'	$v_i = x_{best} + F_1(x_{r2} - x_{r3} + x_{r4} - x_{r5})$

The mutation procedure generates the trial vector whose mechanism is without violation of the predefined limits and constrictions. The prospective responses to attempt this problem contain rearranging systems, penalty techniques, etc. A modest technique to establish the irreverent element to stand at the center of the disrupted limit and the resultant elements of the parental separation, i.e.

$$v_{j,i,g} = \frac{x_j^{low} + x_{j,i,g}}{2} \qquad if\ v_{j,i,g} < x_j^{low} \tag{3.12a}$$

$$v_{j,i,g} = \frac{x_j^{up} + x_{j,i,g}}{2} \qquad if\ v_{j,i,g} > x_j^{up} \tag{3.12b}$$

where $v_{j,i,g}$ & $x_{j,i,g}$ are the j^{th} element of the mutation vector $v_{i,g}$ and the parental vector $x_{i,g}$ at generation g. However, this arrangement does well exactly once the best solution is found closer or on the boundary. Here, the new mutation strategy or scheme is used to create the mutation vector as specified by Eq.(3.13) as follows.

$$v_i = x_i + F_1(x_{r1} - x_{r2} + x_{r3} - x_{r4}) + F_2(x_{best} - x_i) \tag{3.13}$$

Crossover: After creation of mutation (alteration), a 'binomial' crossover action gives the ultimate trial vector as in Eq.(3.14).

$$u_{i,g} = u_{1,i,g}, u_{2,i,g}, ..., u_{D,i,g} \tag{3.14}$$

$$\text{where } u_{j,i,g} = \begin{cases} v_{j,i,g} & if \quad rand_j(0,1) \le CR_i\ or\ j = j_{rand} \\ x_{j,i,g} & \qquad Otherwise \end{cases}$$

Here, $rand_j(a,b)$ is an identical random generated on the limits of $[a,b]$. The newly generated for every j, $j_{rand} = randinti(1,D)$ is an integer arbitrarily chosen from 1 to D and the newly generated for each i. The crossover probability (CR_i) is on the interval of $(0,1)$, approximately agree to the mean of vector elements are genetic process from the mutation vector. In classical Differential Evolution algorithm, $CR_i = CR$ is an only one parameter that is used to the creation of all the trial vectors, though in several robust DE algorithms, every specific i is related through its individual crossover probability CR_i.

Selection: The choice of selection action, chooses the improved one after the parental vector xi,g and the trial vector ui,g permitting near to their best values. For example, if it is a minimization problem, the nominated vector given in Eq.(3.15) and use as a parental vector in the following generation.

$$x_{i,g+1} = \begin{cases} u_{i,g} & if \quad f(u_{i,g}) < f(x_{i,g}) \\ x_{i,g} & \quad Otherwise \end{cases} \tag{3.15}$$

The mentioned one to one selection technique is commonly reserved in alternate DE algorithms, although the crossover may take further alternates from binomial operation in Eq.(3.13). Thus, a DE algorithm named, for example, $DE/rand/1/bin$ implying its $DE/rand/1$ mutation strategy and binomial crossover operation.

3.3.2 TLBO Algorithm

RV Rao et al proposed the Teaching Learning Based Optimization algorithm [83] in 2011. Which is analogy of the consequence of teacher impact on the outcome of the learners in a class. This algorithm terms in two phases of the learning over teacher and over interact with in a class of other learners (learner phase). A set of learners in a class along with a teacher are redirected as population, number of subjects obtained by the learners in a class are imitated as design variables and learner outcome is similar to the fitness value of the optimization problem. Initially teacher is the best solution in the population for the optimization of actual parameters or design variables. The corresponding best objective value is the best solution.

Teacher phase

In the teacher phase, teacher is the most educated or best solution for the objective. Then teacher make learners to build the mean of class to his/her level. This is depends on the learners learning efficiency of class can be expressed in Eq.(3.16)

$$X_i' = X_i + r_i(X_{i,best} - TF * M) \tag{3.16}$$

where TF is the teaching factor and TF can be either 1 or 2, this is a heuristic step to decide by arbitrarily having an equal probability as $TF = ceil[0.5 + rand]$ [128]. Then, the updated solution X_i' is established if and only if the solution is better than the previous solution by

following Eq.(3.17)

$$X_i = \begin{cases} X_i' & if \;\; f(X_i') > f(X_i) \\ X_i & if \;\; Otherwise \end{cases} \tag{3.17}$$

Learner phase

In learning phase, the learners gain knowledge not only from the teacher, but also gain knowledge by interaction with each other learners. In each generation, any two learners X_m and X_n of the class interact with each other, the smarter one enhance the others knowledge. It can be formulated in Eq.(3.18)

$$X_i' = \begin{cases} X_i + rand * (X_m - X_n) & if \quad f(X_m) > f(X_n) \\ X_i + rand * (X_n - X_m) & if \quad f(X_m) < f(X_n) \end{cases} \tag{3.18}$$

In Eq.(3.18), $(X_m - X_n)$ is taken as a step [128]. The provisionally updated result is acknowledge if and only if it is better solution than the previous solution as in following Eq.(3.19)

$$X_m = \begin{cases} X_m' & if \;\; f(X_m') > f(X_m) \\ X_m & if \;\; Otherwise \end{cases} \tag{3.19}$$

This advanced optimization algorithm involves only the basic parameters such as the number of population, number of variables and the number of generations. The algorithm does not necessity of algorithm specific controlling parameter other than *TF*. Moreover, this algorithmic controlling parameter is also does not have much effect in optimization of the problem.

3.4 Simulation Results and Discussion

3.4.1 Design of Proposed Controllers

Define a fitness function to optimize the multi-loop FOPID controller parameter gains as in Eq.(3.20)

$$J_{min}(K_p, K_i, \lambda, K_d, \mu) = \int t(e_1^2(t) + e_2^2(t))dt \tag{3.20}$$

where *min* is minimum of cost function J, and $e_1(t)$ is the first loop error function, $e_2(t)$ is the second loop error function. The SCA-FOPID controller design block diagram is as shown in Figure 3.5.

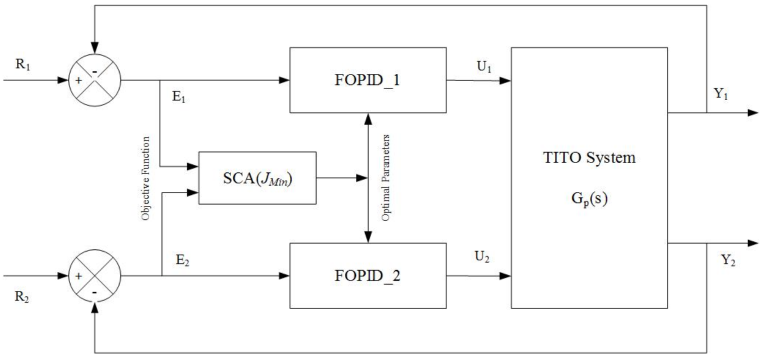


Figure 3.5: FOPID control design for TITO system.

The system originally holds a population of arbitrary selective solutions. Each possible solution designated a candidate solution. The optimization algorithm consists a group of population in an N-dimensional search space where a particular quality measure and fitness are being optimized.

In this work, a optimization approach was adopted for tuning of FOPID controller for TITO system to get optimal or close to optimal controller parameters K_p, K_i, K_d, λ and μ. In this TITO system two FOPID controllers are employed in each loop of multivariable control system. Hence, each individual solution contains ten decision members as said five parameters of each FOPID controllers respectively. To apply the algorithm initially random 'P' solutions are generated as the matrix of the initial population has $N = 10$ number of tuning parameters. The matrix dimensions are '$P \times 10$' for FOPID controller. The tuning processes of the proposed FOPID controller steps are given as follows.

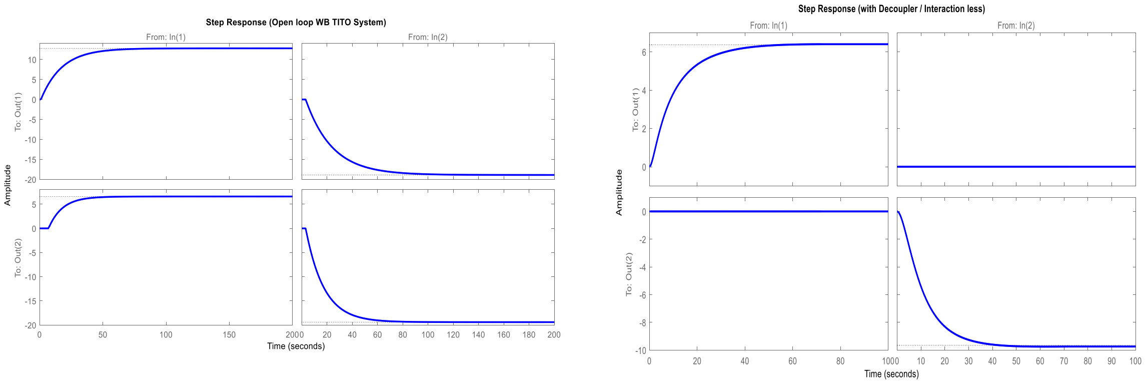
1. Initialize the number of controller tuning parameters (D) with lower & upper bounds, number of random solutions (P), and stopping criterion ($MaxIt$).
2. Obtain the functional evaluation of each individual position updates equation and controller design performance of Eq.(3.20) controller design.
3. Update the position based on the *best* solution.
4. Update the algorithmic parameters (If applicable).
5. Obtain the functional evaluation of the new solution and update the solutions.

6. Update the position of each population.
7. If the stopping criterion is satisfied as the number of iterations reaches the maximum, then go to next step 8, otherwise go to step 2.
8. Record the latest *best* solution positions as the global optimum/best solution of tuning parameters.

3.4.2 MDE based results

The WB distillation column gives the transfer function model of a MIMO system, in which the two-inputtwo-output system has been considered extensively by most of the researchers in the literature [122, 123]. The transfer function model of the WB system is given as in Eq.(3.21)

$$G_p = \begin{bmatrix} \frac{12.8e^{-s}}{16.7s+1} & \frac{-18.9e^{-3s}}{21s+1} \\ \frac{6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix} \tag{3.21}$$



(a) Open loop response.

(b) Decentralized response.

Figure 3.6: Step response of Wood-Berry TITO system.

The open-loop TITO WB system response is given in Figure 3.6a, the pairing of input and output of WB multi-loop control system, the controller design in each loop with the elements of RGA gives as $\lambda_{11} = \lambda_{22} = 2.0094$ and $\lambda_{12} = \lambda_{21} = 1.0094$, therefore the WB TITO system has 11/22 pairing to design multi-loop control system.

For this work first design the decoupler before designing a controller, so that the simplified decoupling method is used to design decoupler as follows in Eq.(3.22) using Eq.(3.9)

$$d_{12}(s) = \frac{(24.6s + 1.47)e^{-2s}}{21s + 1} \tag{3.22a}$$

$$d_{21}(s) = \frac{(4.89s + 0.34)e^{-4s}}{10.9s + 1} \tag{3.22b}$$

The delayed terms in the system are approximated using Eq.(3.23), and the resultant multivariable process is given in Eq.(3.24) as follows

$$e^{-\tau s} = \frac{1}{\tau s + 1} \tag{3.23}$$

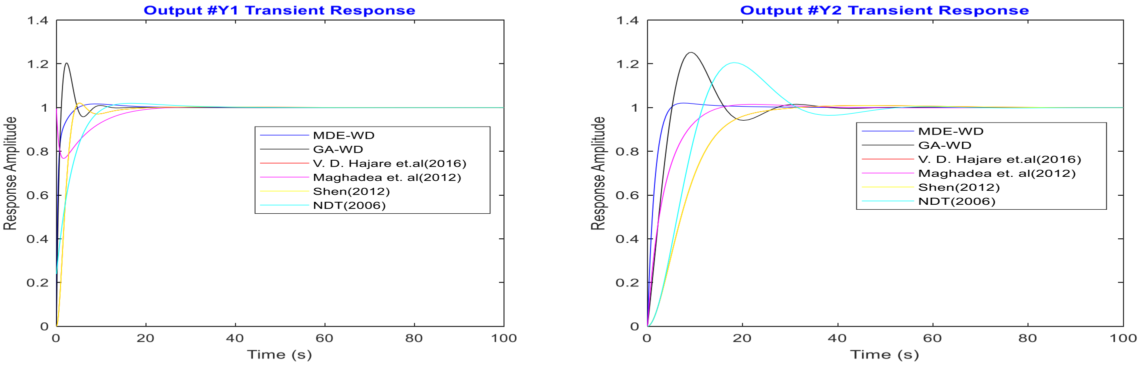
$$P_{11}(s) = \frac{1.104 \times 10^6 s^4 + 6.031 \times 10^5 s^3 + 9.538 \times 10^4 s^2 + 6026s + 123.6}{1.557 \times 10^6 s^6 + 2.609 \times 10^6 s^5 + 1.294 \times 10^6 s^4 + 2.663 \times 10^5 s^3 + 2.575 \times 10^4 s^2 + 1156s + 19.4} \tag{3.24a}$$

$$P_{22}(s) = \frac{-1.104 \times 10^6 s^4 - 6.031 \times 10^5 s^3 - 9.538 \times 10^4 s^2 - 6026s - 123.6}{2.658 \times 10^6 s^6 + 2.707 \times 10^6 s^5 + 1.035 \times 10^6 s^4 + 1.877 \times 10^5 s^3 + 1.717 \times 10^4 s^2 + 759s + 12.8} \tag{3.24b}$$

The decentralized multivariable system with minimum/elimination of interaction as shown in Figure 3.6b. Now the PID controller for each loop or as mentioned above the diagonal PID controller parameter tuning using MDE algorithm as follows in Eq.(3.25)

$$G_c(s) = \begin{bmatrix} g_{c11}(s) & 0 \\ 0 & g_{c22}(s) \end{bmatrix} = \begin{bmatrix} K_{p1} + \frac{K_{i1}}{s} + K_{d1}s & 0 \\ 0 & K_{p2} + \frac{K_{i2}}{s} + K_{d2}s \end{bmatrix} \tag{3.25}$$

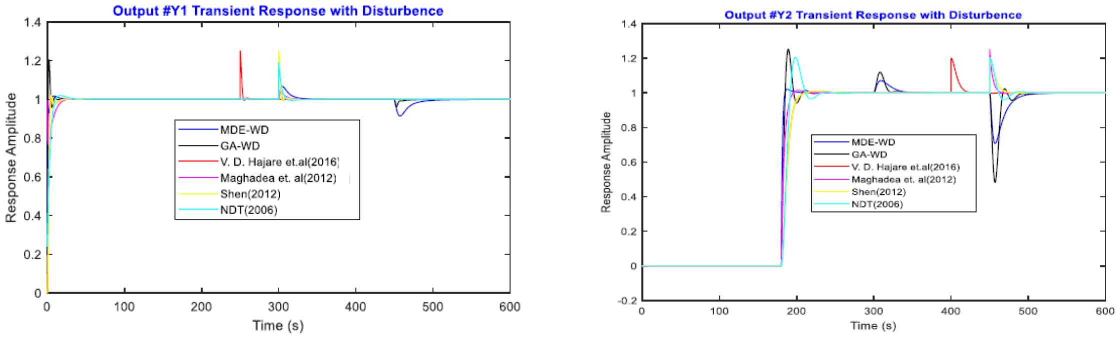
These proposed controller designs are compared with different classical, conventional methods and heuristic optimization techniques are presented in Figure 3.7 to reference tracking performance.



(a) Y1 loop response. (b) Y2 loop response.

Figure 3.7: Transient response of WB TITO system with PID controller design.

The change in step input is given to the initial reference input at time t > 0 keeping another reference input as zero for 0 < t < 180 and the changes made for another reference input at time t >180. The output responses with the disturbance in each loop at t=300 and t=450 in loop1 and loop2 respectively, as shown in the Figure 3.8 with the various design methods of PID controller.



(a) Y1 loop response.

(b) Y2 loop response.

Figure 3.8: Disturbance rejection of WB: y1 response to step in t=600s, y2 response to a step in t >180.

The performance measure values, such as Rise time (Tr), Maximum peak overshoot (Mp) and Settling time (Ts) are separately analyzed for both the top loop (subscript as 1) and bottom loop (subscript as 2) respectively as given in Table 3.1.

Table 3.1: PID controller parameters and performance of WB TITO system.

Tuning Methods	PID_1	PID_1	Rise Time(T_r)	Overshoot(M_p)	Settling Time(T_s)
	$[K_{p1}\ K_{i1}\ K_{d1}]$	$[K_{p2}\ K_{i2}\ K_{d2}]$	$[T_{r1}\ T_{r2}]$	$[M_{p1}\ M_{p2}]$	$[T_{s1}\ T_{s2}]$
MDE	**[2.657 0.427 4.301]**	**[-0.831 -0.0913 -1.575]**	**[1.41 2.89]**	**[1.66 0.06]**	**[3.43 7.59]**
GA	[3.295 2.757 3.381]	[-0.396 -0.0978 -0.858]	[0.84 4.07]	[20.4 5.27]	[7.24 24.87]
Maghadea	[0.973 0.088 2.688]	[0.313 0.0304 0.807]	[2.72 8.75]	[1.27 1.51]	[15.59 11.98]
Hajare	[0.657 0.027 4.301]	[0.312 0.013 2.175]	[2.7 13.53]	[2.08 1.01]	[11.16 22.42]
Shen	[0.0618 0.0025 0.34]	[0.02062 0.00085 0.1141]	[2.71 13.5]	[2.08 1.01]	[11.16 22.42]
NDT	[0.41 0.074 –]	[0.120 0.024 –]	[6.56 8.39]	[1.96 0.59]	[9.18 43.57]

3.4.3 TLBO based results

To confirm the validity of proposed TLBO-FOPID controller design for Wood-Berry (WB) binary distillation system. The WB distillation system [123] denotes an ideal TITO process including great interaction and significant time delays. The process holds the resulting system model in Eq.(3.26), the simulink model of the proposed system is as shown in Figure 3.9 and the step response of WB process without decoupler and without controller in Figure 3.10 as follows

$$G_p = \begin{bmatrix} \frac{12.8e^{-s}}{16.7s+1} & \frac{-18.9e^{-3s}}{21s+1} \\ \frac{6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix} \tag{3.26}$$

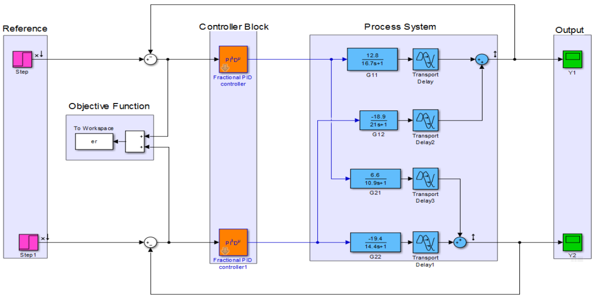


Figure 3.9: Simulink model of Wood-Berry TITO system

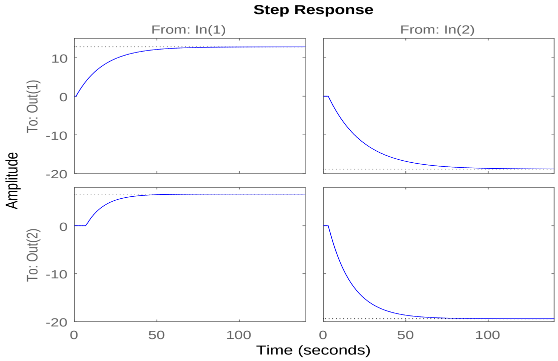


Figure 3.10: Open loop step response of Wood-Berry TITO system.

To select the pairing of the loops in multi-loop control, the 1-1/2-2 pair matching is employed using RGA matrix. As the multivariable system has a practice of decoupler to minimize the interaction in the loops. The steady-state decoupling matrix of the above multivariable plant model is given as follows in Eq.(3.27)

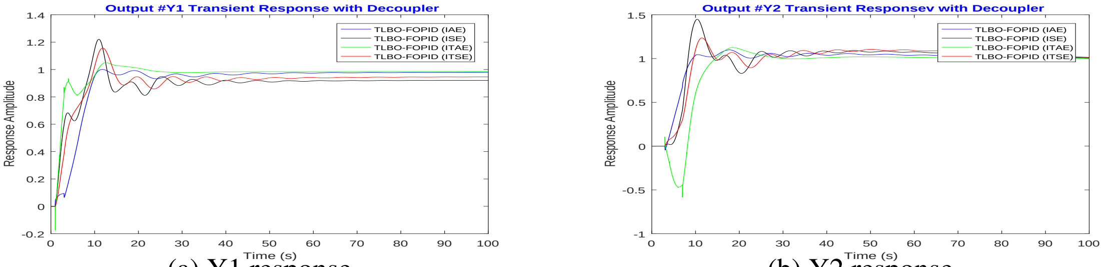
$$D_c = G_m^{-1}(0) = \begin{bmatrix} 0.157 & -0.1529 \\ 0.0534 & -0.1036 \end{bmatrix} \tag{3.27}$$

The performance comparison of the proposed FOPID based TLBO algorithm with other reported controller designs, we design the following three experiments under different conditions:

- Design of multivariable FOPID controller for Wood-Berry plant with decoupler design.
- Design of multivariable FOPID controller for Wood-Berry plant without decoupler.

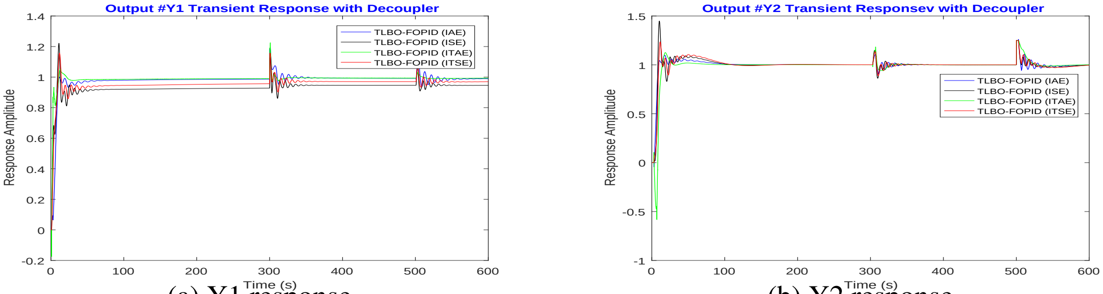
Multivariable FOPID controller with decoupler.

To illustrate the effect of decoupler in multi-loop of the wood-berry TITO system based on the TLBO algorithm tuning of multivariable FOPID controller. The following 2-input and 2-output is chosen as a benchmark example to the experiments for multivariable plant, the objective fitness function is weight coefficients are used for multi-loop controller design in the evaluation of fitness are predefined as follows: $w_1 = 0.5$ in first loop and $w_2 = 0.5$ in the second loop respectively. The four objectives are used in the design of controllers in each loop are IAE, ITAE, ISE and ITSE. The reference tracking and the disturbance rejection with decoupler design of all objectives are compared and shown in Figure 3.11 and Figure 3.12 respectively.



(a) Y1 response (b) Y2 response

Figure 3.11: Set-Point tracking of WB-TITO system with decoupler of different objective

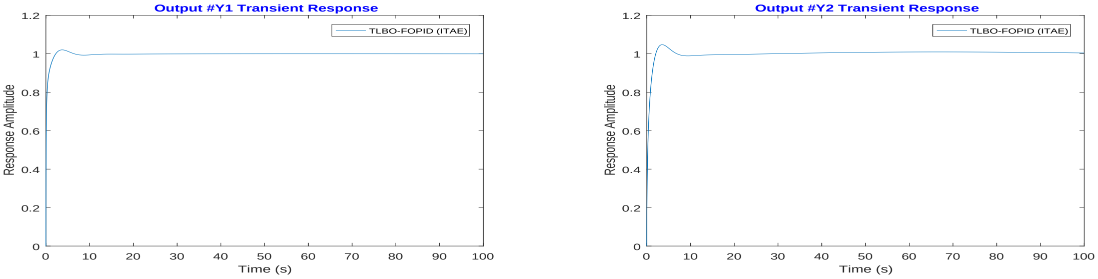


(a) Y1 response (b) Y2 response

Figure 3.12: Disturbence rejection of WB-TITO system with decoupler of different objective.

Multivariable FOPID controller without decoupler.

The proposed method is excluding the requirement of decoupler in multivariable (TITO) system, the interaction of control loops is minimum by the proposed FOPID controller tuning using standard TLBO algorithm. In this, each loop has an individual FOPID controller is used to control the corresponding output and minimization of interactions. According to this method, there are three different objectives of controller design is adopted to have minimum interaction and system reference tacking response as shown in the Figure 3.13.

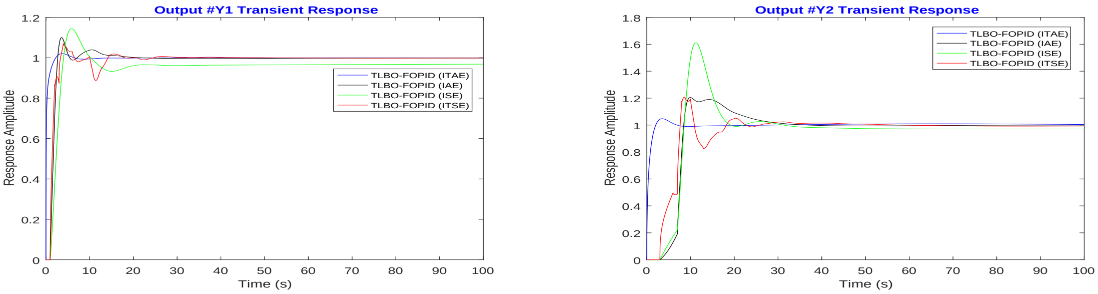


(a) FOPID-ITAE based Y1 response (b) FOPID-ITAE based Y2 response

Figure 3.13: Set-point (reference) tracking of individual loop without decoupler.

As the four different objective, such as IAE, ISE, ITSE and ITAE for FOPID controller is used, the ITAE objective will give better controller performance of $Y1$ loop from Figure 3.13 and the performance of $Y2$ loop is also shown in Figure 3.14.

To show the efficacy of to the effective optimization of FOPID controller tuning parameters by the TLBO algorithm for multivariable system. The statistical overall performance of this



(a) Y1 response

(b) FOPID-ITAE based Y2 response

Figure 3.14: Set-Point tracking of WB-TITO system without decoupler with different objectives of FOPID design.

TLBO algorithms on FOPID controller with different objective fitness functions are evaluated, the convergence of the controller design is from the worst fitness to the best fitness for both the controller at the same time with weighted objective. To illustrate the good convergence characteristic of the proposed FOPID-TLBO algorithm for WB-TITO system. Moreover, the result of convergence presents traditional optimization method of the exceptional fitness for optimal parameters in FOPID controller based on TLBO algorithm as shown in Figure 3.15.

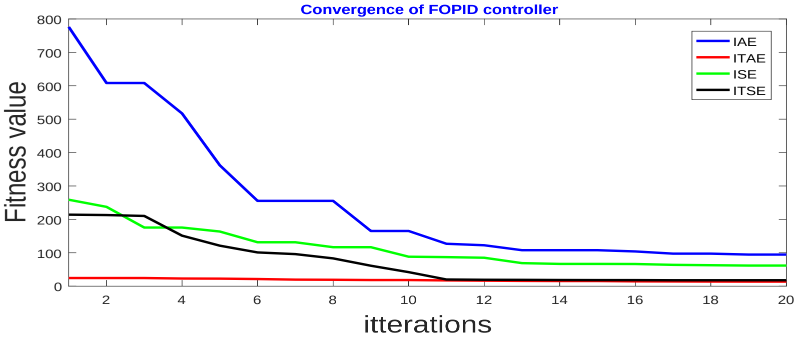
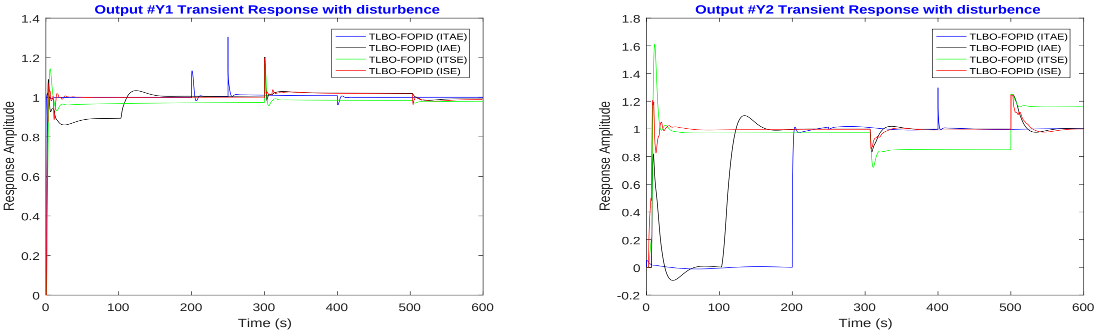


Figure 3.15: Convergence of FOPID controller with different objectives.

To determine some practical applications, the multivariable (TITO) process system has disturbances in the loops. The input step change is at $t = 0s$ and loop disturbance at time $t = 250s$ in the 1-1 loop. The step change is at $t = 200s$ and loop disturbance at $t = 400s$ in the 2-2 loop. The total simulation time is $t = 600s$. The proposed method has minimum integration effects and quick disturbance rejection ability shown in Figure 3.16.

Proposed FOPID controller performance

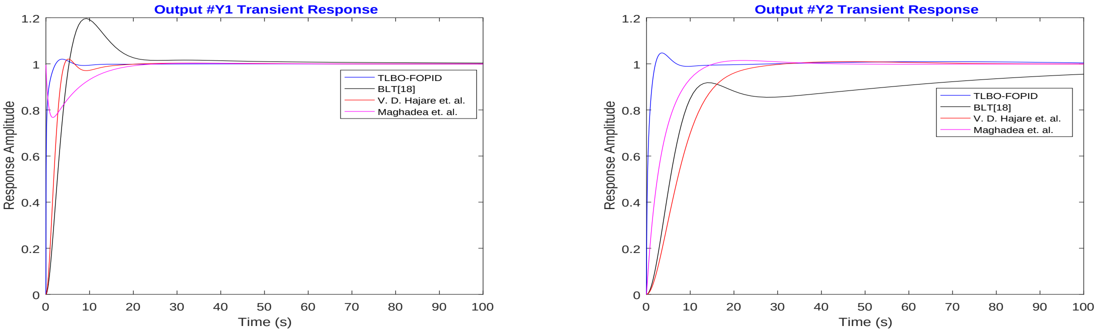
The resulting effect of the proposed method is compared with some classical, conventional methods like Maghadea et al. [60], Hajare et al. [119], BLT method [122] and [123]. The proposed optimization based TLBO-FOPID controller parameter tuning method is more superior



(a) Y1 response (b) FOPID-ITAE based Y2 response

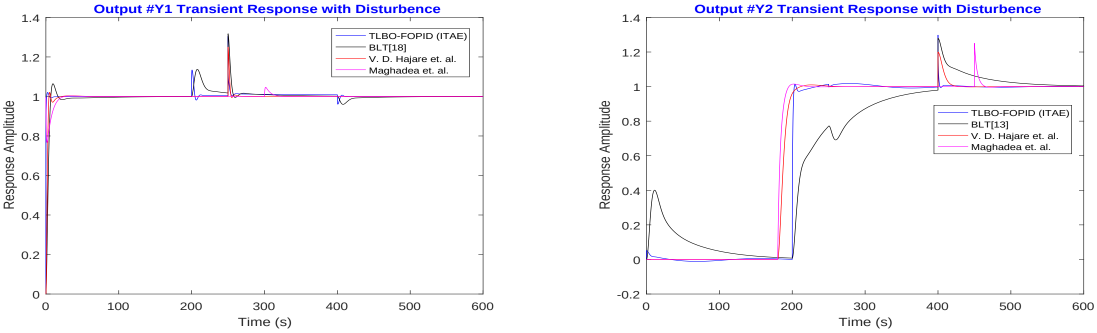
Figure 3.16: Set-Point tracking of WB-TITO system without decoupler with different objectives of FOPID design.

to the other techniques of set point tracing in Figure 3.17 and the disturbance elimination as follows in Figure 3.18.



(a) Y1 response (b) FOPID-ITAE based Y2 response

Figure 3.17: Proposed FOPID controller transient response comparisons of WB-TITO system witout decoupler.



(a) Y1 response (b) FOPID-ITAE based Y2 response

Figure 3.18: Disturbance rejection comparisons of WB-TITO system without decoupler.

The parameters of proposed FOPID controller are given in a Table 3.2. These optimal tuning parameters are chosen to give robust closed loop performance in each loop of multi-loop control system. The comparison study of the proposed scheme with some existing methods like Maghadea et al. [60], BLT method [122] and Hajare et al. [123]. are attained from the existing literature. These are tabulated for the values by simulating it for the FOPID/PID controller for a TITO process in time domain performances is listed in Table 3.3. As the performance of proposed ITAE objective method control system is better than the other objectives and literature reported multivariable control system design methods.

Table 3.2: FOPID/PID controller tuning parameters for Wood-Berry system.

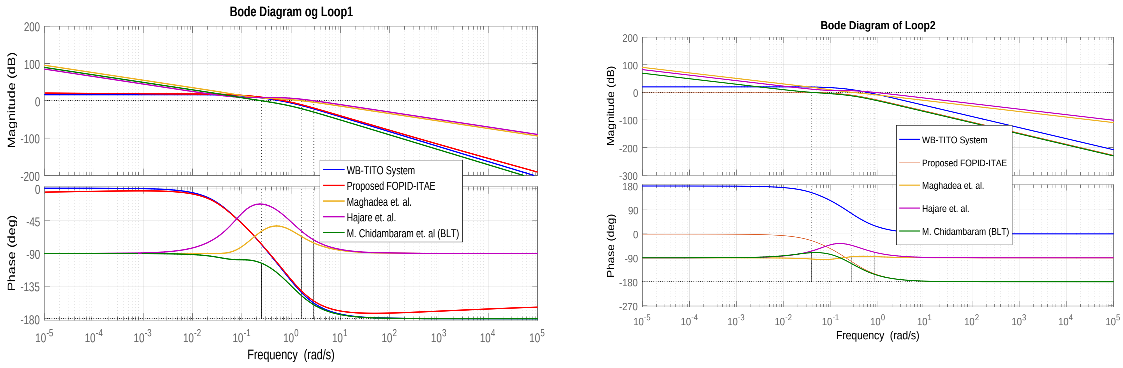
Methods	FOPID/PID in Loop1					FOPID/PID in Loop2				
	Kp	Ki	Kd	λ	μ	Kp	Ki	Kd	λ	μ
TLBO-FOPID (ITAE)	**0.3987**	**0.476**	**0.102**	**0.009**	**0.582**	**-0.009**	**-0.0098**	**-0.0078**	**0.957**	**0.495**
TLBO-FOPID (ITSE)	0.925	0.1546	0.203	0.139	0.235	-0.089	-0.0128	-0.0038	0.0519	0.0245
TLBO-FOPID (IAE)	0.328	0.091	0.147	0.01	0.042	-0.0084	-0.0096	-0.0084	0.149	0.153
TLBO-FOPID (ISE)	0.135	0.335	0.091	0.299	0.177	-0.0097	-0.00857	-0.0079	0.0086	0.0503
Maghadea [60]	0.973	0.088	2.688	1	1	-0.313	-0.034	-0.807	1	1
Hajare [123]	0.657	0.027	4.301	1	1	-0.312	-0.013	-2.175	1	1
BLT [122]	0.375	0.045	0	1	0	-0.075	-0.003	0	1	0

Table 3.3: Performance measure of Wood-Berry system in each control loop.

Methods	Performance parameters					Fitness
	[Tr1 Tr2]	[Mp1 Mp2]	[Ts1 Ts2]	[GM1 GM2]	[PM1 PM2]	
WB system (No controller)	[36.69 31.64]	[0 0]	[66.33 59.34]	[∞ -19.7]	[71.5 -138]	–
TLBO-FOPID (ITAE)	**[1.51 2.85]**	**[1.28 2.38]**	**[3.74 6.986]**	**[∞ 102]**	**[66.5 156]**	**13.48**
TLBO-FOPID (ITSE)	[2.15 4.36]	[1.83 3.53]	[5.27 15.26]	[∞ ∞]	[77.5 -138]	94.6
TLBO-FOPID (IAE)	[3.63 8.95]	[1.85 2.97]	[7.74 22.98]	[∞ ∞]	[91.6 -142]	61.72
TLBO-FOPID (ISE)	[3.86 8.86]	[2.15 6.42]	[6.85 21.53]	[16 102]	[∞ 156]	17.38
Maghadea [60]	[2.72 8.75]	[1.27 1.51]	[15.59 11.98]	[∞ ∞]	[113 94.3]	–
Hajare [123]	[2.7 13.53]	[2.08 1.01]	[11.16 22.42]	[∞ ∞]	[109 111]	–
BLT [122]	[3.74 9.57]	[16.42 NA]	[23.91 NA]	[∞ ∞]	[76.7 109]	–

Robust analysis

It has been stated that the parametric robustness of FOPID controllers is better for the multivariable control system [129]. Here, the robust analysis of the ITAE based FOPID controller is obtained by frequency response in each loop of TITO system. Literature based controller designs are compared with the proposed FOPID controller, then said that the proposed controller for WB-TITO system is robust. Figure 3.19 presents the comparison of frequency response of each loop controller. Clearly, the frequency deviations with the FOPID controller optimized by TLBO were still smaller than those by other controller design in all the cases. In other hand, ITAE based FOPID controller in multi-loop system is superior in terms of parametric robustness.



(a) Y1 loop (b) Y2 loop

Figure 3.19: Frequency response comparisons of WB-TITO system without decoupler.

Computational complexity analysis

The computational analysis of TLBO algorithm depends on the complexity of three main steps including search population size, maximum iterations and fitness function for multivariable FOPID controller. In general, the fitness function evolution (FFE) is the overall computational complexity of controller design for all optimal FOPID parameters tuning. The different objective based controller design would be differently, these computational complexity should be as possible to be minimum. The following Table 3.4 gives the different computational complexity of each objective of controller design. To investigate the computational complexity of controller tuning by the proposed TLBO-FOPID controller method while evaluating the objective function to get optimal tuning parameters, the algorithm in general over 20 iterations with 50 number of solution candidates population.

Table 3.4: Computational time in Sec.

Method	IAE	ITAE	ISE	ITSE
FOPID-TLBO	152	136	161	157
Fitness evolution	1000	1000	1000	1000

3.4.4 SCA based results

To confirm the validity of SCA based controller design in WB binary distillation column process system. The simulation example is included to estimate performance and effectiveness of the proposed SCA based FOPID controller design method.

The Wood-Berry binary distillation column process [119] denotes in Figure 3.20 is an ideal TITO process including great interaction and significant time delays. The process holds the resulting system model as in Eq.(3.28) and the step response of WB binary distillation process without decoupling and no control action step response is shown in Figure 3.21 as follows.

$$G_p(s) = \begin{bmatrix} \frac{12.8}{16.7s+1}e^{-s} & \frac{-18.9}{21s+1}e^{-3s} \\ \frac{6.6}{10.9s+1}e^{-7s} & \frac{-19.4}{14.4s+1}e^{-3s} \end{bmatrix} \tag{3.28}$$

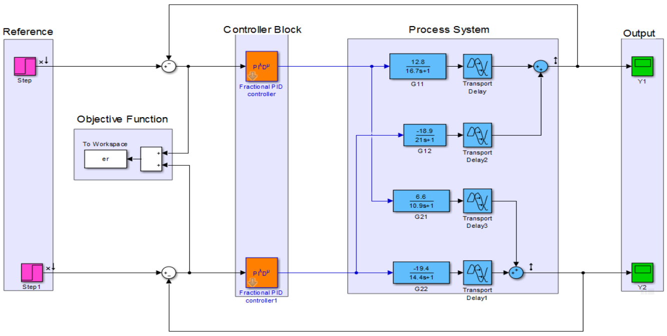


Figure 3.20: Simulink model of Wood-Berry TITO system.

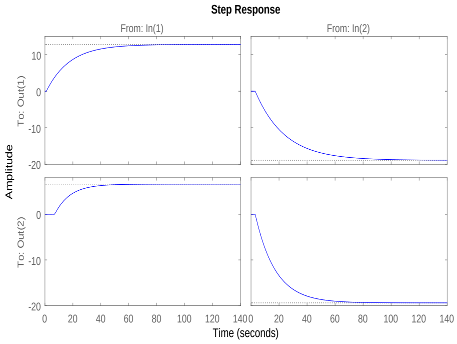


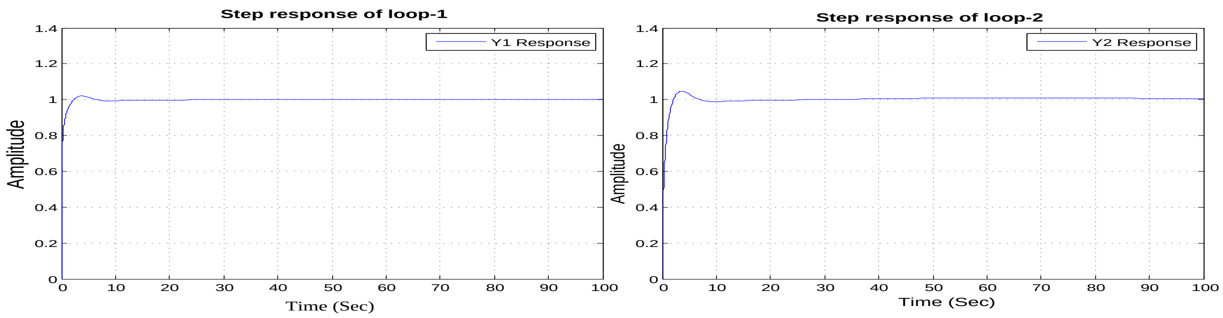
Figure 3.21: Open loop step response of Wood-Berry TITO system.

The relative gain array (Λ) matrix is obtained as follows in Eq.(3.29)

$$\Lambda = \begin{bmatrix} 2.0094 & -1.0094 \\ -1.0094 & 2.0094 \end{bmatrix} \tag{3.29}$$

According to this method, the pairing of loops is 1-1/2-2 loop pair is employed. Thus, the interaction is minimum and system set-point tacking response is much better as shown in the Figure 3.22a for Y_1 response and Figure 3.22b for Y_2 response respectively. To determine some practical applicability of proposed method, the TITO process system (Wood-Berry system) has disturbances in the loops. The loop disturbances are present at time t=250s in the 1-1 loop and t=400s in the 2-2 loop, for a total 600s of simulation time. The proposed method has quick disturbance rejection ability as shown in the Figure 3.23a for Y_1 output disturbance rejection and Figure 3.23b for Y_2 output disturbance rejection.

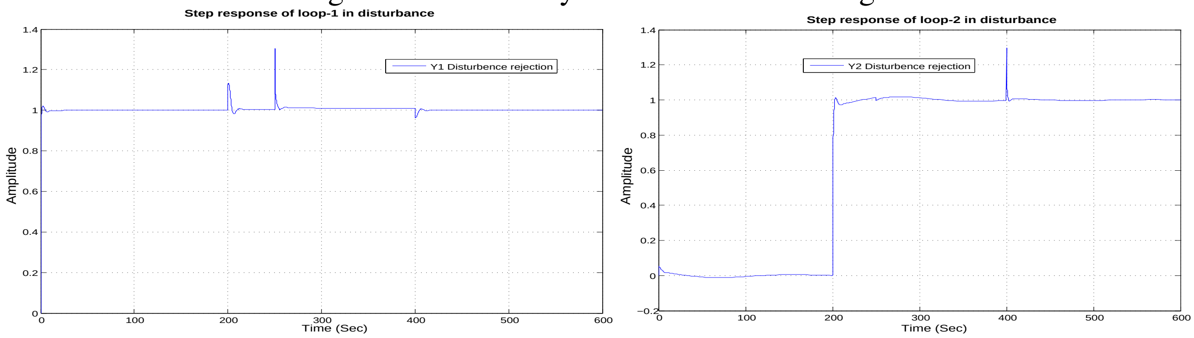
The resulting effect of the proposed method is compared with some classical, conventional methods from literature like Maghadea et al. [60], Hajare et al. [119], BLT method [122], NDT tuning method [130] and Shen et al. [131]. The proposed optimization based SCA-FOPID controller parameter tuning method of TITO system is more superior to the other techniques of set-point tracing and the disturbance elimination.



(a) 1^{st} loop of Y_1.

(b) 2^{nd} loop of Y_2.

Figure 3.22: TITO system Set-Point tracking.



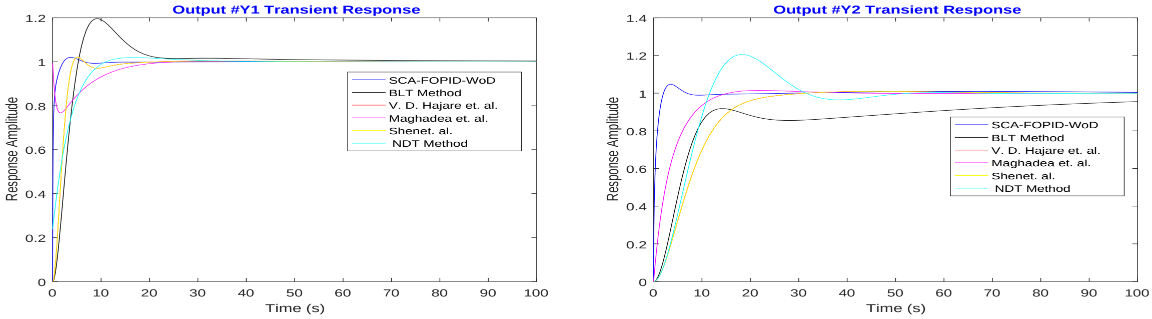
(a) 1^{st} loop of Y_1.

(b) 2^{nd} loop of Y_2.

Figure 3.23: Disturbance rejection of TITO system.

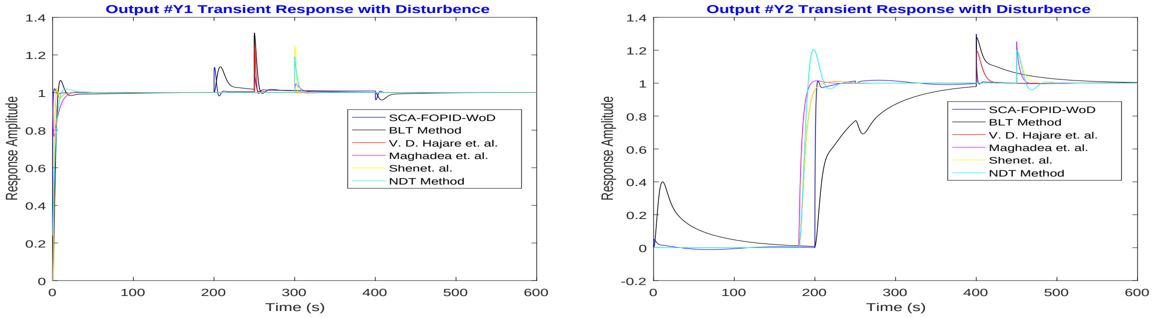
The results of proposed controller design is compared from literature controller design. As compared to this method, the interaction is minimum and system reference tacking response is optimal as shown in the Figure 3.24a for Y_1 response and Figure 3.24b for Y_2 response respectively. To estimate the practical applicability of proposed method, the TITO process system (Wood-Berry system) has disturbances in the loops. In 1-1 loop, the input step changes at time t=0s and the loop disturbance is present at time t=250s as shown in Figure 3.25a for Y_1 output. In loop 2-2, the input step changes at time t=200s and the loop disturbance is at time t=400s as shown in Fig. 3.25b for Y_2 output, for a total 600s of simulation time. The proposed method has quick reference tracking of step changes and best disturbance rejection ability as shown in the Figure 3.24 to Figure 3.25 respectively.

The control action performance in each loop given by the proposed controllers are energy efficient and possibility of actuator saturation problem. The analysis of that FOPID controller is compared with literature controllers to conclude the proposed method has best transient response performance. In the controller design, the integral square error value of each loop is used in proposed FOPID controllers to have optimal controller tuning parameters. Therefore, the saturation chances of actuators are less in case of FOPID controllers in multi-loop control systems. From reference tracking performance, disturbance rejection performance and control



(a) Y_1 Loop (b) Y_2 Loop

Figure 3.24: Comparison of TITO system set-point tracking.



(a) Y_1 Loop (b) Y_2 Loop

Figure 3.25: Comparison of TITO system disturbance rejection.

action performance, in all aspects SCA based FOPID controller is giving better performance than that of other controller. The proposed SCA based FOPID controllers has advantage of no decoupler to minimize the interaction, less overshoot and quick step change response in each loop of MIMO system.

Table 3.5: FOPID controller parameters and performance of W-B TITO system (time in Sec)

Tuning Methods	PID/FOPID_1	PID/FOPID_2	Rise Time	Overshoot	Settling Time
	[Kp1 Ki1 λ1 Kd1 μ1]	[Kp2 Ki2 λ2 Kd2 μ2]	[Tr1 Tr2]	[MP1 MP2]	[Ts1 Ts2]
SCA-FOPID	**[0.37 2.58 0.14 4.63 1.34]**	**[-2.98 -0.01 1.74 -4.16 1.27]**	**[0.705 1.21]**	**[1.62 1.57]**	**[4.284 6.186]**
BLT [122]	[0.375 0.045 1 0 1]	[-0.075 -0.003 1 0 1]	[3.74 9.57]	[16.42 00]	[23.91 NA]
Hajare [119]	[0.657 0.027 NA 4.301 NA]	[0.312 0.013 NA 2.175 NA]	[2.7 13.53]	[2.08 1.01]	[11.16 22.42]
Maghadea [60]	[0.973 0.088 NA 2.688 NA]	[0.313 0.0304 NA 0.807 NA]	[2.72 8.75]	[1.27 1.51]	[15.59 11.98]
Shen [131]	[0.0618 0.0025 NA 0.34 NA]	[0.02062 0.00085 NA 0.1141 NA]	[2.71 13.5]	[2.08 1.01]	[11.16 22.42]
NDT [130]	[0.41 0.074 NA – NA]	[0.120 0.024 NA – NA]	[6.56 8.39]	[1.96 0.59]	[9.18 43.57]

The tabulation in Table 3.5 shows the performance comparison study of the proposed scheme with some existing methods like Maghadea et al. [60], Hajare et al. [119], BLT method [122], NDT tuning method [130] and Shen et al. [131] are attained from the existing literature. These are tabulated for the values by simulating it for the FOPID/PID controller for a MIMO process in time domain performances.

The Wood-Berry TITO process system based on SCA-FOPID controller in multiloop time domain performances is shown on bar chat in Figure 3.26 as follows and it shows the proposed method time domain performances indices has optimal.

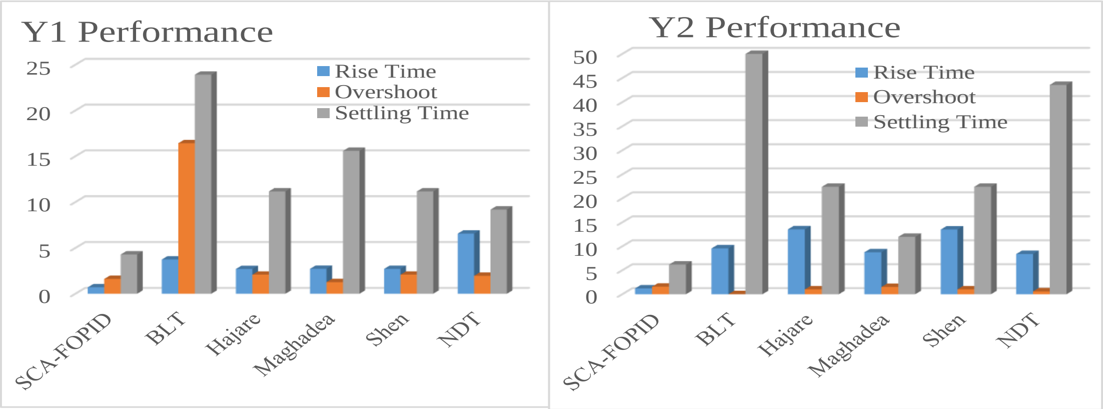


Figure 3.26: Bar chat of FOPID controller performance in each loop for TITO system.

3.5 Summary

This chapter describes the multivariable systems, the FOPID/PID controller design for the MIMO system is proposed based on the meta-heuristic algorithms called MDE, SCA, and TLBO. The proposed FOPID/PID controller algorithm is used to control the multi-loop of the MIMO control system. The performance of the multivariable system has been improved by eliminating the interaction between the loops.

Firstly, the MDE based decentralized PID controller design method is derived to improve the MIMO systems performance. Then, the proposed method's performance is validated and concluded as better in the performance of the benchmark TITO system. Further, the SCA and TLBO based centralized FOPID controller is proposed for the TITO (Wood-Berry) system to control the individual output using the multi-loop control system. Here, the MIMO system's centralized FOPID controller algorithm has been eliminating the interactions of multi-loop and the optimal controller parameters in each loop using the design objective of the FOPID/PID controller design simultaneously. However, it improved the TITO wood-berry system's performance by eliminating decoupler in the TITO control system using advanced optimization algorithms. Finally, the proposed meta-heuristic algorithm-based controller design for the TITO system performance has been improved.

Chapter 4

Hybrid MFO-WOA algorithm for global optimization and FOPID/PID Design

This chapter focuses on the proposed novel Hybrid MFO-WOA algorithm for global optimization. The Evolutionary algorithms, genetic algorithms, and tabu search are the most efficient heuristics algorithms. In certain applications, keeping a more global perspective, that is, considering the entire solution space in each iteration, might be beneficial. However, a hybrid technique combining a meta-heuristic algorithm with another optimization algorithm that can show more robust behavior and exhibit greater versatility for complex and difficult problems would improve global optimization accuracy. The proposed hybrid algorithm has been used to design FOPID/PID controller for SISO/MIMO industrial systems.

4.1 Introduction

Evolutionary computation-based methods [75] have gained more attention over the last decade from engineers struggling with issues that traditional debugging strategies cant solve. The standard genetic algorithm (GA) [69] is to attain the best states in this context, a predefined set of available random parameters associated with a process model. A current survey of evolutionary algorithms for measuring and learning algorithms [84] and control system technologies [123] is improved. In the argument, the global question of the evolutionary algorithm-based studies of the engineering problems is exercising in different design methods. In evolutionary algorithm approaches, A significant amount of work is focused between them on designing a Fuzzy Controller [132]. GA may be the fuzzy membership feature that can be used in linear rules and fine-tuning to establish information about the controlled process[133]. Subsequently, nature-propelled heuristic algorithms are built with improvement issues to succeed. GA be the first and well-known algorithm motivated by regular genetic variation and reasonable determination.

In the latest state-of-art, bio/nature-inspired optimization algorithms have received much attention in the multidisciplinary fields for the global optimum. The optimization problems are inevitable in technology & Science [134]. These optimization problems can be presented mathematically as following in Eq.(4.1)

$$\begin{aligned} minimization\ of \quad & f(x) \\ subjected\ to \quad & p_j(x) \leqslant 0, j = 1, 2, ..., m \\ & q_j(x) = 0, j = 1, 2, ..., n \end{aligned} \tag{4.1}$$

where $f(x)$ - objective function, $p_i(x)$ - inequality constrained function and $q_i(x)$ - equality constrained function.

The evolutionary algorithms, genetic algorithms, and tabu search are the most efficient heuristics algorithms. In certain applications, keeping a more global perspective, that is, considering the entire optimal solution in each iteration, might be beneficial. Furthermore, in global optimization, a hybrid optimization strategy is an effective combination of a meta-heuristic algorithm and another optimization algorithm that can exhibit more robust behavior and greater versatility when dealing with complex and difficult problems.

As of now, the hybridization of algorithms is known as a fundamental part of its groundbreaking algorithm. A unique algorithm can not succeed in an ideal arrangement for a reasonable time [135]. Moreover, the improper assembly in unique algorithms is inexorable. Along with these sets, the first algorithm quite often significant than the other hybridization algorithm. As far as hybridization is concerned, this is the first instance when the hybridization among moth-Flame optimization (MFO) [136] and whale optimization algorithm (WOA) [137] has proposed for taking care of advancement issues and search boundaries. This hybridization is expected to improve the first MFO and to manage with the interferences of WOA execution.

4.2 Current Algorithms

The proposed algorithm is to build up a combination of two algorithms, based on the advantage of the individual algorithm. These algorithms are depicted in detail in the accompanying subsections.

4.2.1 Moth-Flame optimization

In the MFO algorithm [136] is expected that the up-and-comer arrangements are moths and the factors of the issue are the situation of moths in the room. The moths in 1-D can commute along those same lines to modify 2-D, 3-D, or hyper-dimensional terms among their position vectors.

After the calculation of the MFO is a population-based algorithm, the moth's agreement in a grid as demonstrated as:

$$M = \begin{bmatrix} m_{1,1} & m_{1,2} & \dots & \dots & m_{1,d} \\ m_{2,1} & m_{2,2} & \dots & \dots & m_{2,d} \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ m_{n,1} & m_{n,2} & \dots & \dots & m_{n,d} \end{bmatrix} \tag{4.2}$$

where n is moth number and d is the variable number. Assume that, for all moths, there is an array to store the respective fitness values as follows:

$$OM = \begin{bmatrix} OM_1 \\ OM_2 \\ . \\ . \\ . \\ OM_n \end{bmatrix} \tag{4.3}$$

where n is the moths number. Consider that for each moth, and the fitness value is the returned value of the (objective) fitness function. The beast position vector (for example, first row in matrix M) is passed on to the fitness function. The fitness function's performance is assigned to the corresponding moth as its fitness value (for example, OM1 in the matrix of OM). The other critical components in the proposed algorithm are flames that also a matrix similar to the moth matrix is considered as follows:

$$F = \begin{bmatrix} f_{1,1} & f_{1,2} & \cdots & \cdots & f_{1,d} \\ f_{2,1} & f_{2,2} & \cdots & \cdots & f_{2,d} \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ f_{n,1} & f_{n,2} & \cdots & \cdots & f_{n,d} \end{bmatrix} \tag{4.4}$$

where n is the moth's number and d is the variables number. It could be seen that the M and F arrays are identical in size. For the flames, a collection for storing the related fitness values is often assumed to be as follows:

$$OF = \begin{bmatrix} OF_1 \\ OF_2 \\ . \\ . \\ . \\ OF_n \end{bmatrix} \tag{4.5}$$

where n is the moths number. Here, the flames and moths are both alternatives for the difference with them is the way we treat and change at every iteration. The moths are standard search agents traveling around the search space, while flames are the best location of moths that have been obtained so far. In all other words, the flames can be regarded as flags or pins dropped by moths while exploring the quest room. Every moth also looks for a flag (flame) and if a better solution is found, upgrades it. With this process, one moth never loses its very best solution.

The MFO algorithm is a three-tuple algorithm [136] that approximates the global optimum of optimization problems and defines it as:

$$MFO = (I, P, T) \tag{4.6}$$

Here, I is a generic function of a random population of moths and the corresponding fitness value. The systematic modeling of this function is as follows in Eq.(4.7):

$$I : \varnothing \rightarrow \{M, OM\} \tag{4.7}$$

The P is a principal function, which affects the moths throughout the search space. This function can use the model of M and eventually return the updated one by the Eq.(4.8).

$$P : M \rightarrow OM \tag{4.8}$$

The T function replies true if the terminus rule is satisfied and false if the terminus rule is not satisfied as in Eq.(4.9):

$$T : M \rightarrow \{true, false\} \tag{4.9}$$

with I, P and T are the overall frames of the MFO algorithm is described as follows:

The function I has to create primary solutions and estimate the objective function values. A random pattern can be used in this function and the resulting method is utilized as follows:

As it noticed, there are two additional arrays termed *min* and *max*. The other forms specify the minimum and maximum limits of the variables as:

$$min = [min_1, min_2, min_3, ..., min_{n-1}, min_n] \tag{4.10}$$

where min_i indicates the maximum limit of the $i - th$ variable

$$max = [max_1, max_2, max_3, ..., max_{n-1}, max_n] \tag{4.11}$$

where max_i denotes the minimum limit of the $i - th$ variable.

The P function goes iterative process after initialization, till the T function delivers accurate. The function P is the primary function that drives the moths across the exploration. As stated earlier, the particular dimension is the basis for this algorithm. The mathematical model of each moth's position in respect of a flame is modified utilizing the corresponding Eq.(4.12):

$$M_i = S(M_i, F_j) \tag{4.12}$$

where M_i specify the $i - th$ moth, F_j describe the $j - th$ flame, and S is the spiral function.

Moreover, a logarithmic spiral is applied because of the significant update process for moths. However, anyone of the spiral pattern can be used, according to the following guidelines:

- The Spirals beginning point should begin from the moth.

- The Spirals concluding point should signify the location of the flame.
- The inconstancy of the Spiral range should not exceed from the exploration.

By considering those limits, a logarithmic spiral is determined for the MFO algorithm is as following Eq.(4.13):

$$S(M_i, F_j) = D_i \times e^{bt} \times cos(2\pi t) + F_j \tag{4.13}$$

where D_i designates the distance of $i-th$ moth at $j-th$ flame, b is a constant for the framework of the logarithmic spiral, and t is a arbitrary number in the range [-1,1].

Here, D is computed by the Eq.(4.14):

$$D_i = \left|F_j - M_i\right| \tag{4.14}$$

where M_i indicates the $i-th$ moth, F_j is describe the $j-th$ flame, and D_i handle the gap of $i-th$ moth from the $j-th$ flame.

The additional exploitation [139], t is described as an arbitrary number mostly in the range [r, 1], wherever r is decreased by linearly from 1 to 2 throughout all the iterations. Under Eq.(4.15), every moth is restricted to a flame, which may result in the most significant limited stagnation. To overcome this, a list of the flame needs to be updated and organized at each iteration by their fitness values. The moths change their views of their corresponding flames eventually. Since, they spot where moths are modified in respect of n several positions in the search space would demote the exploitation of the most assuring solution, an adaptive method for the flame number has continued as given in the Eq.(4.15):

$$flameno = round\left((N - l) \times \frac{N-1}{T}\right) \tag{4.15}$$

where l is the current iteration, N is the maximum flame number, and T indicates the maximum iterations number.

4.2.2 Whale optimization algorithm

The Whale optimization algorithm (WOA) [137], this algorithm concept is acquired by the nature of hunting mechanism for humpback whales, imitates the shrinking encircling, spiral update location and arbitrary hunting mechanisms for the humpback whale. The algorithm consists of 3-phases [137]: prey encircling, bubbling-net attacking and prey checking.

Encircling Prey

The WOA algorithm [137] allows the target prey is the present optimal solution for encircling prey. The individual whales then strive to adjust to the most desirable location. The next Eqs.(4.16,4.17) illustrate the action of behavior:

$$\overrightarrow{D} = \left|\overrightarrow{C} \cdot \overrightarrow{X^*}(t) - \overrightarrow{X}(t)\right| \tag{4.16}$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X^*}(t) - \overrightarrow{A} \cdot \overrightarrow{D} \tag{4.17}$$

where t represents the existing iteration, $\overrightarrow{X}(t)$ is the location vector, $\overrightarrow{X^*}(t)$ is the position vector of the complete results reach so far, $||$ is the absolute value, and $\cdot$ is an product of element-by-element. $\overrightarrow{A}$ and $\overrightarrow{C}$ are coefficient computation vectors are defined by Eq.(4.18) and Eq.(4.19):

$$\overrightarrow{A} = 2\overrightarrow{a}\,\overrightarrow{r} - \overrightarrow{a} \tag{4.18}$$

$$\overrightarrow{C} = 2\overrightarrow{r} \tag{4.19}$$

where $\overrightarrow{a}$ is linearly decreased from 2 to 0 over the progression of iterations (in both exploration and exploitation) and $\overrightarrow{r}$ is a arbitrary vector in $[0, 1]$.

Bubble-Net Attacking Method

The function is accomplished primarily by lowering the control parameter value a. $\overrightarrow{A}$ in the interval $[a, a]$, a is random number. The new location of the specific whales can be specified wherever among the first place and the contemporary best position when arbitrary values for $\overrightarrow{A}$ are in the range $[-1, 1]$.

A spiral equation is then created among the whale and prey position to imitate the helix-shaped inclination of humpback whales as follows in Eq.(4.20), Eq.(4.21):

$$\overrightarrow{X'}(t+1) = \overrightarrow{D'} \cdot e^{bl} \cdot cos(2\pi l) + \overrightarrow{X^*}(t) \tag{4.20}$$

$$\overrightarrow{D'} = \left|\overrightarrow{X^*}(t) - \overrightarrow{X}(t)\right| \tag{4.21}$$

where b for determining the appearance of the logarithmic spiral, l is an arbitrary number in the range $[1, 1]$, and $\cdot$ is a multiplication of element-by-element.

When humpback whales strike their prey, they travel along a spiral-shaped path concurrently within a diminishing surrounding circle. The concept of mathematics is as follows in Eq.(4.22):

$$\overrightarrow{X^*}(t+1) = \begin{cases} \overrightarrow{X^*}(t) - \overrightarrow{A} \cdot \overrightarrow{D} & if\ p < 0.5 \\ \overrightarrow{D'} \cdot e^{bl} \cdot cos(2\pi l) + \overrightarrow{X^*}(t) & if\ p \geq 0.5 \end{cases} \tag{4.22}$$

Search for Prey (Exploration Phase)

Humpback whales hunt randomly by each other's location. During the hunt toward the food or investigation process, the other individual whale locations will be changed according to a particular whale selection at arbitrary. To discover better food, by establishing $\overrightarrow{A} > 1$, unique whales are driven to run away from the reference whales. This way, the WOA algorithm can do global searching. The model of maths can be expressed as in Eqs.(4.23), Eq.(4.24):

$$\overrightarrow{D} = \left| \overrightarrow{C} \overrightarrow{X_{rand}}(t) - \overrightarrow{X}(t) \right| \tag{4.23}$$

$$\overrightarrow{X^*}(t+1) = \overrightarrow{X_{rand}}(t) - \overrightarrow{A} \cdot \overrightarrow{D} \tag{4.24}$$

where $\overrightarrow{X_{rand}}$ is a position vector randomly selected from the current whale population.

4.3 Proposed hybrid MFO-WOA algorithm

To enhance the application and exploration capabilities of existing algorithms, some researchers are currently proposing multiple approaches to integrated optimization. According to Talbi et al. [140], two types of hybridization i.e., at a high level or low level with evolutionary or relay methods as symmetric or asymmetric.

4.3.1 Hybrid MFO-WOA algorithm

Hybridization of the MFO algorithm with a WOA using a high-level evolutionary hybridization [140] in this paper. To show in hybridization strategies is to develop the exploration skills at the MFO and the health of extraction at WOA. In this paper to cover a wide variety of unknown search spaces due to issues with the logarithmic spiral. The stage of exploration signifies the intensity of variability to move through a large number of possible outcomes. The search agent's position is responsible for the problem being decided globally optimally.

The mainframe of the hybrid MFO-WOA algorithm, the MFO focuses diversity at the starting of the search for a significant step in expanding the search space effectively and preventing trapping into a local minimum. In contrast, the WOA focuses on reinforcement and allows individuals to force towards the best solution of the optimization. This procedure can also cope with topical as well as local searches. It assesses each vector's fitness and obtains the best solution to be applied by MFO as the destination point position in Eq.(4.12), Eq.(4.13). Whereas, WOA is to improve the worst results over the iterations obtained using MFO. Using Eq.(4.12), Eq.(4.13), positions of each individual in the MFO is modified over a series of iterations and then update to new positions by the Eq.(4.20), Eq.(4.21), Eq.(4.22), respectively.

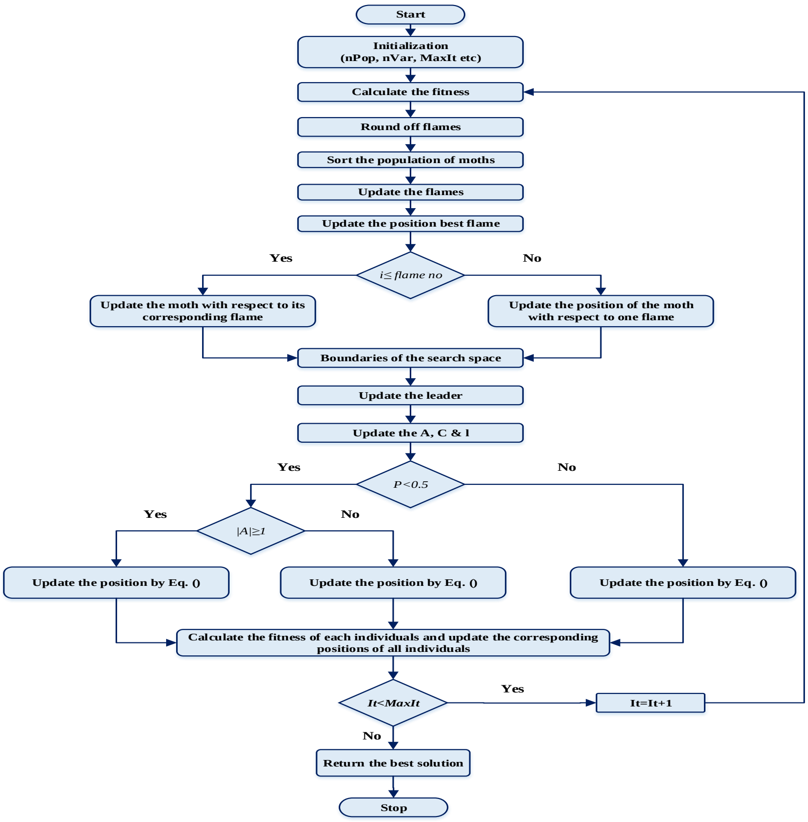

Start
Initialization
(nPop, nVar, MaxIt etc)
Calculate the fitness
Round off flames
Sort the population of moths
Update the flames
Update the position best flame
Yes
No
i≤ flame no
Update the moth with respect to its corresponding flame
Update the position of the moth with respect to one flame
Boundaries of the search space
Update the leader
Update the A, C & l
Yes
No
P<0.5
Yes
No
|A|≥1
Update the position by Eq. ()
Update the position by Eq. ()
Update the position by Eq. ()
Calculate the fitness of each individuals and update the corresponding positions of all individuals
It<MaxIt
Yes
It=It+1
No
Return the best solution
Stop

Figure 4.1: Flow-Chart of proposed MFO-WOA algorithm

The steps of the suggested MFO-WOA hybrid algorithm is described as follows. The detailed pseudo-code in Algorithm 29 and the proposed hybrid MFO-WOA algorithm flow chart is given in Figure 4.1.

Algoritmo 1: Proposed MFO-WOA algorithm

```
Initialize a set of population with random positions for dimensions ($D$) in the search
  space using $X_{i,j} = min_j + rand(max_j - min_j)$
Initialize algorithmic parameters
while Current iteration < Maximum iterations do
    read current
    for each population i do
        Evaluate the fitness value for all
        Update the position of each population by using
          $X_i = S(M_i, F_j) = D_i e^{bt} cos(2\pi t) + F_j$
    end
    Identify the leader in the population
    Update the parameteres $(A, C, I\&p)$
    for each population i do
        Update the position of each population
        if $p > 0.5$ then
            $\overrightarrow{X^*}(t+1) = \overrightarrow{X^*}(t) + \overrightarrow{D} \cdot e^{bl} \cdot cos(2\pi l)$
        else
            if $|A| \leqslant 1$ then
                $\overrightarrow{X^*}(t+1) = \overrightarrow{X^*}(t) - \overrightarrow{A} \cdot \overrightarrow{D}$
            else
                $\overrightarrow{X^*_{rand}}(t+1) = \overrightarrow{X^*_{rand}}(t) - \overrightarrow{A} \cdot \overrightarrow{D}$
            end
        end
    end
    Verify the search space boundary conditions
    if $f(\overrightarrow{X^*}(t+1)) < f(\overrightarrow{X^*}(t))$ then
        Update the new positions of all populations
    end
    Update the best position of better population
end
Return the best/global optimal solution
```

- *Step 1:* Initialize the algorithm parameters, such as maximum iteration or generation (*it*), number of search agents (*N*), A, C, p. The initial population is defined as follows:
  $X_{i,j} = min_j + rand(max_j - min_j)$
  where min_j and max_j are the lower and upper bounds of the variables of a search space, *rand* is a random number uniformly between [0 1].

- *Step 2:* Evaluate the fitness of all candidates of X, i.e., to estimate the fitness value of X for all N.

- *Step 3:* An adaptive mechanism for the number of flames are uses by the following equation.
  $flame\ no = round(N - tt \cdot \frac{N-1}{T})$.

- *Step 4:* For every moths, the associated fitness values are included in an array.

- *Step 5:* Update r and t values are obtain by the the following equation. $r = -1 + tt * (\frac{-1}{T})$, $t = (r - 1) \cdot rand + 1$
  where tt is current iteration, T is maximum number of iterations and *rand* random number from [0 1].

- *Step 6:* Calculate D using the equation for the corresponding moth. $D_i = \left|F_j - M_i\right|$

- *Step 7:* Update the position of all solution candidates using the corresponding moth.
  $M(i, j) = S(M_i, F_j) = D_i \times e^{bt} \times cos(2\pi t) + F_j$

- *Step 8:* Update (Find) the leader in search agents.

- *Step 9:* Update the values of algorithmic parameters are a, A, C, r and p.

- *Step 10:* Update the position of the current search agent by the following equation.
  $$\overrightarrow{X^*}(t+1) = \begin{cases} \overrightarrow{X^*}(t) - \overrightarrow{A} \cdot \overrightarrow{D} & if\ p < 0.5 \\ \overrightarrow{X^*}(t) + \overrightarrow{D'} \cdot e^{bl} \cdot cos(2\pi l) & if\ p \geq 0.5 \end{cases}$$

- *Step 11:* if $|A| > 1$ then, select a random search agent (X_{rand}), and update the current search agent's position by the following equation.
  $\overrightarrow{X^*}(t+1) = \overrightarrow{X_{rand}}(t) - \overrightarrow{A} \cdot \overrightarrow{D}$
  where $\overrightarrow{D} = \left|\overrightarrow{C}\overrightarrow{X_{rand}}(t) - \overrightarrow{X}(t)\right|$

- *Step 12:*Verify the search space boundaries as all the positions of individual within the defined space.

- *Step 13:* Evaluate the fitness of new search agents.

- *Step 14:* If $f(X_{new}) < f(X)$ then, update the positions of all the search agents population.

- *Step 15:* Update the best position of a better agent.

- *Step 16:* If the termination condition is satisfied, record the global solution. Otherwise, go to step 2.

4.3.2 Convergence analysis

The proposed hybrid MFOWOA algorithm is assessed in 23 ideal benchmark test functions [141], all of which are features of minimization by many researchers in their recent studies. The substantial analysis, hybrid MFO-WOA algorithm is evaluated over $NP = 50$ & 100 for 1000 iterations in independent runs of 30.

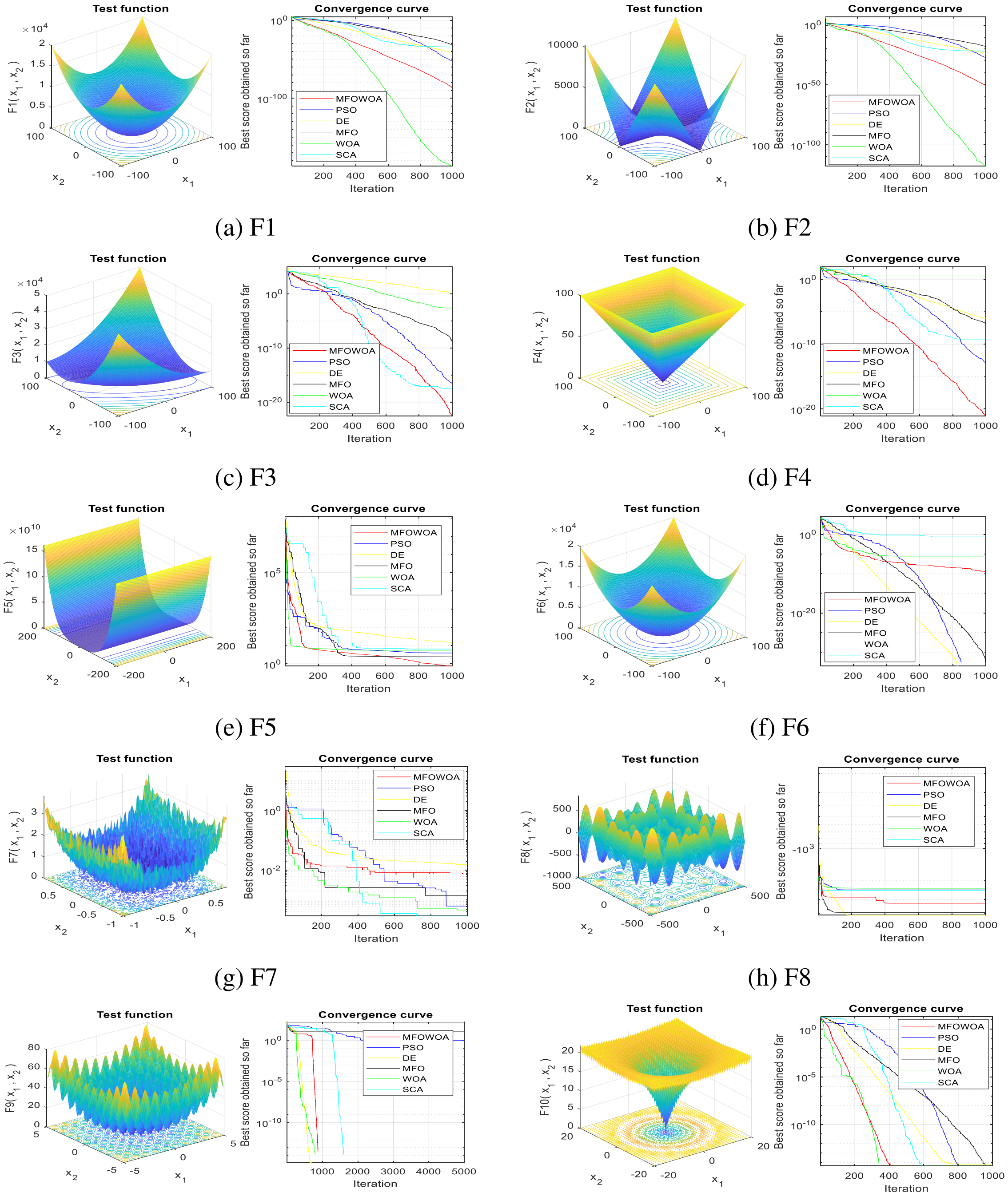
The benchmark functions of F1-F23, the investigation study is recommended by taking the population size of $NP = 50$ & 100 for each algorithm. The range of the multi-modal and unimodal benchmark functions is set at 30. For all algorithms, the criterion for stop condition is the maximum number of function evaluations. The algorithm parameters for all the algorithms compared are described in Table 4.1. These algorithm parameters are predefined in the original algorithmic parameters to define the exploration and exploitation in a convergence of the algorithm and improve the precision of the solution.

Table 4.1: The parameter setting of the algorithms.

DE	Parameters	Mutation Factor (F)		Crossover rate (CR)	
	Value	0.4 - 0.8		0.5	
PSO	Parameters	w	C_1, C_2	V_{Max}	
	Value	0.3 - 0.95	2, 2	4	
SCA	Parameters	r_2	r_3	r_4	a
	Value	$2\pi \cdot rand$	$2\pi \cdot rand$	$rand$	2
MFO	Parameters	t	a	b	
	Value	[-1, 1]	$-1 - t\frac{-1}{T}$	1	
WOA	Parameters	a	a_2	p	
	Value	$2 - t\frac{2}{T}$	$-1 + t\frac{-1}{T}$	$rand$	
MFO-WOA	Parameters	t	a	b	
	Value	[-1, 1]	$-1 - t\frac{-1}{T}$	1	
	Parameters	a	a_2	p	
	Value	$2 - t\frac{2}{T}$	$-1 + t\frac{-1}{T}$	$rand$	

The feasibility of the new MFO-WOA basic hybrid scheme is evaluated with other state-of-the-art algorithms, such as SCA[68], PSO[122], DE[126], MFO[136] and WOA[137]. These algorithms have many parameters to initialize before they run. Table 4.1 displays the initial values of parameters required for all algorithms. Typically, the optimal control parameters depend on conditions, and they are unknown without prior knowledge.

The convergence speed for benchmark functions (F1-F23) is tested using proposed hybrid MFO-WOA algorithm in addition to compare with SCA[68], PSO[122], MFO[136], WOA[137] and DE[126] algorithms, as shown in Figure 4.2, respectively. Throughout the iteration process, the downward trend shows the potential of the hybrid WOA-mGWO algorithm to achieve improved convergence to the optimal global solution. The same is compared to other algorithms like SCA[68], PSO[122], MFO[136], WOA[137] and DE[126] for 23 benchmark functions.



(a) F1

(b) F2

(c) F3

(d) F4

(e) F5

(f) F6

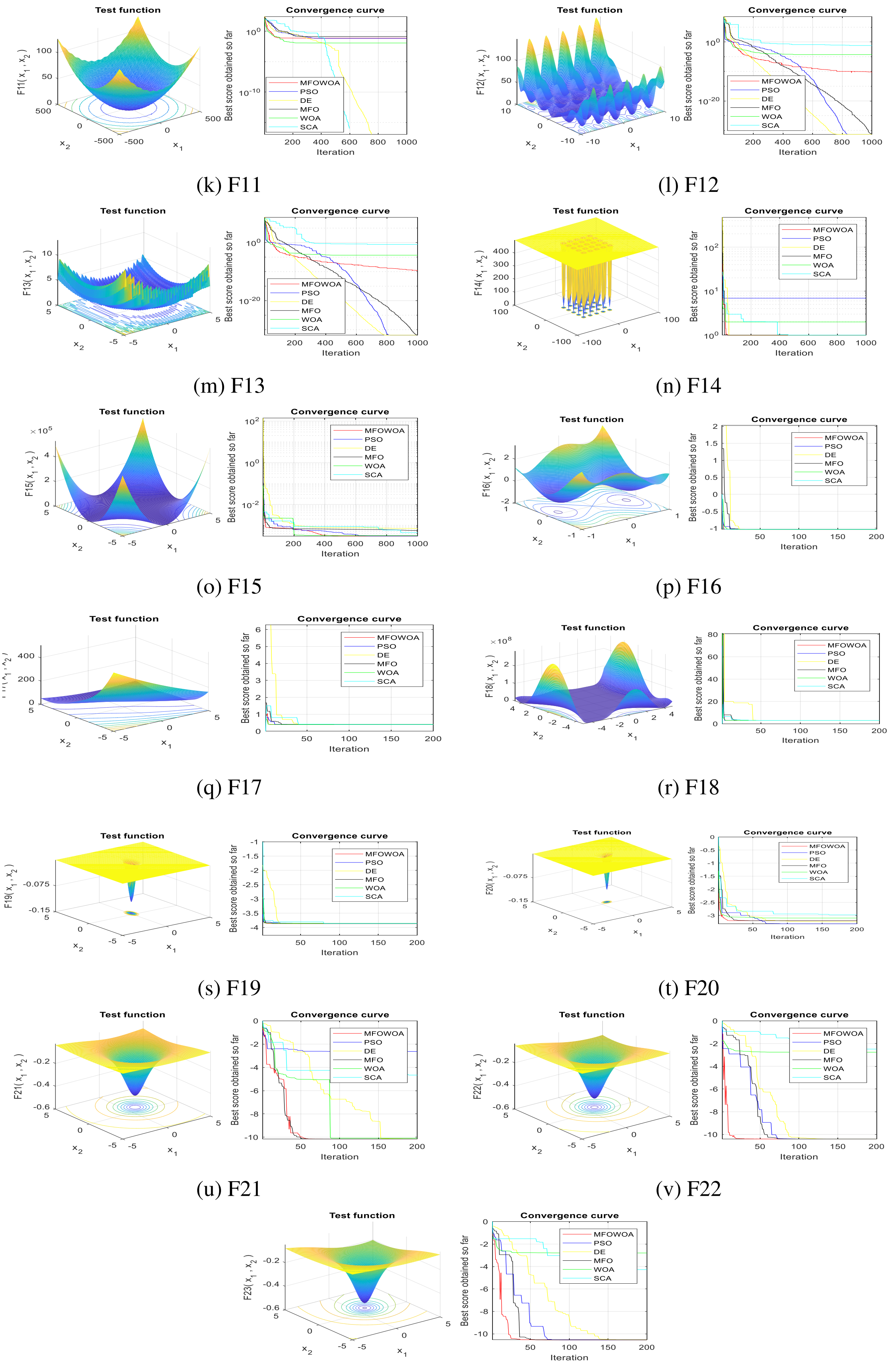
(g) F7

(h) F8

(i) F9

(j) F10

Figure 4.2: Continued...



(k) F11

(l) F12

(m) F13

(n) F14

(o) F15

(p) F16

(q) F17

(r) F18

(s) F19

(t) F20

(u) F21

(v) F22

(w) F23

Figure 4.2: Convergence comparisons of proposed algorithms and others.

4.3.3 Statistical analysis

The statistical test of the hybrid MFO-WOA algorithm is correlated with other algorithms based on 23 traditional benchmark functions [134] are given in Table 4.2, to evaluate the effectiveness of the introduced algorithm. The simulation tests evaluated of the benchmark mentioned above functions by 30 individual runs employing introduced hybrid MFO-WOA algorithm are related with additional algorithms such as SCA[68], PSO[122], MFO[136], WOA[137] and DE[126]. In Table 4.2 & Table 4.3, Mean (M), Best (B), Worst (W), and Standard Deviation (SD) are listed for 23 benchmark functions obtained after the final iteration for the population size of $NP = 50$ & $NP = 100$ over 30 independent runs, respectively. The validation of the proposed hybrid MFO-WOA algorithm for the benchmark functions reveals that the outperforms 19 of 23 standard benchmark functions.

Table 4.2: Statistical results for $F1 - F23$ functions over 30 independent runs ($NP = 50$)

Fun		MFO-WOA	MFO	WOA	SCA	PSO	DE
F1	Best	2.94E-90	1.22E-34	1.86E-91	2.69E-36	1.26E-53	6.11E-43
	Average	1.14E-85	1.47E-31	1.53E-78	4.76E-29	2.45E-48	6.70E-42
	SD	3.58E-85	2.32E-31	0	1.51E-28	7.63E-48	9.39E-42
	Med	2.87E-88	3.01E-32	1.16E-84	6.23E-33	4.30E-51	1.96E-42
	Worst	1.13E-84	7.34E-31	1.48E-77	4.76E-28	2.42E-47	3.08E-41
F2	Best	2.05E-52	3.71E-20	1.74E-21	6.71E-25	2.25E-29	5.46E-25
	Average	6.54E-51	6.80E-19	2.02E-82	7.02E-21	9.49E-27	1.44E-24
	SD	1.20E-50	1.23E-18	6.30E-92	2.04E-20	2.66E-26	8.94E-25
	Med	2.74E-51	3.10E-19	1.77E-95	6.93E-23	2.45E-28	1.08E-24
	Worst	4.00E-50	4.14E-18	2.00E-81	6.50E-20	8.50E-26	2.97E-24
F3	Best	7.24E-25	2.18E-13	4.55E-08	5.49E-21	3.13E-17	6.45E-02
	Average	1.78E-20	1.71E-09	5.21689	4.51E-11	3.77E-15	1.80E-01
	SD	3.38E-20	4.97E-09	12.15415	1.31E-10	7.21E-15	1.84E-01
	Med	1.54E-21	2.23E-11	0.706362	1.49E-14	3.75E-16	1.06E-01
	Worst	9.97E-20	1.58E-08	39.20765	4.16E-10	2.30E-14	6.75E-01
F4	Best	4.06E-23	3.84E-10	9.76E-14	1.26E-12	9.03E-14	7.43E-08
	Average	7.40E-20	0.009285	0.041382	2.31E-09	4.45E-13	1.45E-07

	SD	2.20E-19	0.029361	0.066278	4.96E-09	6.18E-13	4.14E-08
	Med	2.28E-21	2.09E-08	0.001792	5.11E-11	1.53E-13	1.40E-07
	Worst	6.99E-19	0.092848	0.156086	1.47E-08	1.88E-12	2.03E-07
F5	Best	0.535995	0.19356	5.200004	6.162045	0.13815	0.17133
	Average	1.128302	63.05099	5.598858	6.822644	3.104964	2.52303
	SD	1.014791	173.1727	0.337705	0.37006	1.338365	2.116125
	Med	0.856349	4.961773	5.493336	6.794048	3.728973	2.171838
	Worst	3.975478	555.1401	6.088107	7.319884	4.306164	6.129426
F6	Best	4.03E-11	0	3.54E-06	0.107641	0	0
	Average	1.80E-10	7.40E-32	9.50E-06	0.261724	0	0.00E+00
	SD	1.37E-10	9.28E-32	7.70E-06	0.099956	0	0.00E+00
	Med	1.38E-10	3.24E-32	6.41E-06	0.265173	0	0.00E+00
	Worst	4.78E-10	2.62E-31	2.76E-05	0.399885	0	0.00E+00
F7	Best	0.000783	0.000467	0.000129	0.000126	0.000384	4.95E-04
	Average	0.002332	0.00274	0.000785	0.000893	0.001709	0.002189
	SD	0.001049	0.002374	0.00136	0.000869	0.000821	0.000912
	Med	0.002357	0.001971	0.000317	0.000615	0.001694	0.002333
	Worst	0.003953	0.008028	0.004582	0.00306	0.002761	0.003362
F8	Best	-4189.83	-4071.39	-4189.8	-2387.24	-3005.43	-4189.83
	Average	-3395.69	-3407.22	-3867	-2210.55	-2452.48	-4177.99
	SD	357.8439	356.5105	593.989	104.4756	320.111	37.45349
	Med	-3242.32	-3476.16	-4130.06	-2205.06	-2393.37	-4189.83
	Worst	-3005.45	-2760.98	-2569.91	-2036.39	-1998.17	-4071.39
F9	Best	0	6.964708	0	0	0.994959	0
	Average	0	18.01581	0	0	2.885381	0.00E+00
	SD	0	9.39909	0	0	1.280198	0.00E+00
	Med	0	14.42688	0	0	2.984877	0
	Worst	0	33.89951	0	0	4.974795	0.00E+00
F10	Best	8.88E-16	4.44E-15	8.88E-16	8.88E-16	4.44E-15	4.44E-15
	Average	3.02E-15	4.44E-15	4.44E-15	4.80E-15	4.44E-15	4.44E-15
	SD	1.83E-15	0	2.37E-15	3.11E-15	0	0.00E+00
	Med	4.44E-15	4.44E-15	4.44E-15	4.44E-15	4.44E-15	4.44E-15
	Worst	4.44E-15	4.44E-15	7.99E-15	1.15E-14	4.44E-15	4.44E-15

F11	Best	0	0.014772	0	0	0.039319	0
	Average	0.062979	0.156263	0.019722	0.011984	0.11343	0
	SD	0.055109	0.11707	0.034714	0.028414	0.048353	0
	Med	0.059056	0.144008	0	0	0.102138	0.00E+00
	Worst	0.16231	0.457417	0.09271	0.087518	0.19184	0
F12	Best	1.52E-11	4.71E-32	1.79E-06	0.028889	4.71E-32	4.71E-32
	Average	4.37E-11	0.124403	1.78E-05	0.055445	4.71E-32	4.71E-32
	SD	2.85E-11	0.160603	1.43E-05	0.01565	1.15E-47	1.15E-47
	Med	4.02E-11	7.83E-32	1.40E-05	0.059949	4.71E-32	4.71E-32
	Worst	1.05E-10	0.311007	4.71E-05	0.077508	4.71E-32	4.71E-32
F13	Best	8.75E-12	1.35E-32	2.55E-05	0.144472	1.35E-32	1.35E-32
	Average	1.40E-10	0.001099	0.001174	0.261648	1.35E-32	1.35E-32
	SD	1.27E-10	0.003475	0.003461	0.066753	2.88E-48	2.88E-48
	Med	9.56E-11	1.72E-32	7.96E-05	0.275608	1.35E-32	1.35E-32
	Worst	3.74E-10	0.010987	0.011024	0.350477	1.35E-32	1.35E-32
F14	Best	0.998004	0.998004	0.998004	0.998004	0.998004	0.998004
	Average	0.998004	1.493439	2.951039	1.395012	2.479232	0.998004
	SD	1.96E-16	1.070019	4.11736	0.836472	2.139526	0
	Med	0.998004	0.998004	0.998004	0.998042	1.992031	0.998004
	Worst	0.998004	3.96825	10.76318	2.982105	6.903336	0.998004
F15	Best	0.000307	0.000698	0.000312	0.000328	0.000308	0.000403
	Average	0.000307	0.001347	0.000677	0.000849	0.000685	0.000652
	SD	1.64E-09	0.001488	0.000392	0.000421	0.000304	1.11E-04
	Med	0.000307	0.000783	0.000554	0.000729	0.000725	0.000699
	Worst	0.000307	0.005515	0.001489	0.001372	0.001068	0.000753
F16	Best	-1.03163	-1.03163	-1.03163	-1.03162	-1.03163	-1.03163
	Average	-1.03163	-1.03163	-1.03163	-1.03161	-1.03163	-1.03163
	SD	2.87E-16	0	1.19E-11	1.27E-05	0	0.00E+00
	Med	-1.03163	-1.03163	-1.03163	-1.03162	-1.03163	-1.03163
	Worst	-1.03163	-1.03163	-1.03163	-1.03158	-1.03163	-1.03163
F17	Best	0.397887	0.397887	0.397887	0.3979	0.397887	0.397887
	Average	0.397887	0.397887	0.397887	0.398151	0.397887	0.397887
	SD	5.39E-14	0	7.05E-08	0.000231	0	0

	Med	0.397887	0.397887	0.397887	0.39811	0.397887	0.397887
	Worst	0.397887	0.397887	0.397888	0.398536	0.397887	0.397887
F18	Best	3	3	3	3.000003	3	3
	Average	3	3	3.000001	3.000013	3	3
	SD	1.07E-14	9.71E-16	1.51E-06	1.45E-05	3.31E-16	4.19E-16
	Med	3	3	3	3.000008	3	3
	Worst	3	3	3.000005	3.00005	3	3
F19	Best	-3.86278	-3.86278	-3.86278	-3.86277	-3.86278	-3.86278
	Average	-3.86278	-3.86278	-3.86243	-3.85641	-3.86278	-3.86278
	SD	6.15E-14	9.36E-16	0.000539	0.003304	9.36E-16	9.36E-16
	Med	-3.86278	-3.86278	-3.86271	-3.85489	-3.86278	-3.86278
	Worst	-3.86278	-3.86278	-3.86123	-3.85248	-3.86278	-3.86278
F20	Best	-3.322	-3.322	-3.32198	-3.1725	-3.322	-3.322
	Average	-3.25066	-3.23806	-3.21374	-2.70975	-3.28633	-3.322
	SD	0.061396	0.057935	0.079892	0.623354	0.057431	5.34E-16
	Med	-3.2031	-3.2031	-3.19825	-3.01018	-3.322	-3.322
	Worst	-3.2031	-3.19957	-3.11136	-1.45523	-3.2031	-3.322
F21	Best	-10.1532	-10.1532	-10.1532	-6.68568	-10.1532	-10.1532
	Average	-8.6238	-7.61787	-9.64326	-4.34527	-7.88064	-10.1532
	SD	2.462569	2.672523	1.612085	1.536392	3.018276	9.33E-11
	Med	-10.1532	-7.62699	-10.1531	-4.8065	-10.1532	-10.1532
	Worst	-5.0552	-5.0552	-5.05518	-0.88163	-2.63047	-10.1532
F22	Best	-10.4029	-10.4029	-10.4029	-5.73628	-10.4029	-10.4029
	Average	-9.33989	-9.344	-9.87095	-3.63368	-9.344	-10.4029
	SD	2.241114	2.23246	1.680674	1.983269	2.23246	1.60E-14
	Med	-10.4029	-10.4029	-10.4024	-4.77576	-10.4029	-10.4029
	Worst	-5.08767	-5.08767	-5.08767	-0.90729	-5.08767	-10.4029
F23	Best	-10.5364	-10.5364	-10.5363	-5.04159	-10.5364	-10.5364
	Average	-8.37324	-6.97982	-8.01145	-4.56556	-10.0003	-10.5364
	SD	2.792643	3.821151	3.323159	0.534556	1.695222	1.87E-15
	Med	-10.5364	-7.85603	-10.5357	-4.82897	-10.5364	-10.5364
	Worst	-5.12848	-2.42734	-2.80663	-3.50791	-5.17565	-10.5364

Table 4.3: Statistical results for $F1 - F23$ functions over 30 independent runs ($NP = 100$)

Fun		MFOWOA	MFO	WOA	SCA	PSO	DE
F1	Best	1.28E-99	3.15E-35	1.28E-98	4.79E-43	2.62E-61	5.90E-43
	Average	3.33E-96	1.03E-32	3.26E-93	5.57E-36	2.73E-58	4.12E-42
	SD	8.89E-93	1.61E-32	0	1.76E-35	7.31E-58	3.25E-42
	Med	2.64E-94	2.40E-33	9.25E-89	7.62E-40	8.13E-60	3.46E-42
	Worst	2.85E-99	4.77E-32	2.71E-99	5.56E-35	2.33E-57	9.37E-42
F2	Best	2.18E-99	7.44E-21	2.85E-96	5.17E-28	2.66E-33	7.80E-25
	Average	4.56E-59	3.84E-19	8.31E-57	1.28E-24	4.04E-32	1.43E-24
	SD	7.49E-89	4.64E-19	2.60E-96	2.75E-24	4.01E-32	6.08E-25
	Med	1.00E-59	2.35E-19	1.08E-90	1.17E-25	1.86E-32	1.32E-24
	Worst	2.37E-88	1.45E-18	8.22E-96	8.88E-24	1.04E-31	2.89E-24
F3	Best	5.28E-31	1.37E-14	1.37E-07	2.86E-25	9.39E-22	0.08166
	Average	3.75E-24	2.49E-11	0.023035	9.17E-18	1.80E-19	0.21658
	SD	1.19E-23	6.30E-11	0.05397	1.93E-17	4.70E-19	0.14126
	Med	8.82E-29	9.03E-13	0.00214	3.67E-19	1.73E-20	0.16627
	Worst	3.75E-23	2.02E-10	0.17163	6.11E-17	1.51E-18	0.48825
F4	Best	8.18E-29	2.47E-11	3.25E-13	1.84E-14	9.44E-19	1.07E-07
	Average	4.86E-27	1.50E-10	4.23E-06	8.79E-13	1.07E-16	1.77E-07
	SD	8.17E-27	1.22E-10	9.15E-06	1.92E-12	1.16E-16	6.02E-08
	Med	8.23E-28	1.20E-10	1.45E-08	1.49E-13	8.19E-17	1.61E-07
	Worst	2.13E-26	3.79E-10	2.56E-05	6.25E-12	3.55E-16	3.19E-07
F5	Best	0.08427	0.28553	4.15771	6.45554	0.00031	0.40297
	Average	0.18934	21.3708	4.79959	6.88535	3.07874	1.22918
	SD	0.08386	46.9083	0.28895	0.32674	1.75284	1.03471
	Med	0.1663	4.36164	4.88663	6.95772	3.51002	0.80308
	Worst	0.3671	153.166	5.10277	7.27263	5.63625	3.84601
F6	Best	4.94E-12	0	2.12E-07	0.12431	0	0
	Average	9.65E-12	8.63E-33	7.33E-07	0.24028	0	0
	SD	3.50E-12	1.80E-32	6.30E-07	0.14226	0	0
	Med	9.10E-12	1.54E-33	4.59E-07	0.16471	0	0
	Worst	1.48E-11	5.85E-32	2.19E-06	0.56784	0	0

F7	Best	0.00025	0.00074	6.00E-06	3.08E-05	0.00067	0.00086
	Average	0.00143	0.00185	0.00056	0.00042	0.00127	0.00204
	SD	0.00116	0.00067	0.00056	0.00036	0.00065	0.00069
	Med	0.00116	0.00185	0.00039	0.00035	0.0012	0.00209
	Worst	0.00323	0.00292	0.0017	0.00135	0.00288	0.00319
F8	Best	-4071.39	-3951.43	-4189.82	-2783.41	-3143.5	-4189.82
	Average	-3633.06	-3408.42	-3835.89	-2484.03	-2671.61	-4189.82
	SD	321.05	371.86	591.72	175.02	211.75	9.59E-13
	Med	-3597.63	-3486.79	-4189.81	-2447.06	-2600.41	-4189.82
	Worst	-3123.88	-2759.46	-2549.04	-2261.86	-2452.61	-4189.82
F9	Best	0	6.9647134	0	0	0	0
	Average	0	13.2329	0	0	1.89042	0
	SD	0	6.06206	0	0	1.72012	0
	Med	0	11.9394	0	0	0.99495	0
	Worst	0	24.8739	0	0	5.96974	0
F10	Best	8.88E-16	4.44E-15	8.88E-16	8.88E-16	4.44E-15	4.44E-15
	Average	3.02E-15	4.44E-15	3.38E-15	4.09E-15	4.44E-15	4.44E-15
	SD	1.83E-15	0	2.40E-15	1.12E-15	0	0
	Med	4.44E-15	4.44E-15	4.44E-15	4.44E-15	4.44E-15	4.44E-15
	Worst	4.44E-15	4.44E-15	7.99E-15	4.44E-15	4.44E-15	4.44E-15
F11	Best	0	0.07386	0	0	0.04183	0
	Average	0.03222	0.15426	0.020568	0.0776	0.1431	0
	SD	0.04229	0.08881	0.06504	0.16383	0.08209	0
	Med	0	0.14159	0	0	0.10948	0
	Worst	0.09591	0.36404	0.20568	0.39838	0.29028	0
F12	Best	2.84E-13	4.71E-32	5.05E-07	0.02377	4.71E-32	4.71E-32
	Average	3.57E-12	0.03116	1.56E-06	0.0474	4.71E-32	4.71E-32
	SD	2.88E-12	0.09834	1.30E-06	0.01923	1.15E-47	1.15E-47
	Med	2.65E-12	4.81E-32	1.28E-06	0.04487	4.71E-32	4.71E-32
	Worst	8.52E-12	0.31101	4.85E-06	0.08402	4.71E-32	4.71E-32
F13	Best	2.43E-12	1.35E-32	1.08E-06	0.08472138	1.35E-32	1.35E-32
	Average	1.29E-11	0.00329	5.76E-06	0.20034	1.35E-32	1.35E-32
	SD	5.85E-12	0.00532	6.23E-06	0.07881	2.88E-48	2.88E-48

	Med	1.33E-11	1.47E-32	3.02E-06	0.166608357	1.35E-32	1.35E-32
	Worst	2.07E-11	0.01098	2.11E-05	0.31359	1.35E-32	1.35E-32
F14	Best	0.998003	0.998003	0.998003	0.998003	0.998003	0.998003
	Average	0.998003	0.998003	0.998003	0.998013	1.19685	0.998003
	SD	1.66E-16	0	3.00E-12	2.66E-05	0.41911861	0
	Med	0.998003	0.998003	0.998003	0.998004	0.998003	0.998003
	Worst	0.998003	0.998003	0.998003	0.998088	1.9929	0.998003
F15	Best	0.000307	0.000459	0.000309	0.000328	0.000307	0.000465
	Average	0.000399	0.000833	0.000722	0.000782	0.000529	0.000575
	SD	0.000289	0.00034	0.000444	0.00043	0.000277	9.08E-05
	Med	0.000307	0.000732	0.000498	0.000603	0.000351	0.000566
	Worst	0.001223	0.001655	0.00123	0.001313	0.000986	0.000732
F16	Best	-1.03162	-1.03162	-1.03162	-1.03162	-1.03162	-1.03162
	Average	-1.03162	-1.03162	-1.03162	-1.03162	-1.03162	-1.03162
	SD	7.40E-17	0	5.23E-14	5.83E-06	0	0
	Med	-1.03162	-1.03162	-1.03162	-1.03162	-1.03162	-1.03162
	Worst	-1.03162	-1.03162	-1.03162	-1.03167	-1.03162	-1.03162
F17	Best	0.39788	0.39788	0.39788	0.39792	0.39788	0.39788
	Average	0.39788	0.39788	0.39788	0.39823	0.39788	0.39788
	SD	2.80E-15	0	4.72E-09	0.00036	0	0
	Med	0.39788	0.39788	0.39786	0.3980	0.39788	0.39788
	Worst	0.39788	0.39788	0.3978	0.39885	0.39788	0.39788
F18	Best	3	3	3	3.000064	3	3
	Average	3	3	3.00005	3.00009	3	3
	SD	9.18E-15	8.63E-16	4.46E-07	9.55E-07	6.28E-16	5.92E-16
	Med	3	3	3.00003	3.00077	3	3
	Worst	3	3	3.00051	3.00003	3	3
F19	Best	-3.8627	-3.8627	-3.8627	-3.8623	-3.8627	-3.8627
	Average	-3.8627	-3.8627	-3.862	-3.8577	-3.8627	-3.8627
	SD	3.21E-15	9.36E-16	0.00011	0.00371	9.36E-16	9.36E-16
	Med	-3.8627	-3.8627	-3.86261	-3.8548	-3.8627	-3.8627
	Worst	-3.8627	-3.8627	-3.8624	-3.8547	-3.8627	-3.8627
	Best	-3.3219	-3.3219	-3.3219	-3.18817	-3.3219	-3.3219

F20

	Average	-3.23876	-3.2268	-3.2331	-3.0698	-3.2863	-3.3219
	SD	0.05743	0.05012	0.0844	0.0668	0.05743	4.68E-16
	Med	-3.2031	-3.2031	-3.20232	-3.04376	-3.32192	-3.3219
	Worst	-3.2031	-3.2031	-3.08165	-3.0046	-3.2031	-3.32199
F21	Best	-10.1531	-10.1531	-10.1531	-5.22711	-10.1531	-10.1531
	Average	-8.62379	-7.63953	-10.1531	-3.70951	-6.09758	-10.1531
	SD	2.46256	3.34057	1.97E-05	2.04567	2.13759	0
	Med	-10.1531	-10.1531	-10.153	-4.93351	-5.1007	-10.1531
	Worst	-5.05519	-2.68286	-10.1531	-0.49652	-5.05519	-10.1531
F22	Best	-10.4029	-10.4029	-10.4029	-7.63639	-10.4029	-10.4029
	Average	-8.80835	-9.20766	-10.4027	-5.56358	-8.82076	-10.4029
	SD	2.56751	2.54151	0.00048	1.05902	2.5476	1.57E-15
	Med	-10.4029	-10.4029	-10.402	-5.15943	-10.4029	-10.4029
	Worst	-5.08767	-3.72437	-10.4016	-4.23197	-5.12887	-10.4029
F23	Best	-10.5364	-10.5364	-10.5364	-8.03951	-10.5364	-10.5364
	Average	-9.99561	-10.5364	-9.99325	-6.2502	-10.5364	-10.5364
	SD	1.71013	1.87E-15	1.7175	1.34447	1.87E-15	1.87E-15
	Med	-10.5364	-10.5364	-10.5363	-5.90402	-10.5364	-10.5364
	Worst	-5.12848	-10.5364	-5.1065	-4.98221	-10.5364	-10.5364

4.3.4 Non-Parametric test analysis

In each of the individual runs, a comparison of algorithms based on the best, mean, worst, and standard deviation values for 30 separate runs. Therefore, there is always the possibility that the predominance may have happened accidentally in 30 runs, even with low probability. Further analysis is then carried out using the Wilcoxon rank-sum statistical test [142] at a level of 5 percent. The advantage is stated by analyzing the p-values of each run, as listed in Table 4.4. Here, the same algorithm can not be compared with itself, and it has resulted in 1 for each function. However, most of the benchmark functions showing the algorithm's statistical importance, the hybrid MFO-WOA algorithm results in p-values much less than 0.05, except for function $F21$. Thus, the MFO-WOA hybrid algorithm's overall success shows its predominance over other research competition algorithms.

4.3.5 Discussions

The hybrid MFO-WOA and other algorithms like SCA, PSO, DE, MFO, and WOA are considered with the parameters of initial population size $NP = 50$ and 100 on each twenty-three

Table 4.4: P-Values of Wilcoxon rank-sum test results

		P-Values				
Functions	MFO-WOA	MFO	WOA	SCA	PSO	DE
F1	1	6.8E-08	6.8E-08	6.8E-08	6.8E-08	6.8E-08
F2	1	6.8E-08	6.8E-08	6.8E-08	6.8E-08	6.8E-08
F3	1	6.8E-08	6.8E-08	7.9E-08	6.8E-08	6.8E-08
F4	1	6.8E-08	6.8E-08	6.8E-08	6.8E-08	6.8E-08
F5	1	1.2E-06	1.2E-06	6.8E-08	3.9E-07	1.6E-06
F6	1	4.9E-08	6.8E-08	6.8E-08	8.0E-09	8.0E-09
F7	1	1.7E-03	4.7E-03	1.8E-03	3.9E-01	2.3E-03
F8	1	6.9E-03	1.5E-01	6.8E-08	9.2E-08	8.0E-09
F9	1	7.9E-09	NaN	3.4E-01	2.2E-08	NaN
F10	1	1.0E+00	7.9E-01	5.8E-01	1.0E+00	1.0E+00
F11	1	1.4E-07	8.3E-01	1.3E-01	4.6E-06	4.5E-03
F12	1	9.8E-07	6.8E-08	6.8E-08	8.0E-09	8.0E-09
F13	1	1.4E-05	6.8E-08	6.8E-08	8.0E-09	8.0E-09
F14	1	8.6E-08	1.1E-07	5.2E-07	2.8E-07	8.6E-08
F15	1	7.8E-04	1.6E-04	1.6E-05	2.3E-03	1.2E-03
F16	1	3.2E-09	3.2E-09	3.4E-08	3.2E-09	3.2E-09
F17	1	7.4E-10	9.7E-09	1.1E-08	7.4E-10	7.4E-10
F18	1	7.9E-09	6.7E-08	6.7E-08	7.9E-09	7.9E-09
F19	1	6.6E-09	5.9E-08	5.9E-08	6.6E-09	6.6E-09
F20	1	6.6E-04	2.0E-01	2.6E-07	4.3E-01	8.0E-09
F21	1	7.7E-02	7.1E-03	3.0E-07	9.3E-03	8.0E-09
F22	1	8.6E-05	1.6E-04	1.4E-06	4.0E-06	8.0E-09
F23	1	6.0E-05	8.3E-05	1.4E-06	2.7E-06	8.0E-09

benchmark functions for comparison of 30 independent runs. The empirical results contain several statistical test parameters, like standard deviation, median, average, and worst. The best-so-far solution in the last generation is reported, and the statistical test results are tabulated in Table 4.2, Table 4.3. To verify the algorithm efficiency, the previously mentioned algorithm parameters are fixed for all algorithms.

The statistical analysis is essential to verify that a proposed novel algorithm offers a significant advancement over other existing algorithms overall benchmark problems. The best solution in each generation is saved and drawn as the convergence curves in Figure 4.2. The descending trend is evident in the convergence curves of hybrid MFO-WOA on many of the test functions investigated. This strongly proves the hybrid MFO-WOA algorithm's ability to reach the global optimum throughout generations better.

If an algorithm improves its candidate solutions, the average fitness should be enhanced throughout generations. As the average fitness curves in Figure 4.2 suggest, the hybrid MFO-

WOA algorithm shows degrading fitness on all test functions. Another fact worth mentioning here is the accelerated decrease in the average fitness curves, which leads to improved candidate solutions' improvement over generations.

4.4 Case studies

Numerous cases of simulation of the multiple systems and additional conventional control systems are described in this section. The closed-loop responses to reference setpoints tracking and outside disturbances are assessed and compared with different past studies in state-of-art.

4.4.1 Proposed FOPID/PID Controller Design

The typical structure of PID controller [9], [26] is as stated in Eq.(4.25)

$$C(s) = K_p + \frac{K_i}{s} + K_d s \tag{4.25}$$

where K_p is the proportional gain, K_i is the integral gain and K_d is the derivative gain.

The FOPID ($PI^{\lambda}D^{\mu}$) controller [31] is generalization from the standard PID controller [143] as in Eq.(4.26) with a fractional order integral and derivatives.

$$C(s) = K_p + \frac{K_i}{s^{\lambda}} + K_d s^{\mu} \tag{4.26}$$

where λ and μ are the non-integer orders of the integral and derivative terms respectively. This can be quickly noted that, by choosing λ=μ=1, a standard PID controller is achieved.

In the following controller design, the objective of a PID/FOPID controller is given in Eq.(4.27)

$$\begin{aligned} J_{min}(K) &= e^{-\beta}(M_P + E_{ss}) + e^{(1-\beta)}(T_s - T_r) \\ &+ (1 - e^{-\beta})(\frac{1}{GM} + \frac{1}{PM}) \end{aligned} \tag{4.27}$$

In this study, a Hybrid MFO-WOA algorithm for tuning the PID/FOPID controller is as shown in Figure 4.3 to get optimal or near to optimal controller parameters are Kp, Ki, Kd,

λ and μ [144] using the objective of Eq.(4.27), the system initially contains an arbitrarily limited population of solutions. Each potential solution assigned as solution for the candidate. Therefore, the steps of the design procedure for the controller are given as follows.

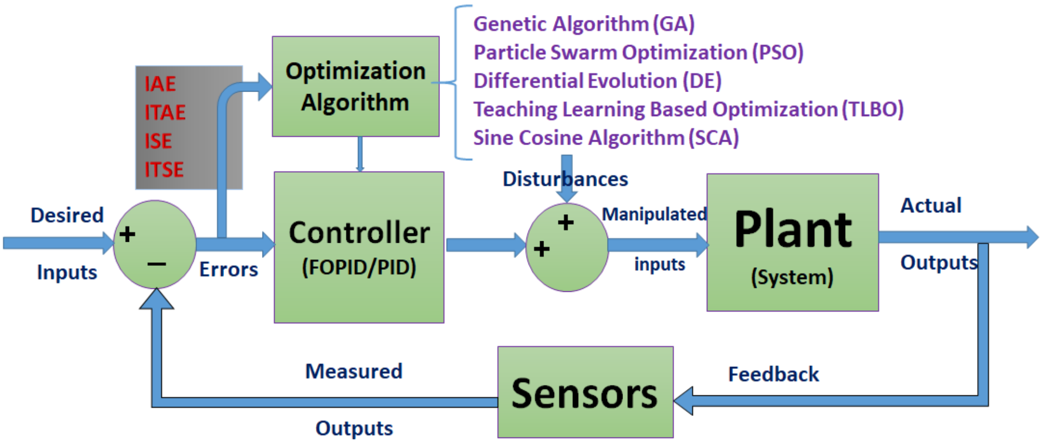


Figure 4.3: Block diagram of optimization control problem formulation.

Steps for optimization based optimal control system design:

1. Initialize the number of controller tuning parameters with minimum & maximum limits, the number of solutions (50), and stopping criterion (500).
2. Obtain the fitness evaluation of each population and controller design performance using Eq.(4.31) in controller design.
3. Update the position based on the *best* solution.
4. Update the algorithmic parameters.
5. Update the position of each population.
6. Record the latest *best* solution positions as the global optimum/best solution of tuning parameters.
7. If the stopping criterion is satisfied, go to step 6.
8. If the stopping criterion is not meet, then go to 2.

The proposed optimization strategy's effectiveness depends on the design metrics in terms of closed-loop performance. This study aims to explain how to successfully designs of

FOPID/PID controller tuning method for situations where the mathematical models are available. Therefore, in numerical instances, a collection of arbitrary but practicable time-domain performance specifications is obtained, rather than an optimum set of performance specifications to achieve excellent closed-loop results.

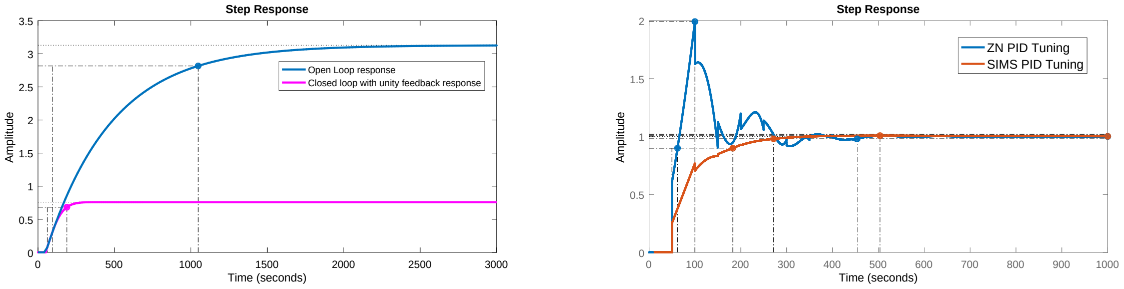
4.4.2 FOPTD system

The case study of the liquid level system model as FOPTD [103] shall be considered for verification of the controller design by the suggested process. The transfer function model for the liquid level system as FOPTD framework is as follows in Eq.(4.28):

$$G_p(s) = \frac{3.13}{433.33s + 1} e^{-50s} \tag{4.28}$$

To verify the advised tuning exercises, the simulation studies on the liquid level system of FOPTD in open-loop and unity feedback closed-loop responses are in the Figure 4.4a. It is clear that from open-loop, the output response has more delay ($50s$) and longer rise time ($902s$) at the undesired response of peak amplitude of 3.3, a steady-state of 3.3 at a settling time of 1695s. In the unity feedback closed-loop, the output response has the same delay time and the rise time is $78.13s$ at the undesired response of peak amplitude of 0.758, a steady-state of 0.758 at a settling time of 202s. Hence, the liquid level system of FOPTD is the need for a controller to give the desired response.

The better closed-loop response can be achieved when the controller is utilized in the loop, with traditional tuning methods [145]. The application PID controller with standard tuning may results in an overshoot in the responses are shown in the Figure 4.4b. These conventional controller design methods lead to the need for proposed design techniques to preferred response and performance matrices.

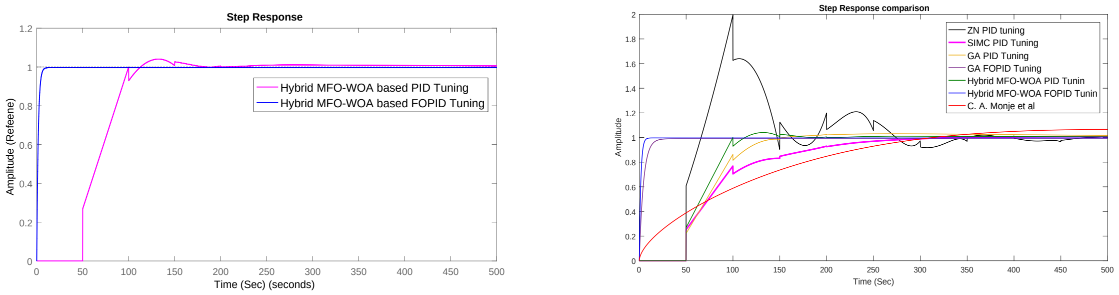


(a) No Controller

(b) PID controller

Figure 4.4: Step change responses of FOPTD system.

The search space constraints will help to solve problems with the hybrid MFO-WOA algorithm to extend for more promising areas. Hence, exploration and optimization capabilities from the hybrid MFO-WOA algorithm expanded throughout the search space, by avoiding local optima, and combining the best approach was found to achieve the global optimum.



(a) System with FOPID/PID controller

(b) Comparisons of FOPID/PID controller

Figure 4.5: Step responses

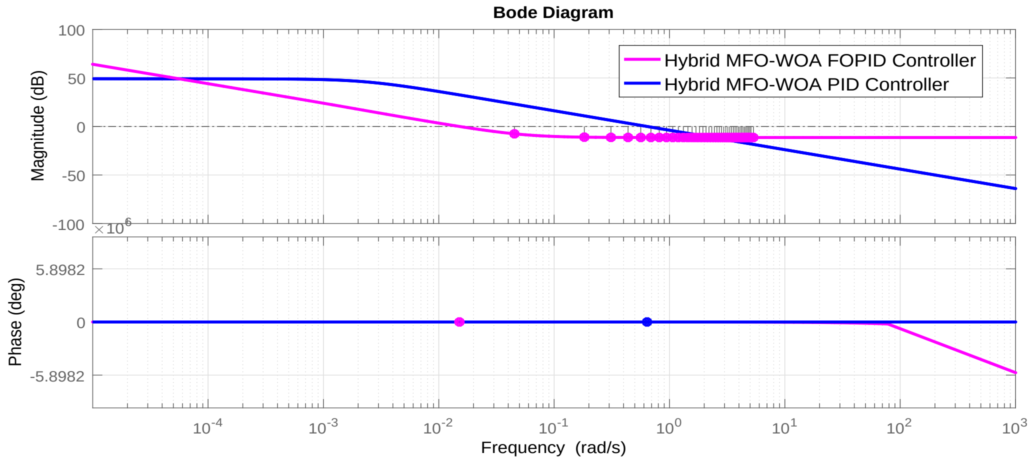


Figure 4.6: Frequency response of a FOPTD system with FOPID/PID.

Figure 4.5 and Figure 4.6 show time responses and frequency response of the proposed FOPID/PID controller designs using a hybrid MFO-WOA algorithm, respectively. Figure 4.5a showed that the current FOPID based algorithm has a better response to the step change configured. It means that the system is more flexible to phase changes, and no over-shoot response during this interval is nearly constant.

In the results, the fractional-order controller's potential in practice compared with other controller designs, regarding performance and robustness aspects. However, the proposed design method involves a hybrid optimization algorithm relating to the model-based controller design specifications. Thus, the purpose now is to simplify the design method so that the controller can be tuned smoothly, with elementary relations among its parameters, and to preserve the robustness characteristics regarding performance. Besides, this new method allowed us to compare with the C. A. Monje [103] FOPID tuning method, Genetic Algorithm (GA) based FOPID controller design and other controller designs are as shown in Figure 4.5b. The FOPID/PID controller design parameters for the liquid level (FOPTD) system and performance for the same controllers are listed in Table 4.5.

Table 4.5: FOPID/PID controller parameters and performance

Methods	Kp	Ki	Kd	λ	μ	Tr	Mp	Ts	Ess
ZN PID Tuning	3.392	0.03416	84.21	-	-	82.026	99.57	403.79	0.138
SIMC PID Tuning	1.502	0.003557	35.09	-	-	72.55	0.74	221.46	0.0916
GA PID Tuning	1.8239	0.005103	30.7007	-	-	63.42	3.05	401.5	0.0653
GA FOPID Tuning	9.9255	10.3503	10.8017	0.01119	0.00691	9.7485	0.02	17.28	0.0614
Hybrid MFO-WOA PID	**2.1028**	**0.0051**	**37.2677**	-	-	**43.52**	**3.43**	**150.74**	**0.0725**
Hybrid MFO-WOA FOPID	**28.24**	**29.45**	**30.74**	**0.0181**	**0.0112**	**3.514**	**0.01**	**6.21**	**0.00359**
C. A. Monje et al	0.6152	0.01	4.3867	0.89	0.4773	266.94	0	384.79	-0.0653

4.4.3 AVR system

The AVR system terminal voltage produced by a generator output at the desired level needs to improve the electrical power quality. Suppose the generator output terminal voltage has voltage fluctuations or a deviation occurs. A sensor identifies the terminal voltage and the actual measured voltage is compared with a reference voltage for the difference. Afterward, the difference is used to power the AVR system terminal voltage. An amplifier magnifies the power signal, and it is used to power the generator with the exciter.

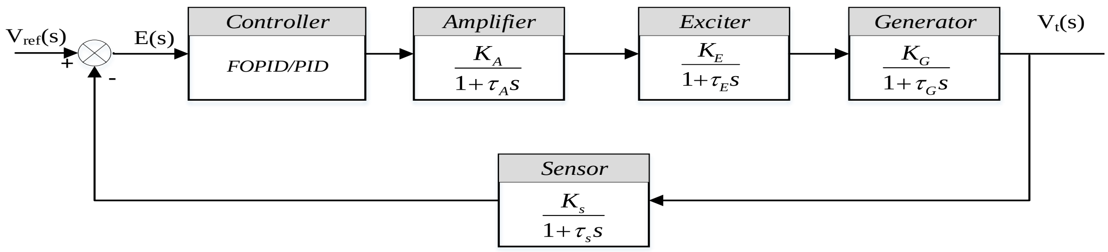


Figure 4.7: Block diagram of AVR system with FOPID/PID controller.

A linear model of the AVR system consists of 4-blocks, the design of a realistic and reliable automatic voltage regulation is of common concern in electrical power systems. It shows each block of the AVR system [32, 108] with controller block diagram in Figure 4.7.

The linear model of each block transfer function is taken from [32],the closed-loop transfer function is as follows:

$$\frac{V_t(s)}{V_{ref}(s)} = \frac{0.1s + 10}{0.0004s^4 + 0.0454s^3 + 0.555s^2 + 1.51s + 11} \tag{4.29}$$

Figure 4.8 presents the output response without any controller for the AVR device. It is evident from the figure that the output response has many oscillations with a peak amplitude of 1.5V (65.4% overshoot), a rise time of 0.261s, at a steady-state of 0.909 and a settling time of 6.970. A PID controller is included to keep the terminal voltage to a fixed value and improve the dynamic response of the AVR system.

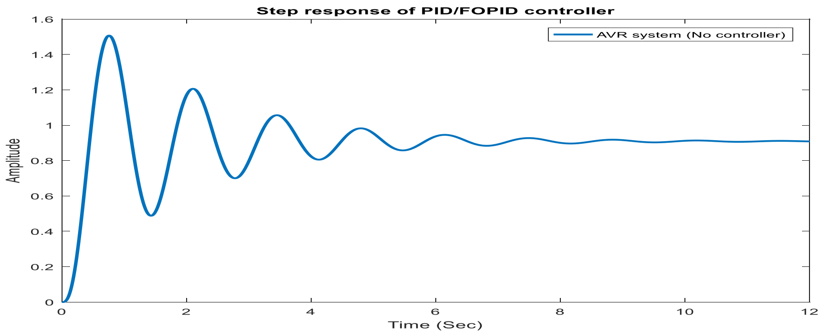
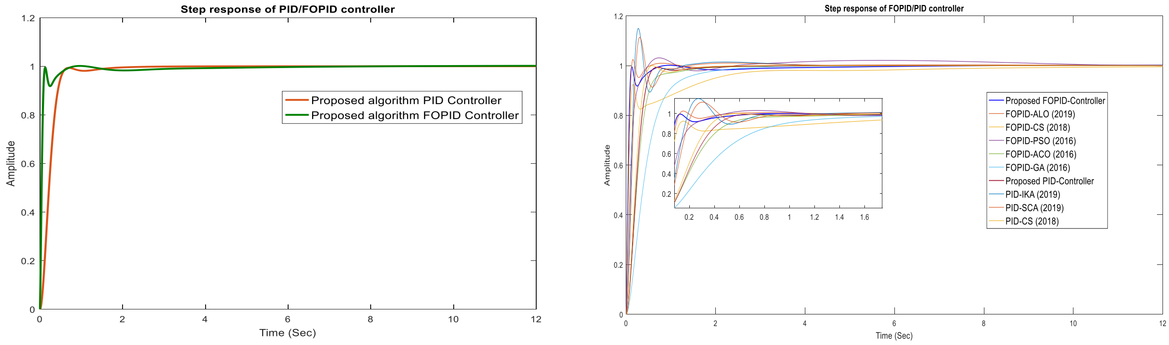


Figure 4.8: AVR system without controller closed-loop step response.

In recent years, PID controllers have been adopted primarily in industries to find solutions with process control applications, as they provide reliable performance, simple design and easy implementation. Several improvements have been made to the PID controller configuration, including a FOPID controller. The addition of a fractional derivative and fractional integration increases the consistency of the system.

The search space range of all gains are in between $[0, 2]$ to improve the optimization process's precision. These constraints will help Hybrid MFO-WOA algorithm explore more promising areas in the search space. Hence, from the Hybrid MFO-WOA's discovery and optimization capabilities, globally exploring the search space, avoiding local optima and instead optimizing the best solutions found to achieve the optimum. However, the hybrid MFO-WOA algorithm is used to evaluate the parameters of the FOPID/PID controller to achieve the optimum response of the AVR system. Figure 4.9a and Figure 4.9b shows time responses of the proposed FOPID/PID controller and comparison with other algorithm-based controllers such as IKA [116], RM-PSO [87], CS [90], ALO [108], PSO [146], ACO [146], GA [146], SCA [147], respectively. Compared to other optimized FOPID/PID controllers, From Figure 4.9b and Figure 4.10, it is found that the current FOPID-based algorithm has a better in time response and frequency response, respectively. Furthermore, the proposed controller parameters for the FOPID/PID controller, along with the performance are listed in Table 4.6.



(a) Proposed FOPID/PID controller. (b) Comparisons of FOPID/PID controller.

Figure 4.9: Step responses of AVR system.

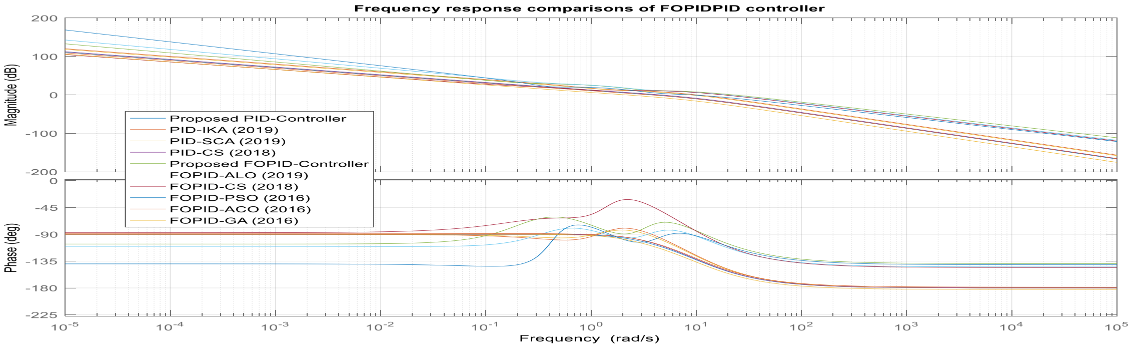


Figure 4.10: Frequency response of AVR system.

Table 4.6: Best parameters and controller performance using various controllers.

Methods	Kp	Ki	La	Kd	Mu	Tr	Ts	Mp	Ess
Proposed WOAMFO-FOPID	**2.1184**	**0.947763**	**0.923733**	**0.576725**	**1.403384**	**0.0518**	**0.792**	**0.12**	**0**
FOPID-CS (2018)	0.8912	0.2485	0.97	0.5105	1.38	0.0986	1.4591	0.1779	0
FOPID-ACO (2016)	0.5	0.33	0.9897	0.1667	1.015	0.377	1.102	5.08	0
FOPID-GA (2016)	0.295	0.2003	0.9989	0.0939	0.9712	0.693	1.08	0.558	0
FOPID-ALO (2019)	3.407	1.0468	1.2244	0.5568	1.3918	0.0626	0.353	10.9	0
FOPID-PSO (2013)	1.2623	0.5526	1.2559	0.2381	1.1832	0.1603	0.2655	0.01	0
FOPID-RM-PSO (2016)	1.9592	0.4919	1.5509	0.2359	1.4325	0.1592	0.2114	0.0108	0
Proposed WOAMFO-PID	**0.59706**	**0.411746**	**1**	**0.2**	**1**	**0.0729**	**0.915**	**0.52**	**0**
PID-CS (2018)	0.6198	0.4165	1	0.2126	1	0.308	0.426	0.0001	0
PID-IKA (2019)	1.0426	1.0093	1	0.599	1	0.128	0.753	15	0
PID-SCA (2019)	0.9826	0.8337	1	0.4982	1	0.148	0.724	1.11	0

The results analysis is tabulated in Table 4.6, it is clear that the proposed hybrid MFO-WOA algorithm-based FOPID/PID controller gives the best dynamic results in terms of rise times (tr), settling times ts and peak time (tp) compared to the meta-heuristic algorithms like IKA [116], RM-PSO [87], CS [90], ALO [108], PSO [146], ACO [146], GA [146], SCA [147] and the third peak overtime performance (Mp).

4.4.4 Wood-Berry system

In industrial production, the Wood-Berry double distillation towel model [60] is a classic multivariable model. Due to the complex characteristics of multi-parameter, time-delay, and strong coupling relationships, the Wood-Berry system control has drawn the attention of academic and industrial researchers [60]. In this paper, some comparative simulations were conducted between the presented FOPID/PID controller based on the Hybrid MFO-WOA and the other methods for estimating the Wood-Berry model's parameters.

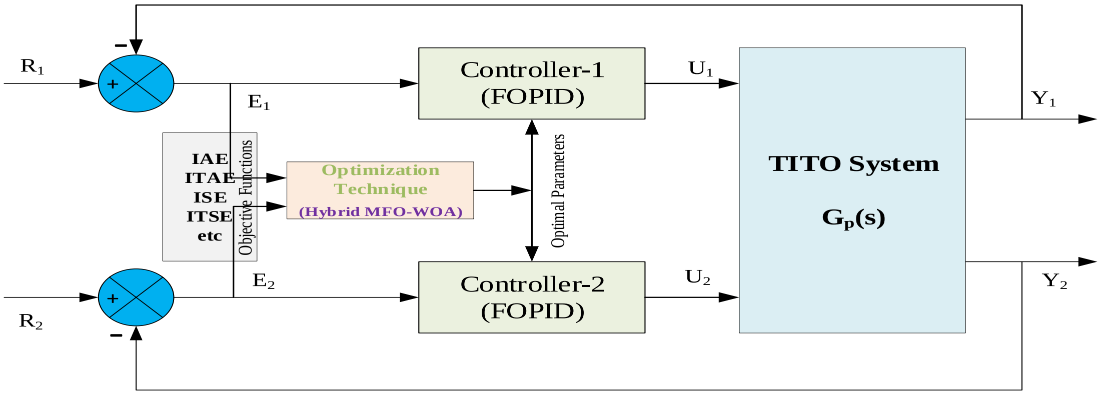


Figure 4.11: FOPID control design for TITO system.

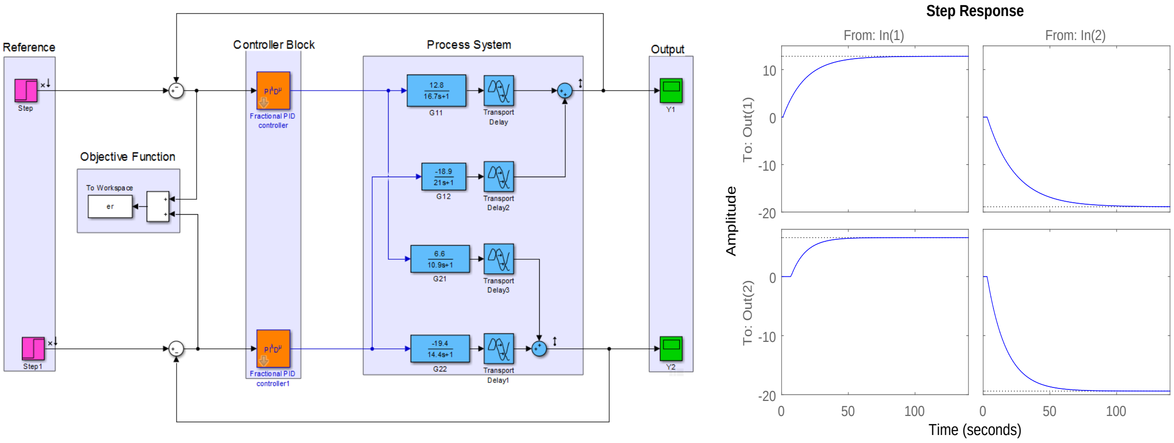
The simulation example is included to estimate the efficiency and effectiveness of the proposed hybrid MFO-WOA algorithm based FOPID controller design to confirm the validity of the proposed FOPID/PID controller in Wood-Berry (WB) binary distillation process system as shown in Figure 4.11. In Eq.(4.30), the Wood-Berry binary distillation process [60, 123] denotes an ideal TITO process including great interaction and significant time delays. The proposed FOPID controller design for WB-TITO system is implemented in the simulation model is as shown in Figure 4.12a. The method keeps the resulting transfer function and the Wood-Berry binary distillation process phase response without decoupling, and no control action response is shown in Figure 4.12b.

$$G_p(s) = \begin{bmatrix} \frac{12.8}{16.7s+1}e^{-s} & \frac{-18.9}{21s+1}e^{-3s} \\ \frac{6.6}{10.9s+1}e^{-7s} & \frac{-19.4}{14.4s+1}e^{-3s} \end{bmatrix} \tag{4.30}$$

The definition or objective of controller design is a fitness function to optimize the multi-loop FOPID controller parameter gains as given in Eq.(4.31)

$$J_{min}(K_p, K_i, \lambda, K_d, \mu) = \int_0^{\infty} (|t \times e_1(t)| + |t \times e_2(t)|)dt \tag{4.31}$$

where '*min*' is minimum of cost function J, and $e_1(t)$ is the first loop error function of ITAE, $e_2(t)$ is the second loop error function of ITAE.



(a) Simulink model of Wood-Berry TITO system.

(b) Step response of WB system.

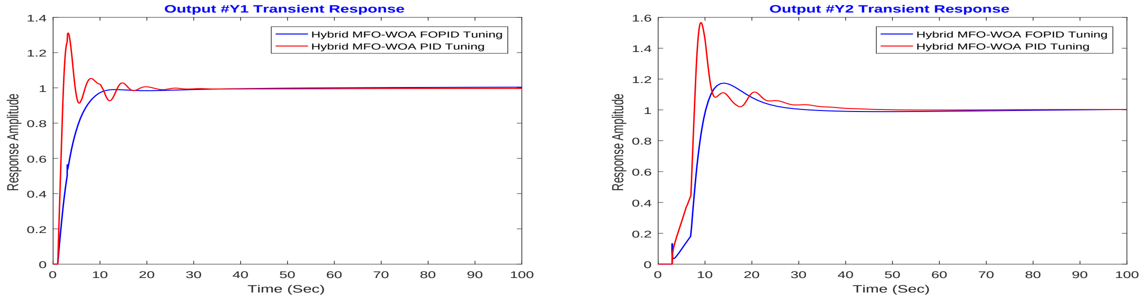
Figure 4.12: TITO (Wood-Berry) system.

Consider the RGA matrix [123, 148], the pairing of loop is $1 - 1/2 - 2$ pair is employed. As the multivariable system has a practice of decoupler to minimize the interaction of the loops. The proposed method is eliminating the requirement of decoupler in the MIMO system. In this, each loop possesses an individual FOPID controller is used to control the corresponding output and minimization of interactions.

The relative gain array (Λ) matrix is obtained as follows in Eq.(4.32)

$$\Lambda = \begin{bmatrix} 2.0094 & -1.0094 \\ -1.0094 & 2.0094 \end{bmatrix} \tag{4.32}$$

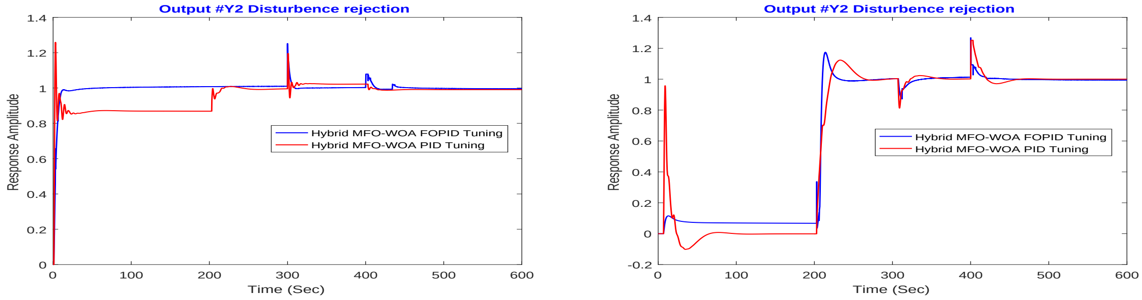
The proposed FOPID controller parameters gain in the each loop are obtained for Wood-Berry TITO system using hybrid MFO-WOA algorithm within the boundary conditions. According to this method, the interaction is minimum and the system reference tracking response is much better, as shown in Figure 4.13a for Y_1 response and Figure 4.13b for Y_2 response, respectively. To determine some practical applicability of the proposed method, the TITO process system (Wood-Berry system) has disturbances in the loops. The loop disturbances are present at time $t = 250s$ in the $1 - 1$ loop and $t = 400s$ in the $2 - 2$ loop for $600s$ of simulation time. The proposed method has quick disturbance rejection ability, as shown in Figure 4.14a for Y_1 output disturbance rejection and Figure 4.14b for Y_2 output disturbance rejection.



(a) Y1 (Loop1)

(b) Y2 (Loop2)

Figure 4.13: Reference tracking of WB (TITO) system with FOPID controller design



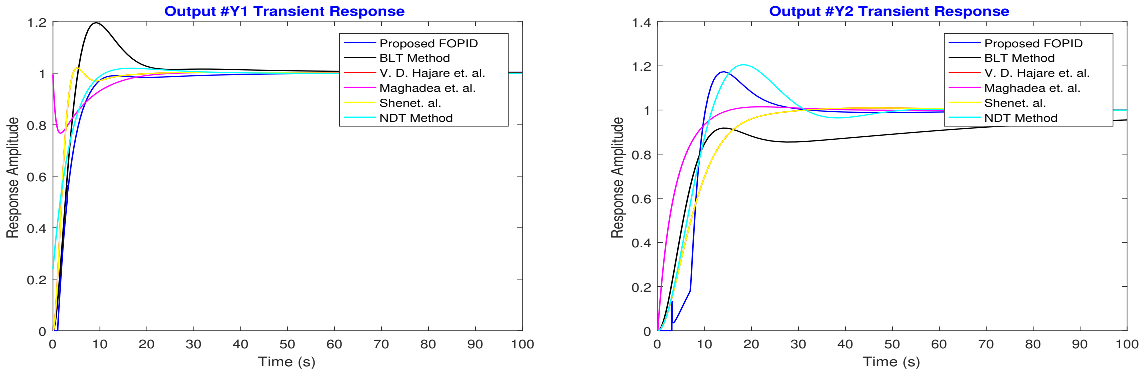
(a) Y1 (Loop1)

(b) Y2 (Loop2)

Figure 4.14: Disturbance rejection of WB (TITO) system with FOPID controller design

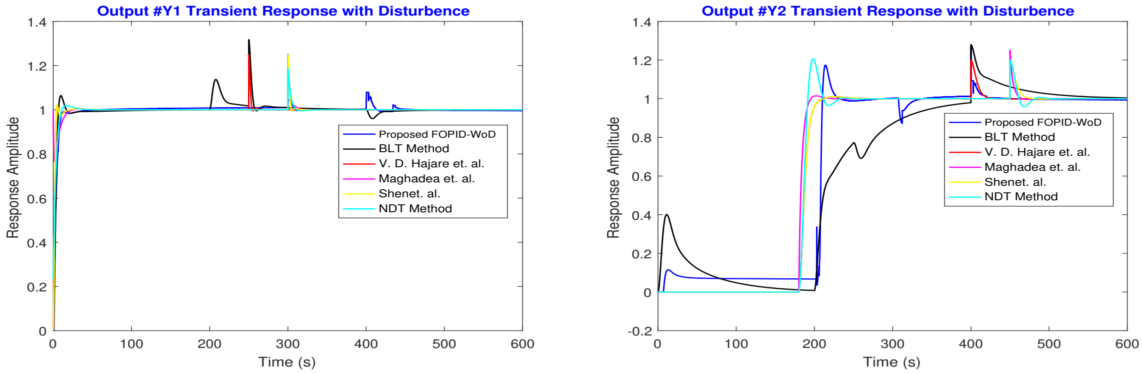
The proposed method's performance is compared with some classical, conventional methods from literature like Maghadea et al. [60], BLT method [122], Hajare et al. [123], NDT tuning method [130] and Shen et al. [131]. The proposed optimization-based FOPID controller parameter tuning method of the TITO system is more superior to the other set-point tracing techniques and disturbance elimination techniques.

The results of the proposed controller design are compared to the literature controller design. As compared to this method, the interaction is minimum and the system reference tacking response is optimal, as shown in Figure 4.15a for Y_1 response and Figure 4.15b for Y_2 response respectively. To estimate the proposed method's practical applicability, the TITO process system has disturbances in the loops. In a $1-1$ loop, the input step changes at time $t = 0s$ and the loop disturbance is present at time $t = 300s$, as shown in Figure 4.16a for Y_1 output. In a $2-2$ loop, the input step changes at time $t = 200s$ and the loop disturbance is at time $t = 400s$, as shown in Figure 4.16b for Y_2 output, for a total $600s$ of simulation time. The proposed method has quick reference tracking of step changes and best disturbance rejection ability, as shown in Figure 4.15a to Figure 4.16b respectively.



(a) Y1 (Loop1) (b) Y2 (Loop2)

Figure 4.15: Comparison of reference tracking in WB (TITO) system with FOPID controller design



(a) Y1 (Loop1) (b) Y2 (Loop2)

Figure 4.16: Comparison of disturbance rejection in WB (TITO) system with FOPID controller design

The control performance in each loop given by the proposed controllers is energy-efficient and the possibility of an actuator saturation problem. The FOPID controller's analysis is compared with literature controllers to conclude the proposed method has the best transient response performance. In the controller design, each loop's integral square error value is used in proposed FOPID controllers to have optimal controller tuning parameters. Therefore, actuators' saturation changes are less in the case of FOPID controllers in multi-loop control systems. From reference tracking performance, disturbance rejection performance and control action performance, in all aspects MFO-WOA algorithm based FOPID controller is giving better performance than that of other controllers. The proposed MFO-WOA based FOPID controllers have the advantage of no decoupler to minimize the interaction, less overshoot and quick step change response in each loop of the MIMO system.

The tabulation in Table 4.7 shows the numerical results in comparison study of the proposed scheme with some existing methods like Maghadea et al. [60], BLT method [122], Hajare et al. [123], NDT tuning method [130] and Shen et al. [131] are attained from the existing literature.

Table 4.7: FOPID parameters and performance of W-B TITO system (Time in Sec)

Tuning Methods	PID/FOPID_1	PID/FOPID_2	Rise Time	Overshoot	Settling Time
	[Kp1 Ki1 λ1 Kd1 μ1]	[Kp2 Ki2 λ2 Kd2 μ2]	[Tr1 Tr2]	[MP1 MP2]	[Ts1 Ts2]
Proposed FOPID	**[0.895 0.165 0.573 0.321 0.283 1.34]**	**[-0.0102 -0.0434 0.6103 -0.1894 0.875]**	**[1.705 3.21]**	**[0.27 5.57]**	**[8.095 15.41]**
Proposed PID	**[0.1127 0.1033 NA 0.1006 NA]**	**[-0.0153 -0.0249 NA -0.0105 NA]**	**[2.05 5.13]**	**[1.02 7.65]**	**[9.284 16.186]**
BLT [122]	[0.375 0.045 NA 0 NA]	[-0.075 -0.003 NA 0 NA]	[3.74 9.57]	[16.42 0.52]	[23.91 NA]
Hajare [123]	[0.657 0.027 NA 4.301 NA]	[0.312 0.013 NA 2.175 NA]	[2.7 13.53]	[2.08 1.01]	[11.16 22.42]
Maghadea [60]	[0.973 0.088 NA 2.688 NA]	[0.313 0.0304 NA 0.807 NA]	[2.72 8.75]	[1.27 1.51]	[15.59 11.98]
Shen [131]	[0.0618 0.0025 NA 0.34 NA]	[0.02062 0.00085 NA 0.1141 NA]	[2.71 13.5]	[2.08 1.01]	[11.16 22.42]
NDT [130]	[0.41 0.074 NA – NA]	[0.120 0.024 NA – NA]	[6.56 8.39]	[1.96 0.59]	[9.18 43.57]

4.5 Summary

This chapter presents a novel hybrid MFO-WOA algorithm for solving global optimization of complex problems and optimal FOPID/PID controller design method. The proposed hybrid MFO-WOA algorithm has validated using a 23-benchmark test function to get an optimal global solution and has a better fast convergence capability than the other optimization algorithms. Then, the proposed hybrid MFO-WOA algorithm has used in the control system application of FOPID/PID controller tuning parameters for industrial systems. Further, the proposed algorithm-based FOPID/PID controller design for the FOPTD system, AVR system, and TITO Wood-Berry systems, the FOPID/PID controller parameters tuning ability by the proposed algorithm was achieved in terms of a comparative study of controller design performance, then proposed controller efficiency is better. However, all these case studies of controller designs are proven to be better in SISO/MIMO systems performance.

Chapter 5

Implementation of PID controller and experimental validation of IoT application

In this chapter, the real-time hardware implementation of single tank liquid level control system with different algorithm based PID controller designs and IoT based applications.

5.1 Introduction

Liquid level control of chemical processes is a common problem in industrial process control. Many industries rely on liquid level controllers for the proper functioning of the plant. The bottle filling systems, dairy industries, petrochemical plants, water purification plants, pharmaceutical industries and boilers require the liquid level controllers at various stages. Use of liquid level controllers in industrial applications was given in [149]. Traditionally, these liquid controllers are controlled and monitored using PLC/SCADA based platforms. Nowadays, these are being replaced by IoT based systems. The use of IoT technology in the industrial systems has advantages like remote monitoring and control, on-line fault diagnosis and provides security. Because of these merits, IoT based industrial process control has become an active research area.

Various controlling strategies have been used in the literature to solve this problem by considering different operating conditions. Most of these controllers use the PID controller with different types of tuning rules to identify K_p, K_i and K_d parameters. [150] presented the nonlinear approach for the design of P, PI and PID controllers for the coupled tank system.

Tuning of PID controller for Water level control of a single tank system using MATLAB was described in [150]. A Model Predictive Control (MPC) strategy was proposed by[151] to control the liquid level in the coupled tank system. In addition to traditional approaches, the PID controllers can also be tuned using various optimization algorithms. An analytical method was used in [152] to design a proportional-integral controller for the optimal control of the liquid level loop. An approach to improve the PID controller parameters for liquid level control was mentioned in [153]. Two different types of sliding mode controllers were developed by [154] to control the liquid flow in the coupled tank system. A methodology to control nonlinear liquid level system using a neural network based reinforcement learning approach was implemented in [155]. A fuzzy logic-based liquid level controller for a single tank system using SCADA was developed in [156]. Particle Swarm Optimization (PSO) based PID controller for controlling liquid flow in a two-tank system was presented in [157]. A genetic algorithm was used by [158] for PID controller parameter identification in on-line mode to control liquid level. Moreover, the various meta-heuristic optimization algorithms were proposed in the literature in control system designs.

Due to the rapid developments in the IoT, new approaches came into existence to solve process control problems. A brief overview of the IoT and various events that are happened recently in this field was described in [159]. Because of these advancements, remote monitoring and control of multiple processes have become a reality.

5.2 Integration of IoT in Control System

The invention of the Internet of Things (IoT) is regarded as one of the key innovations in the technological revolution of today. For most organizations that seek to achieve greater efficiency and control of processes and properties, digital transformation has become unavoidable.

Innovation has reached a point in today's tech-driven era, where computers have started to replace humans. Constant research and progress have been made to restore relaxation and add meaning to our lives. The IoT is one of those technological advances that plays an important role in facilitating. Compared to the previous initiation of new ventures, the IoT has reached the next stage, where the focus is mainly on integration and convergence across industrial verticals. Industries are more focused on the importance and priorities of connected devices than on their versatility. The key factor taken into account is developments taking place in IoT embedded devices, platforms, applications, frameworks, apps, and networking, among others. Owning or creating instruments and equipment with versatility in design, movement, re-usability, and interconnectivity is essential for manufacturers. Increasingly, manufacturers rely on the integration of subsystems and modular modules that can help increase performance and ultimately reduce production costs.

A smart system of control based on IoT technologies has been implemented [160]. It is possible to explain the control device scheme in two ways: first a wireless router, computers, smartphones, etc. A machine or smartphone is used over the Internet as a remote controller. Second, it consists of a smart central controller, switch modules, modules for RF, modules for modification, modules for the environment, data collectors, etc. A linking connection between the previous and the following is the smart central controller in that it integrates the first part and the second part together. It is the duty of these flexible control modules to control appliances and communicate with the central controller. When such a control device is mounted in a home, the householder may use a smartphone to track home appliances and conduct certain operations remotely, such as turning a light on or off. Thanks to the following four benefits, the smart control system is very useful [161]. First of all, all the control modules have been built to be easily mounted on walls in regular sizes. Secondly, a WSAN is made up of the smart central controller and all these modules, which means that no additional wiring is needed. Thirdly, by self-configuration and self-organization, the smart home control system retains a dynamic balance. Fourthly, it is easy to setup and manages the smart home control system.

5.3 Industrial prototype plant description

5.3.1 Modeling of cylindrical single tank

A mathematical explanation of the dynamic behavior of the process systems [162] needs to be explained. However, the mathematical model of most physical processes is not linear [121]. On the other hand, most methods for the analysis, simulation, and design of the controllers presume that the process dynamics are linear. To connect this extent, a linearization of the nonlinear model is often required. This linearization refers to a specific operating point of the system. Throughout this section, we will explain the nonlinear mathematical behavior of the method and the linearization of the model. Here, exercise a typical example of a liquid level system to explain the behavior of the processing system. The cylindrical tank structure can originate employing a differential equation of the first order equation to obtain the model. The liquid level in the tank must be determined as a SISO [163] liquid level system.

In Figure 5.1, let F_i and F_o be the inflow rate and outflow rate (m^3/sec) of the tank, and h is the height of the liquid level at any time. Suppose the cross-section area of the tank is A. In a steady-state, both F_i and F_o are the same, and the height H of the tank should be constant. The mathematical model [164] can be written as follows according to the mass balance equation,

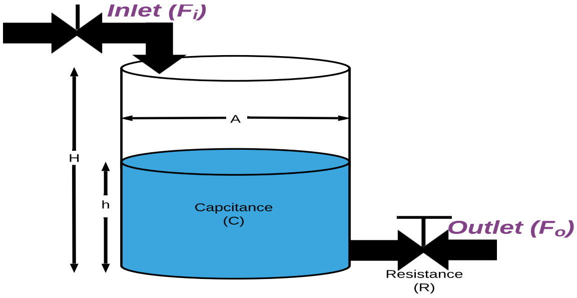


Figure 5.1: A schematic diagram of the liquid level system.

Rate of change of fluid volume in tank = flow in flow out

$$F_i - F_o = \tfrac{dV}{dt} \quad whare\ V = A \times h$$

$$F_i - F_o = A\frac{dh}{dt} \tag{5.1}$$

The cross sectional area of the tank can be replaced by capacitance

$$F_i - F_o = C\frac{dh}{dt} \tag{5.2}$$

Where the resistance R may be written as

$$R = C\tfrac{dh}{dt} = \tfrac{h}{F_o}$$

Then rearranging the above equation, to get

$$F_o = \frac{h}{R} \tag{5.3}$$

Substitute Eq.(5.3) in Eq.(5.2), to get

$$C\frac{dh}{dt} = F_i - \frac{h}{R} \tag{5.4}$$

After simplifying the Eq.(5.4) becomes

$$RC\frac{dh}{dt} + h = RF_i \tag{5.5}$$

Taking Laplace transform of Eq.(5.5) by considering the initial conditions to zero

$$RCsH(s) + H(s) = RF_i(s) \tag{5.6}$$

The transfer function can be obtained as

$$\frac{H(s)}{F_i(s)} = \frac{R}{RCs + 1} \quad whaer\ RC = \tau \tag{5.7}$$

where gain (R) depends on the resistance of outlet pipe and time constant (τ) relies on the resistance of the outlet pipe and cross-sectional area (C). The transfer function of the system is obtained as in Eq.5.8, by considering the liquid tank specifications are listed in Table 5.1.

$$\frac{H(s)}{F_i(s)} = \frac{0.2099e^{-7s}}{619.78s + 1} \tag{5.8}$$

Table 5.1: Specifications of Liquid process plant

S.no	Parameters	Value
1	Max. flow rate	750LPH
2	Tank Height	60CM
3	Diameter	14CM
4	Surface area	153.86CM
5	Steady state	45CM
6	delay	7s

5.3.2 Optimization based controller design

The advancement of the control system design problems switches between evolutionary optimization algorithms and adaptive mechanism, either of which can be applied directly to replace traditional techniques. Otherwise, they can be used in conjunction with conventional approaches and benefit from both perspectives. Advanced optimization algorithms can not have sufficient convergence, distinct from traditional techniques, but provide a wide variety of problem areas. The optimization methods begin by looking at the work of optimizing the multiminimal objective. The algorithm is then modified to determine the PID controller's parameters to control the given system, as shown in Figure 5.2 adequately.

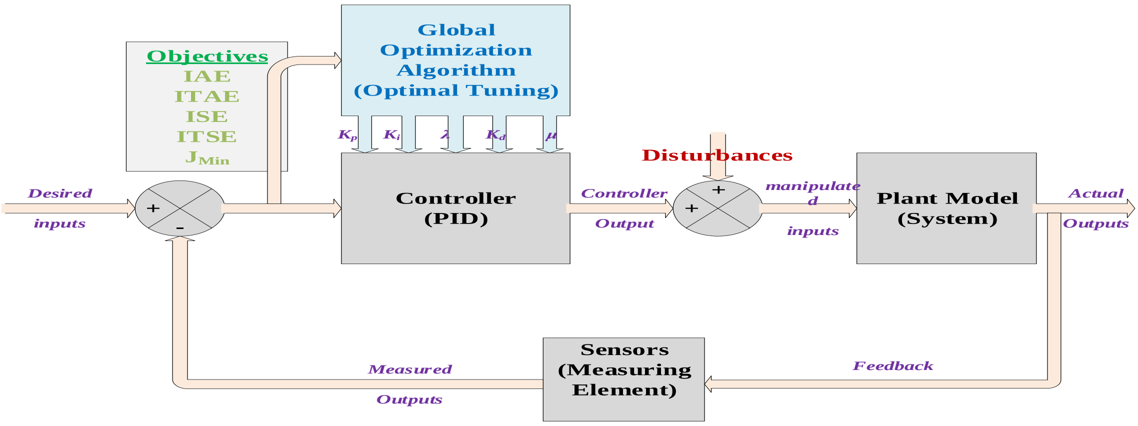


Figure 5.2: Problem formulation of Optimization based control system.

Several performance indices exist for controller design in the literature for analysis and design [86]. In this controller design, a transient response performance principle is also applied, as given by Gaing et.al. in Eq.(5.9). In the objective, time domain specifications are included by a weighting factor β. The performance measures in the time domain response include overshoot (M_P), rise time (T_R), settling time (T_S), and steady-state error (E_{SS}). Hence, a new performance measure J$_{Min}$(K) [32] is expressed in Eq.(5.9)

$$J_{Min}(K) = (1 - e^{-\beta})(M_P + E_{SS}) + e^{-\beta}(T_S - T_R) \tag{5.9}$$

where K is [K_P, K_i, K_d] for PID control and β is the weighting factor in the range of 0.7 to 1.4.

5.4 Proposed method

5.4.1 mGWO algorithm

According to Mirjalili et al. [165], the gray wolf stay together and hunt in groups. The search and hunting method can be summarized as follows [166]: (1) If the prey is recognized, it is first tracked and hunted and approached. (2) When the prey runs, the gray wolf pursues, encircle, and attack the prey until it stops moving. (3) In the end, the assault begins.

As described in the above section, the GWO gray wolf encircles the prey during the hunt. The hunting function is accomplished by the equations as follows:

$$\overrightarrow{D} = \left|\overrightarrow{C} \cdot \overrightarrow{X_p}(t) - \overrightarrow{X}(t)\right| \tag{5.10}$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X_p}(t) - \overrightarrow{A} \cdot \overrightarrow{D} \tag{5.11}$$

where t indicates the current iteration, $\overrightarrow{X_p}$ is the positional vector of prey, $\overrightarrow{X}$ is the positional vector of grey wolf and $\overrightarrow{A}$, $\overrightarrow{C}$ are coefficient vectors can be defined as in Eq.(5.12).

$$\overrightarrow{A} = 2\overrightarrow{a} \cdot \overrightarrow{r}_1 - \overrightarrow{a} \tag{5.12}$$

$$\overrightarrow{C} = 2\overrightarrow{r}_2 \tag{5.13}$$

where $\overrightarrow{a}$ can be decreasing linearly from 2 to 0 over the iterations as given in Eq.(5.14) and the other vector $\overrightarrow{C}$ is also modified as in Eq.(5.15)

$$a = 2(1 - \frac{t^r}{MaxIt^r}) \tag{5.14}$$

$$C = 2(1 - \frac{t^{r_2}}{MaxIt^{r_2}`}) \tag{5.15}$$

where $r = 1$ to 6 and r_2 are random vectors in [0, 1].

The variance of r is throughout maximum iterations, which helps calculate the value of r for a small increase in the number of iterations allocated for exploration. On the other hand, as the value of r increases, non-linearity increases, thus increasing the algorithm's overall ability to explore. To maintain the right relationship between discovery and exploration, the value of r is chosen to be 2 in the Mittal et al. [167] for this algorithm.

Gray wolf can identify and encircle the position of the prey. The alpha usually directs hunting. Beta and delta can also regularly engage in hunting. However, in an abstract search space, we have no idea where the optimum (prey) is determined. To predict the hunting behavior of gray wolfs mathematically, it is believed that the alpha (best solution) beta and delta have an excellent knowledge of the prey's possible location. Therefore, save the first three best solutions to date and oblige the other search agents to update their positions according to the best search agents' place. The following formulas are suggested here.

$$\overrightarrow{D}_\alpha = \left|\overrightarrow{C}_1 \cdot \overrightarrow{X}_\alpha - \overrightarrow{X}\right|,\ \overrightarrow{D}_\beta = \left|\overrightarrow{C}_2 \cdot \overrightarrow{X}_\beta - \overrightarrow{X}\right|,\ \overrightarrow{D}_\delta = \left|\overrightarrow{C}_3 \cdot \overrightarrow{X}_\delta - \overrightarrow{X}\right| \tag{5.16}$$

$$\overrightarrow{X}_1 = \overrightarrow{X}_\alpha - \overrightarrow{A}_1 \cdot \overrightarrow{D}_\alpha,\ \overrightarrow{X}_2 = \overrightarrow{X}_\beta - \overrightarrow{A}_2 \cdot \overrightarrow{D}_\beta,\ \overrightarrow{X}_3 = \overrightarrow{X}_\delta - \overrightarrow{A}_3 \cdot \overrightarrow{D}_\delta \tag{5.17}$$

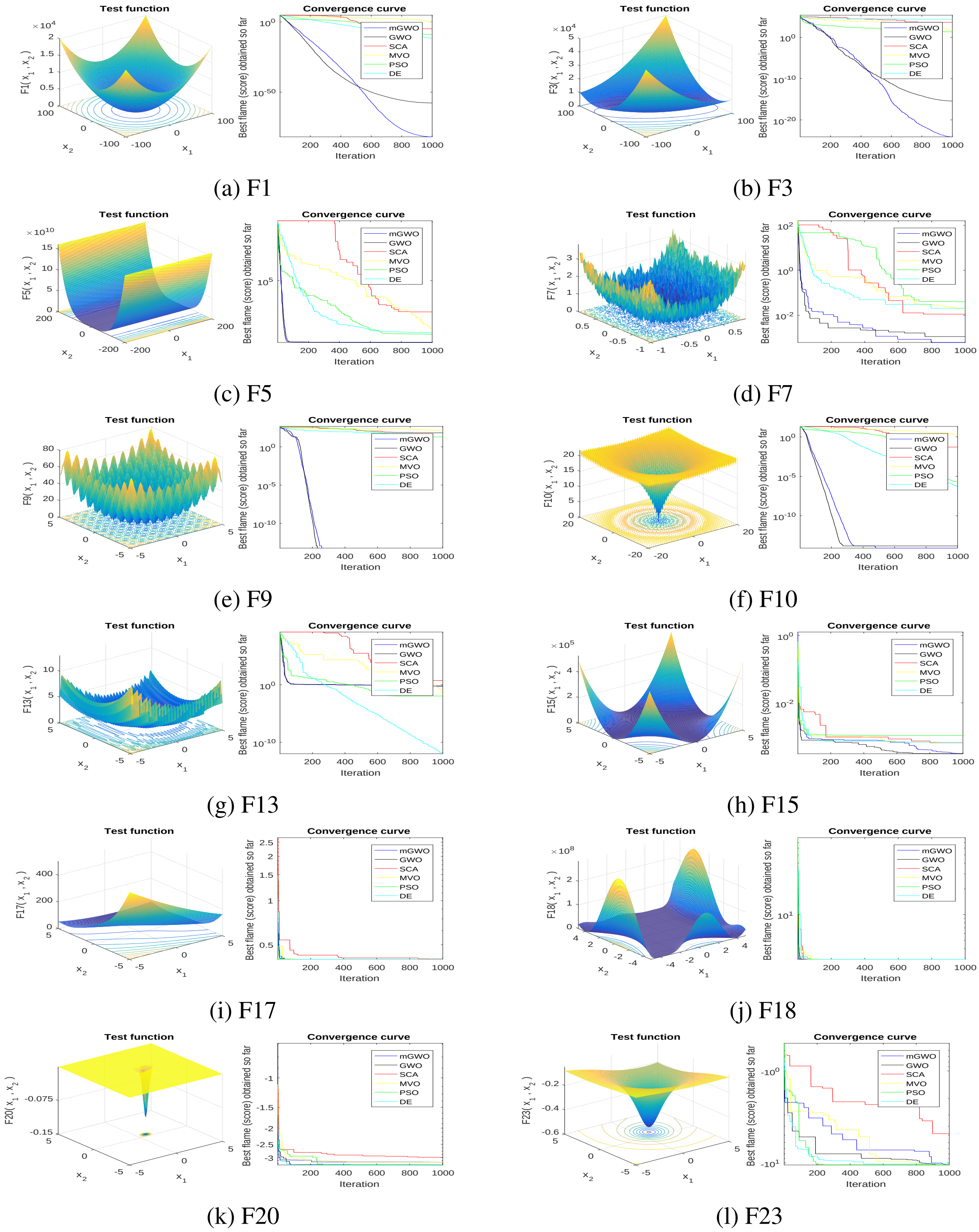
$$\overrightarrow{X}(t+1) = \frac{\overrightarrow{X}_1 + \overrightarrow{X}_2 + \overrightarrow{X}_3}{3} \tag{5.18}$$

Exploration and exploitation are caused while scanning and attacking the prey using parameters $\overrightarrow{a}$ and $\overrightarrow{C}$. Parameter $\overrightarrow{a}$ is of from 2 to 0 to balance discovery and production. The gray wolfs deviate from each other in search of prey when $\left|\overrightarrow{A}\right| > 1$ and $\left|\overrightarrow{A}\right| < 1$ converge to attack each other. Randomness helps to prevent being stuck in the local minima.

Adaptive values of $\overrightarrow{a}$ the parameter sustain discovery to avoid being stuck in the local optima and to deal with accuracy. It is a crucial parameter in the fine-tuning of the solution vectors and can theoretically be used to change the algorithm's convergence rate. The mGWO algorithm is proposed to improve efficiency to avoid being stuck in pre-mature convergence, convergence rate and accuracy. The advances are made in the proposed mGWO algorithm to address the issues of performance and convergence. The parameter $\overrightarrow{a}$ is known to be a random vector in the range [0-1] whose values are essential to the balance between discovery and exploitation.

mGWO Validation results

All unique optimization algorithms need to be validated on benchmark functions and analyzed among different conventional optimization algorithms. The performance of mGWO is related to additional algorithms for 12 standard and accessible benchmark functions out of 23 functions [109] to test the effectiveness of the suggested algorithm. The simulation effects of 12 benchmark functions are analyzed to evaluate the performance proposed mGWO algorithm over Grey Wolf Optimizer (GWO) [165], Sine Cosine algorithm (SCA) [68], Multi-verse optimizer (MVO) [168], Particle Swarm Optimization (PSO) [71] and Differential Evolution (DE) [126]. The convergence rate of mGWO, GWO, SCA, MVO, PSO and DE algorithms has examined for 12 functions shown in Figure 5.3. The fast convergence proves that the ability of mGWO to obtain a better global optimum throughout iterations. However, the results show that the potential of mGWO in solving optimization problems as superior to the other algorithms.



(a) F1

(b) F3

(c) F5

(d) F7

(e) F9

(f) F10

(g) F13

(h) F15

(i) F17

(j) F18

(k) F20

(l) F23

Figure 5.3: Benchmark functions convergence performance

The numerical results listed in Table 5.2, gives the algorithm's best, mean, worst, standard deviation and median values over 30 independent runs. Therefore, there is continual feasibility that the control may have occurred unexpectedly, even including a weak probability within 30 runs.

Table 5.2: Statistial results of 30 independent runs over $NP = 50$.

	Best	Average	SD	Median	Worst
F1	1.10E-85	1.61E-81	5.04E-81	3.29E-84	1.60E-80
F3	1.75E-22	7.31E-21	1.50E-20	1.25E-21	4.91E-20
F5	26.21647	26.96052	0.539466	27.1446	27.93874
F7	0.000176	0.000542	0.000317	0.000455	0.000978
F9	0	0	0	0.00E+00	0
F10	4.44E-15	7.64E-15	1.12E-15	7.99E-15	7.99E-15
F13	0.202296	0.399805	0.223561	0.362373	0.954864
F15	0.000308	0.004422	0.008407	0.000332	0.020363
F17	0.397887	0.398041	3.22E-04	0.397889	0.398754
F18	3	3.000005	6.46E-06	3.000002	3.00002
F20	-3.32199	-3.28168	0.086827	-3.32196	-3.08211
F23	-10.5354	-10.5322	0.003118	-10.5323	-10.527

5.4.2 ESP32 Module

The ESP32 microcontroller module was developed as an integrated platform for Wi-Fi, Bluetooth and Internet of Things (IoT) applications [169]. The module consists of Xtensa dual-core 32-bit microprocessor, which supports up to 600 MIPS with a maximum clock rate of 80MHz. The module has 34 programmable GPIO ports and includes 18 channel 12-bit ADC, Two 8-bit DAC, along with the support of SPI, I2C, UART and CAN communication protocols. It supports famous wireless communication protocols like IEEE 802.11/b/g/n and Bluetooth and standard versions.

The proposed method is implemented using the ESP32 embedded hardware platform. The plant uses the HART communication protocol to send the sensor values to the computing devices [170]. The plant Differential Pressure Meter generates a current proportional to the water level, in the range of (5-20)mA. This current is converted to corresponding voltage (analog) using I/V converter to the range of (0-5)V. This signal is given to the internal ADC of the ESP32 module. The ADC was set to 10-bit resolution to generate the values between 0-1023 for the voltage range of 0-5V. The signal was fed as input to the PID controller program based on which the controller causes the controlling signal. This controlling signal was used to drive the pneumatic activator of the plant using PWM (Pulse Width Modulation). The pneumatic valve responds to the input current in the range of (5-20)mA. Therefore the controlling signal value generated by the ESP32 module needs to be converted to current. For this purpose, the V/I converter was used to interface the ESP32 module to the pneumatic valve. The PWM is set in an 8-bit mode with an operating frequency of 5KHz. The V/I converter generates the equivalent current proportional to the PWM value. The PID controller was programmed into the ESP32 module using Arduino IDE. The block diagram of proposed method for a single tank liquid level controller is shown in Figure 5.4.

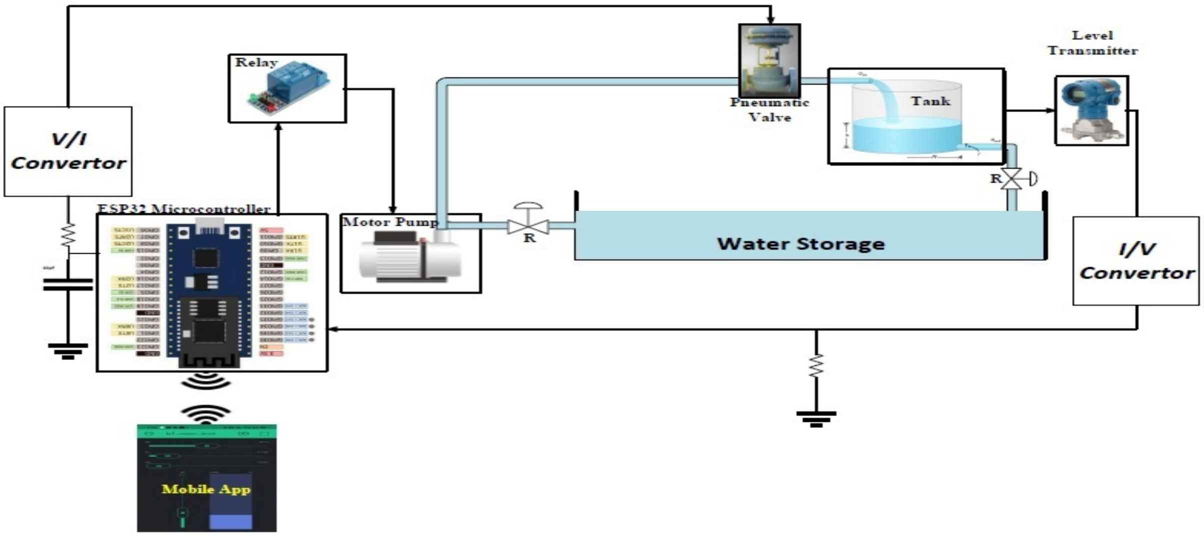


Figure 5.4: Block diagram of proposed method of process control.

5.4.3 Implementation of IoT

The automatic control process in persistent production processes is a combination of computer engineering and electrical engineering disciplines using industrial control systems to achieve a manufacturing level of performance, efficiency and protection that could not be accomplished only by manual control. This automation is widely applied in industries such as oil refining, pulp and paper manufacturing, chemical processing, and power generation plants.

The extensive industrial plant process management has developed over several phases. The logical concept was for all plant measurements to be sent to a permanently operated central control room. This was mostly the centralization of all the regional tables, with the benefits of lower manning rates and a more uncomplicated process overview. Both automatic and manual control outputs were regularly transmitted back to the factory. Nevertheless, this system was inflexible while providing a central control emphasis, as each control loop had its controller and continuous operator movement within the control was needed to view various parts of the operation.

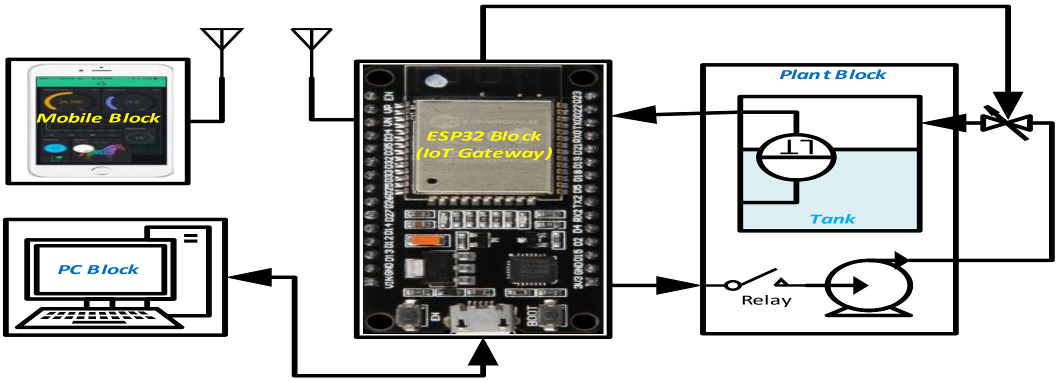


Figure 5.5: Schematic diagram of proposed IoT based process control.

The real-time implementation of liquid level control using a PID controller in the ESP32 microcontroller is proposed, as shown in Figure 5.5. The plan for real-time implementation is as follows. The pump ON/OFF can be controlled using the ESP32 through a relay operation. However, the reference liquid level in a tank and PID controller parameters are given from an external user interface using the Internet. These values are read into the ESP32 microcontroller with wi-fi and the control signal will be generated according to the reference defined by the user. This control signal is conditioned to adjust the pneumatic valve of the plant to achieve the desired liquid level in the tank. This valve will control the liquid inflow rate into the tank. Since, the plant consists of a differential pressure meter, the liquid level in the tank can be continuously read in the ESP32 microcontroller. It can be monitor in the user interface mobile app as an IoT application.

Even though, the core logic of the controller (PID) algorithm runs on the ESP32 module, to provide the limited access of the plant, the real-time development of an IoT based Android application for the liquid level control using the Blynk platform [171]. The application consists of a user interface to set the controller gains(Kp, Ki, Kd), the setpoint, and the liquid level indicator and ON/OFF button for the pump (motor). The parameter values in the Android app can be transmitted to the ESP32 module over the internet using wi-fi with Blynk backend. The interfacing code for the Android app and the ESP32 module was developed using Arduino IDE.

The working of IoT based plant controller is as follows, open the app from mobile and Turn on the plant pump using the ON/OFF button. Then set K_p, K_i, K_d and setpoint of the PID controller in the app using the slider bars. These values will be sent to the ESP32 module using the Blynk IoT platform. The controller will control the water level in the tank according to the setpoint in the app. During the operation, the tank liquid level's value will be sent back to the application using IoT platform from ESP32. For this purpose, the app liquid level indicator was added, which shows a real-time water level in the tank. The complete hardware implemented system in real-time is shown in the Figure 5.6.

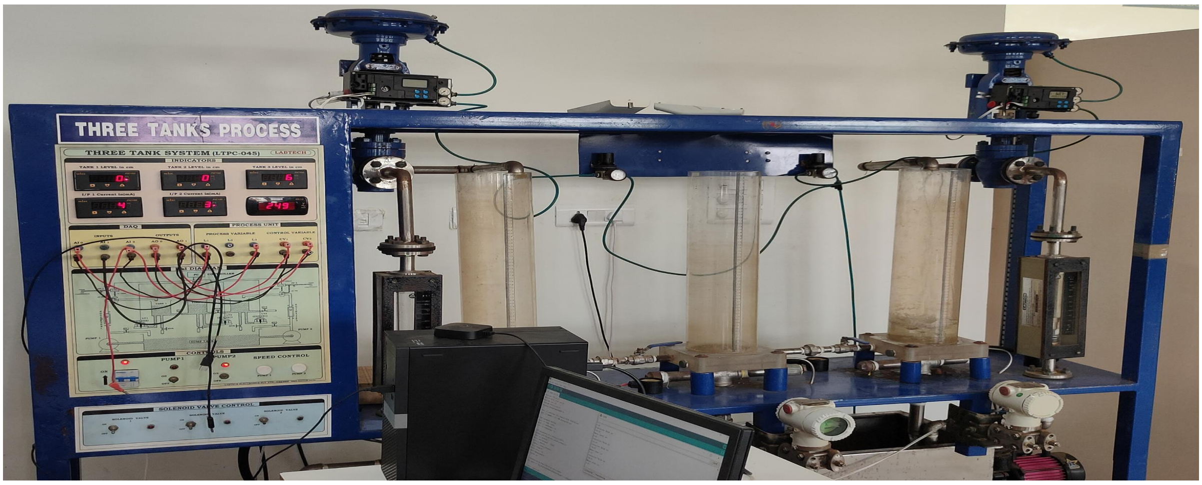
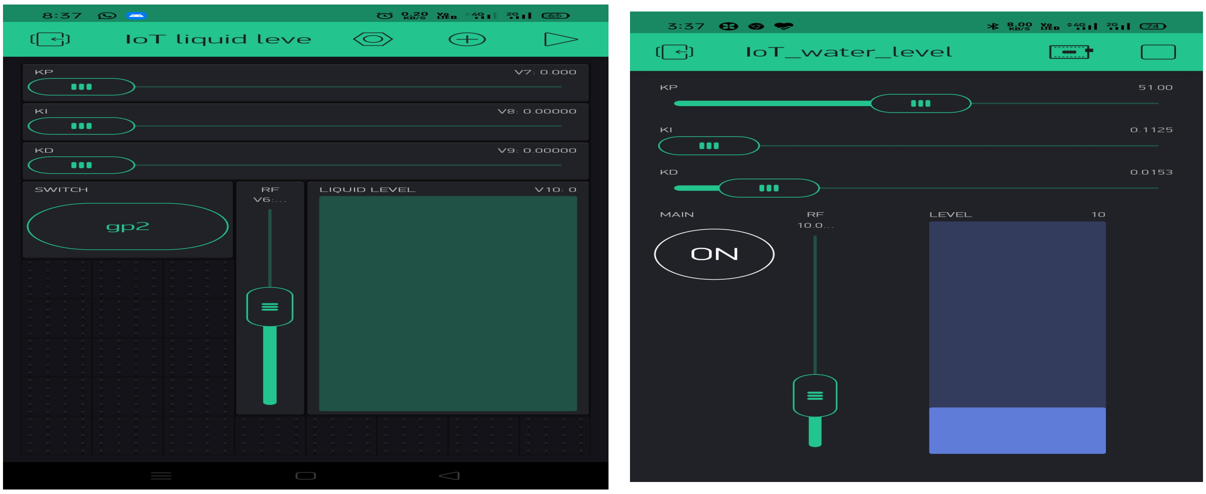


Figure 5.6: Real-time plant for the proposed method.

5.5 Results & Discussion

The steady-state response of the system for the maximum flow rate of 750LPH in the cylindrical tank is given in Figure 5.6. The system has reached the steady-state at 13cm height and took 470s at a flow rate of 450LPH.


IoT liquid leve

KP V7: 0.000

KI V8: 0.00000

KD V9: 0.00000

SWITCH gp2

RF V6...

LIQUID LEVEL V10: 0

IoT_water_level

KP 51.00

KI 0.1125

KD 0.0153

MAIN ON

RF 10.0..

LEVEL 10

(a) Interface of control and monitor.

(b) Level monitor in the App.

Figure 5.7: User interface App

The proposed algorithm is implemented in MATLAB on a desktop with an Intel Core i5 processor with 8GB RAM. Once the PID controller parameters (Kp, Ki, Kd) are obtained from the user interface as shown in Figure 5.7a, these values are given to the ESP32 module using IoT as presented in section 4. Initially, the setpoint is set at a 5cm level and the maximum flow rate was fixed to 750LPH. When the plant is in use, the controller generated PWM with a 92.4% duty cycle. This value was maintained up to 30cm water level. After this level, again the controller started action and reduced the duty cycle to 70% and finally, when the setpoint is reached, the controller maintained a duty cycle between 40%-50%. The finite value of the duty cycle at the setpoint indicates a steady outflow in the tank as shown in Figure 5.7b. To compensate for this outflow, the controller producing a 40%-50% duty cycle PWM signal to the valve. The process was repeated for 10cm and 14cm, and similar controller action was observed.

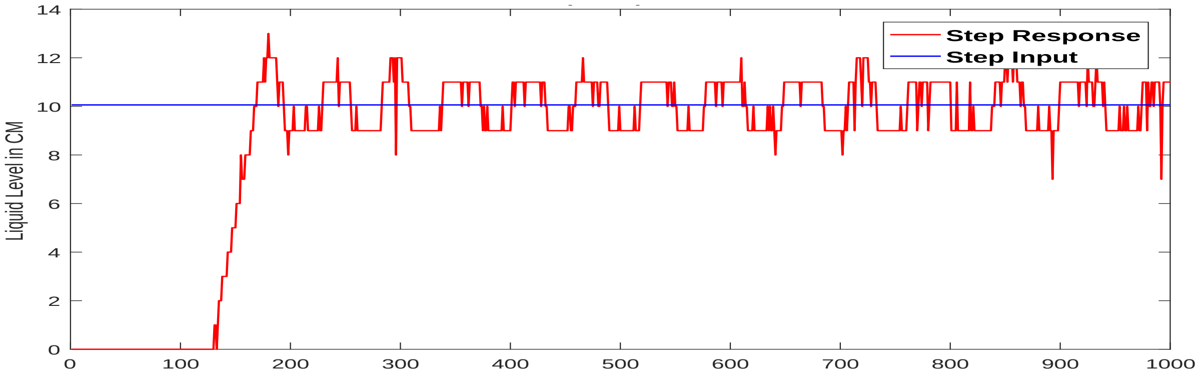


Figure 5.8: Response of proposed controller for different set points

The response of implemented method time response is obtained as shown in Figure 5.8. The proposed method is compared with ZieglerNichols and SIMC tuning algorithms. Figure 5.9 shows that the proposed method was performed better in terms of settling time, rise time, overshoot, and steady-state error.

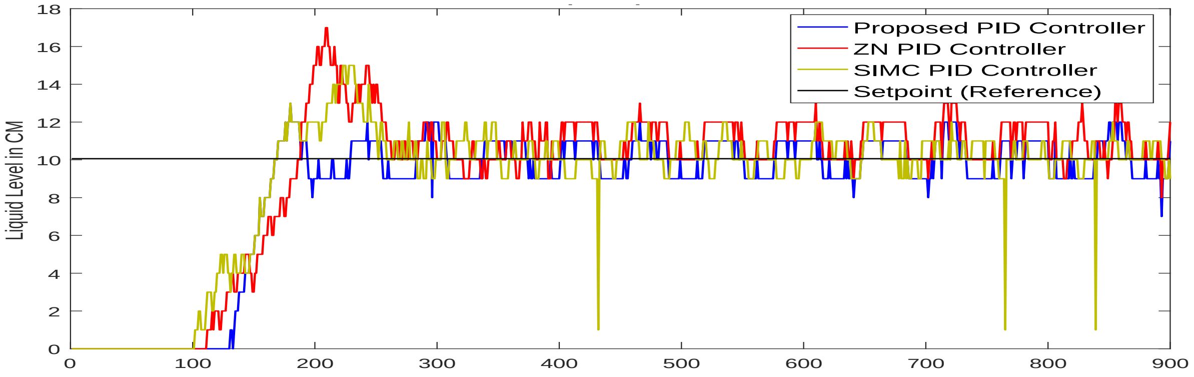


Figure 5.9: Comparison of controllers.

The response of the proposed controller for different set points is shown in Figure 5.10. From the figure, it can be observed that the controller was able to track all the set points precisely.

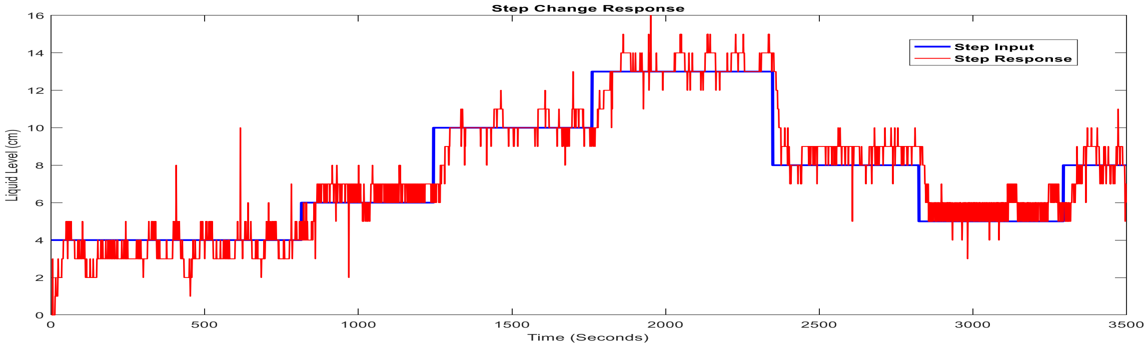


Figure 5.10: Response of proposed controller for different set points

The corresponding inflow rate and Pneumatic valve positions during setpoint tracking were observed. The valve position changes between 0-92.5 and the inflow rate vary between 90-750 LPH. Figure 5.11 shows that during the dips in the valve position, the inflow rates were reduced. It indicates that the valve position controls the water flow into the tank.

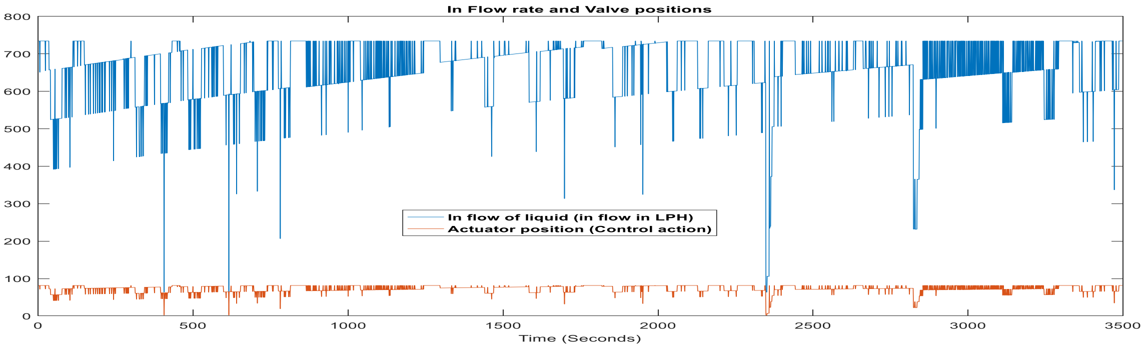


Figure 5.11: Inflow rate and Valve position of plant

Later to test the robustness of the controller, during the controlling action external disturbance was added. The controller quickly responded for the external disturbance as shown in Figure 5.12 by decreasing the inflow rate into the tank. The numerical performance of the controller is obtained and listed in the Table 5.3.

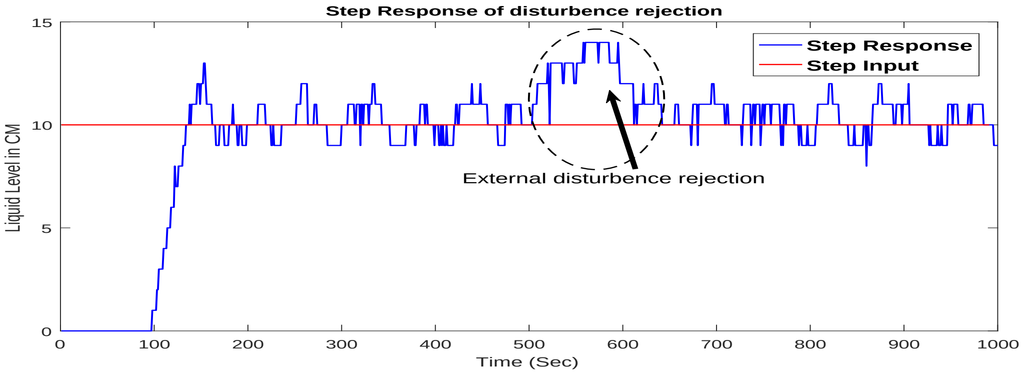


Figure 5.12: Inflow rate and Valve position of plant

Table 5.3: Proposed method numerical results.

Methods	PID Controller Parameters			PID Controller Performance		
	K_p	K_i	K_d	T_r	T_s	M_p
ZN Tuning	31.26	0.112	52.38	51.03	227.79	14.77
SIMC Tuning	38.97	1.159	72.52	34.65	129.43	9.84
Proposed mGWO-PID	**56.312**	**0.175**	**0.2175**	**27.31**	**104.15**	**3.71**

5.6 Summary

This chapter reports an efficient method to implement a PID controller for the liquid level system and experimental validation of integrating IoT application. Here, the industrial prototype plant of the liquid level system is considered for experimental validation in real-time. However, the cylindrical liquid level is controlled and monitored continuously using the proposed method. The PID controller is designed using mGWO algorithm for liquid level control and integration of the ESP32 module for monitoring using IoT application. The proposed IoT application in control system is successfully demonstrated in real-time for industrial systems.

Chapter 6

Conclusions and Future Scope

6.1 Conclusions

In this Thesis, the proposed methods are straightforward to explore the optimal parameters of a FOPID/PID controller to give the desired response at the system's output. Specific objective functions are used to construct a FOPID / PID controller and a time-domain objective function. However, the time domain objective function obtains the best optimal parameters of FOPID/PID controller design.

The FOPID/PID controller is designed for SISO/MIMO systems using several conventional and proposed optimization algorithms. Here, the proposed optimization algorithms are verified using 23 benchmark functions with their analysis. Finally, the same algorithm is used to design the FOPID/PID controllers. Though, the proposed FOPID/PID controller is much better than other methods of controllers.

A fractional-order PID controller parameter tuning technique for TITO (MIMO) system is proposed based on the optimization algorithm. In this multivariable system, interactions are minimized without decoupler by the proposed controller design. The FOPID controller design was studied by applying to Wood-Berry distillation (TITO) processes to achieve the desired performance with a controller design objective function of different functions. The simulation results of the proposed method and various methods are compared to the controller's verification of performance norms. This method gives better set-point tracking and quickly rejects the disturbances by minimum interactions in the TITO system's loops.

A novel optimization algorithm for solving global optimization problems and optimal FOPID/PID controller designs for some case study of benchmark systems are designated to validate experimentally, including integrating of IoT application. The proposed hybrid MFO-WOA algorithm is validated and has a better fast convergence capability than the other optimization algorithms. The proposed hybrid MFO-WOA algorithm is used in the control problem of FOPID/PID controller designs in case studies are discussed. The FOPID/PID controller parameters tuning ability by this hybrid MFO-WOA algorithm is achieved in terms of a comparative study of controller design performance, the proposed controller's efficiency is presented.

In the process control, experimental validation is performed with the proposed controller in integrating IoT applications. The PID controller design using mGWO is implemented on the ESP32 microcontroller module and an Android-based application is developed using the Blynk platform for monitoring and control of plants using IoT applications. However, the IoT application-based process control is more predominant of multidisciplinary technology in monitoring and control. A single tank liquid level system is considered to utilize the IoT connectivity and control systems have presented as real-time implementation. The overall system has control and monitoring using integration of IoT based applications.

6.2 Future Scope

During the thesis work, some issues were understood. Maybe these issues are carried out for future scope. A specific description of the identified problems is provided in the following.

The proposed methods of optimal FOPID/PID controller design have limits to off-line tunings, and as in the future, these controllers can be tuned on-line. The optimal tuning of the FOPID/PID controller can be included in the IoT application to avoid the off-line tuning of controllers. Therefore, there is a scope of on-line automatic controller designs to full fill the complete IoT-based applications. Additionally, the automated controller parameter tuning scheme for multivariable control systems will be adequately useful.

Also, a usual design for the embedded implementation of fractional system identification can be studied. However, the reference model is not produced on the prototype, but IIR filters can hard-coded. These filter designs can begin up for further possibilities in the success of influential and precise automatic tuning algorithms.

Appendix A

A.0.1 Special functions

1. Gamma Function

   The gamma function is the extension of the factorial for non-integer numbers and defined by the following equation.

$$\Gamma(z) = \int_0^\infty e^{-u} u^{z-1} du \qquad \forall z \in \mathbb{R} \tag{A.1}$$

2. Beta Function

   The beta function defined as

$$\beta(p, q) = \int_0^1 (1-u)^{p-1} u^{q-1} du \qquad \forall p,\ q \in \mathbb{R}+ \tag{A.2}$$

   The relation between beta and gamma function given in Eq. (A.3)

$$\beta(p, q) = \frac{\Gamma(p)\Gamma(q)}{\Gamma(p+q)} \qquad \forall p,\ q \in \mathbb{R}+ \tag{A.3}$$

3. Mittag-Leffler Function

   The Mittag-Leffler (ML) function most frequently used in the solution of fractional order differential equations similar to the exponential functions as the solution of integer order ordinary differential equations. It is actually a generalized higher transcendental which reduces to the commonly used transcendentals for special cases. The one parameter Mittag-Leffler function defined as in Eq. (A.4)

$$E_\alpha(z) = \sum_{K=0}^{\infty} \frac{z^K}{\Gamma(\alpha K + 1)} \qquad \alpha > 0 \tag{A.4}$$

The Mittag-Leffler function reduces to exponential function for $\alpha = 1$. The two-parameter Mittag-Leffler function defined as in Eq. (A.5)

$$E_{\alpha,\beta}(z) = \sum_{K=0}^{\infty} \frac{z^K}{\Gamma(\alpha K + \beta)} \qquad \alpha, \beta > 0 \tag{A.5}$$

The following relation holds between the one and two parameter Mittag-Leffler function in Eq. (A.6)

$$E_{\alpha,1}(z) = \sum_{K=0}^{\infty} \frac{z^K}{\Gamma(\alpha K + 1)} = E_{\alpha}(z) \tag{A.6}$$

Some commonly used transcendental that can be obtain from the two-parameter ML function are shown below. The variation in the second parameter of the Mittag-Leffler function yields

$$E_{1,1}(z) = \sum_{K=0}^{\infty} \frac{z^K}{\Gamma(K + 1)} = \sum_{K=0}^{\infty} \frac{z^K}{K!} = e^z \tag{A.7}$$

$$E_{1,2}(z) = \sum_{K=0}^{\infty} \frac{z^K}{\Gamma(K + 2)} = \sum_{K=0}^{\infty} \frac{z^K}{(K + 1)!} = \frac{1}{z} \sum_{K=0}^{\infty} \frac{z^{K+1}}{(K + 1)!} = \frac{e^z - 1}{z} \tag{A.8}$$

$$E_{1,3}(z) = \sum_{K=0}^{\infty} \frac{z^K}{\Gamma(K + 3)} = \sum_{K=0}^{\infty} \frac{z^K}{(K + 2)!} = \frac{1}{z^2} \sum_{K=0}^{\infty} \frac{z^{K+1}}{(K + 2)!} = \frac{e^z - 1 - z}{z^2} \tag{A.9}$$

$$E_{2,1}(z) = \sum_{K=0}^{\infty} \frac{z^{2K}}{\Gamma(2K + 1)} = \sum_{K=0}^{\infty} \frac{z^{2K}}{(2K)!} = cosh\ (z) \tag{A.10}$$

$$E_{2,2}(z) = \sum_{K=0}^{\infty} \frac{z^{2K}}{\Gamma(2K + 2)} = \sum_{K=0}^{\infty} \frac{z^{2K}}{(2K + 1)!} = \frac{sinh\ (z)}{z} \tag{A.11}$$

A.0.2 Fractional order approximations

In Table A.1, approximations of $\frac{1}{s^{\alpha}}$ have been given for $\alpha \in \{0.1, 0.2, ..., 0.9\}$ with maximum discrepancy of $2dB$ within $(0.01, 100)$ rad/sec frequency range (Ahmad & Sprott, 2003).

Table A.1: Approximation of $\frac{1}{s^{\alpha}}$ for different α values.

Fractional order	**Approximated transfer function**
$s^{0.1}$	$\frac{1584.89(s+0.1668)(s+27.83)}{(s+0.1)(s+16.68)(s+s+2783)}$
$s^{0.2}$	$\frac{79.4328(s+0.05623)(s+1)(s+17.78)}{(s+0.03162)(s+0.5623)(s+10)(s+177.8)}$
$s^{0.3}$	$\frac{39.8107(s+0.0416)(s+0.3728)(s+3.34)(s+29.94)}{(s+0.02154)(s+0.1931)(s+1.73)(s+15.51)(s+138.9)}$
$s^{0.4}$	$\frac{35.4813(s+0.03831)(s+0.261)(s+1.778)(s+12.12)(s+82.54)}{(s+0.01778)(s+0.1212)(s+0.8254)(s+5.623)(s+38.31)(s+261)}$
$s^{0.5}$	$\frac{15.8489(s+0.03981)(s+0.2512)(s+1.585)(s+10)(s+63.1)}{(s+0.01585)(s+0.1)(s+0.631)(s+3.981)(s+3.981)(s+25.12)(s+158.5)}$
$s^{0.6}$	$\frac{10.7978(s+0.04642)(s+0.3162)(s+2.154)(s+14.68)(s+100)}{(s+0.01468)(s+0.1)(s+0.631)(s+4.642)(s+31.62)(s+215.4)}$
$s^{0.7}$	$\frac{9.3633(s+0.06449)(s+0.578)(s+5.179)(s+46.42)(s+416)}{(s+0.01389)(s+0.1245)(s+1.116)(s+10)(s+89.62)(s+803.1)}$
$s^{0.8}$	$\frac{5.3088(s+0.1334)(s+2.371)(s+42.17)(s+749.9)}{(s+0.01334)(s+0.2371)(s+4.217)(s+74.99)(s+1334)}$
$s^{0.9}$	$\frac{2.2675(s+1.292)(s+215.4)}{(s+0.01292)(s+2.154)(s+359.4)}$

A.0.3 Benchmark functions

Twenty-three bench mark test functions for optimization algorithms:

Function	Formulation	D	Range	f_{min}
Unimodal Functions				
Sphere (F1)	$f_1(x) = \sum_{i=1}^{D} x_i^2$	30	[-100, 100]	0
Schwefel 2.22 (F2)	$f_2(x) = \sum_{i=1}^{D} \lvert x_i \rvert + \prod_{i=1}^{D} \lvert x_i \rvert$	30	[-10, 10]	0

Schwefel 1.2 (F3)	$f_3(x) = \sum_{i=1}^{D}\left(\sum_{j=1}^{i} x_j^2\right)$	30	[-100, 100]	0
Schwefel 2.21 (F4)	$f_4(x) = \overset{max}{i}\{\lvert x_i\rvert, 1 \leqslant i \leqslant D\}$	30	[-100, 100]	0
Rosenbrock (F5)	$f_5(x) = \sum_{i=1}^{D-1}\left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right]$	30	[-30, 30]	0
Step (F6)	$f_6(x) = \sum_{i=1}^{D}(x_i + 0.5)^2$	30	[-100, 100]	0
Quartic (F7)	$f_7(x) = \sum_{i=1}^{D} ix_i^4 + rand[0, 1]$	30	[-1.28, 1.28]	0
Multimodal high-dimensional Functions				
Schwefel (F8)	$f_8(x) = 418.9829D + \sum_{i=1}(x_i sin\sqrt{\lvert x_i\rvert})$	30	[-500, 500]	-418.98
Rastrigin (F9)	$f_9(x) = \sum_{i=1}^{D}(x_i^2 - 10cos(2\pi x_i)) + 10D$	30	[-5.12, 5.12]	0
Ackley (F10)	$f_{10}(x) = -20exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) - exp\left(\frac{1}{D}\sum_{i=1}^{D} cos(2\pi x_i)\right) + 20 + e$	30	[-32, 32]	0
Griewank (F11)	$f_{11}(x) = \frac{1}{400}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D} cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	[-600, 600]	0
Penalize (F12)	$f_{12}(x) = \frac{\pi}{D}\left\{10sin(\pi y_i) + \sum_{i=1}^{D-1}(y_i - 1)^2\left[1 + 10sin^2(\pi y_{i+1})\right]\right\} + \frac{\pi}{D}(y_D - 1)^2 + \sum_{i=1}^{D} u(x_i, 10, 100, 4)$ $where\ y_i = 1 + \frac{x_i + 1}{4}\ and\ u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m if x_i > a \\ 0 if -a < x_i < a \\ k(-x_i - a)^m if x_i < -a \end{cases}$	30	[-50, 50]	0

Penalize2 (F13)	$f_{13}(x) = 0.1\left\{sin^2(3\pi x_i) + \sum_{i=1}^{D}(x_i - 1)^2\left[1 + sin^2(3\pi x_i) + 1\right]\right\}$ $+0.1\left\{(x_n - 1)^2[1 + sin^2(2\pi x_n)]\right\} + \sum_{i=1}^{D} u(x_i, 5, 100, 4)$	30	[-50, 50]	0
Multimodal low-dimensional Functions				
Foxholes (F14)	$f_{14}(x) = \frac{1}{50} + \sum_{j=1}^{25} \frac{1}{j+\sum_{i=1}^{D}(x_i-a_{ij})^6}$	2	[-65.5, 65.5]	0.9984
Kowalik (F15)	$f_{15}(x) = \sum_{i=1}^{11}\left[a_i - \frac{x_1(b_i^2+b_i x_2)}{b_i^2+b_i x_3+x_4}\right]^2$	4	[-5, 5]	0.000375
Six-hump Camel-Back (F16)	$f_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1 x_2 - 4x_2^2 + 4x_2^4$	2	[-5, 5]	-1.031
Branin (F17)	$f_{17}(x) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)cos x_1 + 10$	2	[-5, 5]	0.398
Goldstein-Price (F18)	$f_{18}(x) = \left[1 + (x_1 + x_2 + 1)^2(10 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)\right]$ $\left[30 + (2x_1 - 3x_2^2)(18 - 32x_1 + 12x_1 62 + 48X_2 - 36X_1X_2 + 27X_2^2 + 27X_2^2)\right]$	2	[-5, 5]	3
Hartman3 (F19)	$f_{19}(x) = -\sum_{i=1}^{4} c_i\, exp\left[-\sum_{j=1}^{D} a_{ij}(x_j - a_{ij})^2\right]$	3	[-5, 5]	-3.862
Hartman6 (F20)	$f_{20}(x) = -\sum_{i=1}^{4} c_i\, exp\left[-\sum_{j=1}^{D} a_{ij}(x_j - a_{ij})^2\right]$	6	[-5, 5]	-3.322
Shekel5 (F21)	$f_{21}(x) = -\sum_{i=1}^{D}\left[(x_i - a_i)(x - a_i)^T) + c_i\right]^{-1}$	5	[-5, 5]	-10.15
Shekel7 (F22)	$f_{22}(x) = -\sum_{i=1}^{D}\left[(x_i - a_i)(x - a_i)^T) + c_i\right]^{-1}$	7	[-5, 5]	-10.42
Shekel10 (F23)	$f_{22}(x) = -\sum_{i=1}^{D}\left[(x_i - a_i)(x - a_i)^T) + c_i\right]^{-1}$	10	[-5, 5]	-10.53

Bibliography

[1] Wikipedia, *Control engineering*, (accessed on December 3, 2016), https://en.wikipedia.org/wiki/Control_engineering.

[2] A. T. Azar, *Handbook of research on advanced intelligent control engineering and automation.* IGI Global, 2014.

[3] M. J. Blondin, P. M. Pardalos, and J. S. Sáez, *Computational Intelligence and Optimization Methods for Control Engineering.* Springer, 2019.

[4] S. Ramalingam, K. Baskaran, and D. Kalaiarasan, "Iot enabled smart industrial pollution monitoring and control system using raspberry pi with blynk server," in *2019 International Conference on Communication and Electronics Systems (ICCES).* IEEE, 2019, pp. 2030–2034.

[5] N. H. Carreras Guzman, M. Wied, I. Kozine, and M. A. Lundteigen, "Conceptualizing the key features of cyber-physical systems in a multi-layered representation for safety and security analysis," *Systems Engineering*, vol. 23, no. 2, pp. 189–210, 2020.

[6] M. Salhaoui, A. Guerrero-González, M. Arioua, F. J. Ortiz, A. El Oualkadi, and C. L. Torregrosa, "Smart industrial iot monitoring and control system based on uav and cloud computing applied to a concrete plant," *Sensors*, vol. 19, no. 15, p. 3316, 2019.

[7] S. Boverie, D. D. Cho, H. Hashimoto, M. Tomizuka, W. Wei, and D. Zühlke, "Mechatronics, robotics and components for automation and control ifac milestone report," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 10 800–10 809, 2008.

[8] D. Sreekantha and A. Kavya, "Agricultural crop monitoring using iot-a study," in *2017 11th International Conference on Intelligent Systems and Control (ISCO).* IEEE, 2017, pp. 134–139.

[9] R. Vilanova and A. Visioli, *PID control in the third millennium.* Springer, 2012.

[10] Y. Chen, I. Petras, and D. Xue, “Fractional order control-a tutorial,” in *2009 American control conference*. IEEE, 2009, pp. 1397–1411.

[11] A. Tepljakov, *Fractional-order modeling and control of dynamic systems*. Springer, 2017.

[12] A. Kilbas, *Theory and applications of fractional differential equations*.

[13] M. D. Ortigueira, *Fractional calculus for scientists and engineers*. Springer Science & Business Media, 2011, vol. 84.

[14] F. Padula, A. Visioli *et al.*, *Advances in robust fractional control*. Springer, 2015.

[15] J.-D. Gabano and T. Poinot, “Fractional modeling applied to non destructive thermal characterization,” *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 13 972–13 977, 2011.

[16] J. Wang, “Realizations of generalized warburg impedance with rc ladder networks and transmission lines,” *Journal of the Electrochemical Society*, vol. 134, no. 8, p. 1915, 1987.

[17] I. Podlubny, *Fractional differential equations: an introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications*. Elsevier, 1998.

[18] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, and V. Feliu-Batlle, *Fractional-order systems and controls: fundamentals and applications*. Springer Science & Business Media, 2010.

[19] T. T. Hartley, C. F. Lorenzo, and H. K. Qammer, “Chaos in a fractional order chua’s system,” *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 42, no. 8, pp. 485–490, 1995.

[20] B. M. Vinagre, Y. Q. Chen, and I. Petráš, “Two direct tustin discretization methods for fractional-order differentiator/integrator,” *Journal of the franklin institute*, vol. 340, no. 5, pp. 349–362, 2003.

[21] K. Oldham and J. Spanier, *The fractional calculus theory and applications of differentiation and integration to arbitrary order*. Elsevier, 1974.

[22] C. M. Ionescu, *The human respiratory system: an analysis of the interplay between anatomy, structure, breathing and fractal dynamics*. Springer Science & Business Media, 2013.

[23] S. Dreyfus, “Richard bellman on the birth of dynamic programming,” *Operations Research*, vol. 50, no. 1, pp. 48–51, 2002.

[24] P. N. Paraskevopoulos, *Modern control engineering*. CRC Press, 2017.

[25] M. A. Johnson and M. H. Moradi, *PID control*. Springer, 2005.

[26] A. Visioli, *Practical PID control*. Springer Science & Business Media, 2006.

[27] D. Xue, *Fractional-order control systems: fundamentals and numerical implementations*. Walter de Gruyter GmbH & Co KG, 2017, vol. 1.

[28] J. M. Kimeu, “Fractional calculus: Definitions and applications,” 2009.

[29] S. Das, “Functional fractional calculus for system identification and controls,” 2008.

[30] I. Podlubny, L. Dorcak, and I. Kostial, “On fractional derivatives, fractional-order dynamic systems and pi/sup/spl lambda//d/sup/spl mu//-controllers,” in *Proceedings of the 36th IEEE Conference on Decision and Control*, vol. 5. IEEE, 1997, pp. 4985–4990.

[31] B. Boudjehem and D. Boudjehem, “Fractional pid controller design based on minimizing performance indices,” *IFAC-PapersOnLine*, vol. 49, no. 9, pp. 164–168, 2016.

[32] Z.-L. Gaing, “A particle swarm optimization approach for optimum design of pid controller in avr system,” *IEEE transactions on energy conversion*, vol. 19, no. 2, pp. 384–391, 2004.

[33] O. Katsuhiko, *Modern control engineering*. Pearson, 2010.

[34] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal control*. John Wiley & Sons, 2012.

[35] R. Caponetto, *Fractional order systems: modeling and control applications*. World Scientific, 2010, vol. 72.

[36] V. Bhambhani, “Optimal fractional order proportional and integral controller for processes with random time delays,” 2009.

[37] D. Matignon, “Stability properties for generalized fractional differential systems,” in *ESAIM: proceedings*, vol. 5. EDP Sciences, 1998, pp. 145–158.

[38] J. Sabatier, P. Lanusse, P. Melchior, and A. Oustaloup, “Fractional order differentiation and robust control design,” in *CRONE, H-infinity and Motion Control*. Springer, 2015.

[39] J. Baranowski, W. Bauer, M. Zagórowska, T. Dziwiński, and P. Piątek, "Time-domain oustaloup approximation," in *2015 20th International Conference on Methods and Models in Automation and Robotics (MMAR).* IEEE, 2015, pp. 116–120.

[40] D. Xue, C. Zhao, and Y. Chen, "A modified approximation method of fractional order system," in *2006 International Conference on Mechatronics and Automation.* IEEE, 2006, pp. 1043–1048.

[41] K. Oprzędkiewicz, W. Mitkowski, and E. Gawin, "An estimation of accuracy of oustaloup approximation," in *International Conference on Automation.* Springer, 2016, pp. 299–307.

[42] I. Manotas, C. Bird, R. Zhang, D. Shepherd, C. Jaspan, C. Sadowski, L. Pollock, and J. Clause, "An empirical study of practitioners' perspectives on green software engineering," in *2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE).* IEEE, 2016, pp. 237–248.

[43] M. El-Shorbagy, A. Mousa, and S. Nasr, "A chaos-based evolutionary algorithm for general nonlinear programming problems," *Chaos, Solitons & Fractals*, vol. 85, pp. 8–21, 2016.

[44] Y. Zhang, "Engineering design synthesis of sensor and control systems for intelligent vehicles," Ph.D. dissertation, California Institute of Technology, 2006.

[45] K. E. Parsopoulos and M. N. Vrahatis, "On the computation of all global minimizers through particle swarm optimization," *IEEE Transactions on evolutionary computation*, vol. 8, no. 3, pp. 211–224, 2004.

[46] 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, 1997.

[47] T. Perumal, S. K. Datta, and C. Bonnet, "Iot device management framework for smart home scenarios," in *2015 IEEE 4th Global Conference on Consumer Electronics (GCCE).* IEEE, 2015, pp. 54–55.

[48] V. Hardion, D. P. Spruce, A. M. Otero, J. Lidon-Simon, and M. Lindberg, "The internet of things and control system."

[49] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE Internet of Things journal*, vol. 1, no. 1, pp. 22–32, 2014.

[50] A. H. Alavi, P. Jiao, W. G. Buttlar, and N. Lajnef, "Internet of things-enabled smart cities: State-of-the-art and future trends," *Measurement*, vol. 129, pp. 589–606, 2018.

[51] Chapter, *Introduction to Control Systems*, (accessed on October 12, 2019), http://www.ent.mrt.ac.lk/~rohan/teaching/EN5001/Reading/DORFCH1.pdf.

[52] B. Khalfa and C. Abdelfateh, "Optimal tuning of fractional order pλ d μa controller using particle swarm optimization algorithm," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 8084–8089, 2017.

[53] Z. Shafiei and A. Shenton, "Tuning of pid-type controllers for stable and unstable systems with time delay," *Automatica*, vol. 30, no. 10, pp. 1609–1615, 1994.

[54] J. Ziegler and N. Nichols, "Optimum settings for automatic controllers," 1993.

[55] D. P. Atherton, "Pid controller tuning," *Computing & control engineering journal*, vol. 10, no. 2, pp. 44–50, 1999.

[56] N. Razmjooy, M. Khalilpour, and M. Ramezani, "A new meta-heuristic optimization algorithm inspired by fifa world cup competitions: theory and its application in pid designing for avr system," *Journal of Control, Automation and Electrical Systems*, vol. 27, no. 4, pp. 419–440, 2016.

[57] Y. Wang, D. Hur, H. Chung, N. Watson, J. Arrillaga, and S. Matair, "Design of an optimal pid controller in ac-dc power system using modified genetic algorithm," in *PowerCon 2000. 2000 International Conference on Power System Technology. Proceedings (Cat. No. 00EX409)*, vol. 3. IEEE, 2000, pp. 1437–1442.

[58] V. Mukherjee and S. Ghoshal, "Intelligent particle swarm optimized fuzzy pid controller for avr system," *Electric Power Systems Research*, vol. 77, no. 12, pp. 1689–1698, 2007.

[59] M. Rahimian and K. Raahemifar, "Optimal pid controller design for avr system using particle swarm optimization algorithm," in *2011 24th Canadian conference on electrical and computer engineering (CCECE)*. IEEE, 2011, pp. 000 337–000 340.

[60] D. Maghade and B. Patre, "Decentralized pi/pid controllers based on gain and phase margin specifications for tito processes," *ISA transactions*, vol. 51, no. 4, pp. 550–558, 2012.

[61] B. T. Jevtović and M. R. Mataušek, "Pid controller design of tito system based on ideal decoupler," *Journal of process control*, vol. 20, no. 7, pp. 869–876, 2010.

[62] W. L. Luyben, “Simple method for tuning siso controllers in multivariable systems,” *Industrial & Engineering Chemistry Process Design and Development*, vol. 25, no. 3, pp. 654–660, 1986.

[63] S. Nikita and M. Chidambaram, “Relay auto tuning of decentralized pid controllers for unstable tito systems,” *Indian Chemical Engineer*, vol. 60, no. 1, pp. 1–15, 2018.

[64] Q.-G. Wang, B. Huang, and X. Guo, “Auto-tuning of tito decoupling controllers from step tests,” *ISA transactions*, vol. 39, no. 4, pp. 407–418, 2000.

[65] T. N. L. Vu and M. Lee, “Independent design of multi-loop pi/pid controllers for interacting multivariable processes,” *Journal of Process control*, vol. 20, no. 8, pp. 922–933, 2010.

[66] A. R. Simpson, G. C. Dandy, and L. J. Murphy, “Genetic algorithms compared to other techniques for pipe optimization,” *Journal of water resources planning and management*, vol. 120, no. 4, pp. 423–443, 1994.

[67] I. BoussaïD, J. Lepagnot, and P. Siarry, “A survey on optimization metaheuristics,” *Information sciences*, vol. 237, pp. 82–117, 2013.

[68] S. Mirjalili, “Sca: a sine cosine algorithm for solving optimization problems,” *Knowledge-based systems*, vol. 96, pp. 120–133, 2016.

[69] N. Raju and P. Reddy, “Optimal tuning of fractional order pid controller for automatic voltage regulator system through genetic algorithm,” *International Journal of Engineering and Technology (IJET)*, vol. 8, no. 3, pp. 922–927, 2016.

[70] H. Mühlenbein, “Evolution in time and space–the parallel genetic algorithm,” in *Foundations of genetic algorithms*. Elsevier, 1991, vol. 1, pp. 316–337.

[71] R. Eberhart and J. Kennedy, “Particle swarm optimization,” in *Proceedings of the IEEE international conference on neural networks*, vol. 4. Citeseer, 1995, pp. 1942–1948.

[72] X. Jin and R. G. Reynolds, “Using knowledge-based evolutionary computation to solve nonlinear constraint optimization problems: a cultural algorithm approach,” in *Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406)*, vol. 3. IEEE, 1999, pp. 1672–1678.

[73] E. Atashpaz-Gargari and C. Lucas, “Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition,” in *2007 IEEE congress on evolutionary computation*. Ieee, 2007, pp. 4661–4667.

[74] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis, “Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers,” *IEEE Control Systems Magazine*, vol. 32, no. 6, pp. 76–105, 2012.

[75] H. Zhu, L. Li, Y. Zhao, Y. Guo, and Y. Yang, “Cas algorithm-based optimum design of pid controller in avr system,” *Chaos, Solitons & Fractals*, vol. 42, no. 2, pp. 792–800, 2009.

[76] F. Farhani, C. B. Regaya, A. Zaafouri, and A. Chaari, “Real time pi-backstepping induction machine drive with efficiency optimization,” *ISA transactions*, vol. 70, pp. 348–356, 2017.

[77] B. Ataşlar-Ayyıldız and O. Karahan, “Tuning of fractional order pid controller using cs algorithm for trajectory tracking control,” in *2018 6th International Conference on Control Engineering & Information Technology (CEIT)*. IEEE, 2018, pp. 1–6.

[78] P. Shah and S. Agashe, “Review of fractional pid controller,” *Mechatronics*, vol. 38, pp. 29–41, 2016.

[79] D. Simon, “Biogeography-based optimization,” *IEEE transactions on evolutionary computation*, vol. 12, no. 6, pp. 702–713, 2008.

[80] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, “Gsa: a gravitational search algorithm,” *Information sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.

[81] S. Saremi, S. Mirjalili, S. Mirjalili, and J. S. Dong, “Grasshopper optimization algorithm: theory, literature review, and application in hand posture estimation,” in *Nature-Inspired Optimizers*. Springer, 2020, pp. 107–122.

[82] O. Bendjeghaba, “Continuous firefly algorithm for optimal tuning of pid controller in avr system,” *Journal of Electrical Engineering*, vol. 65, no. 1, pp. 44–49, 2014.

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

[84] S. Bouallègue, J. Haggège, and M. Benrejeb, “Particle swarm optimization-based fixed-structure h8 control design,” *International Journal of Control, Automation and Systems*, vol. 9, no. 2, p. 258, 2011.

[85] Y. Lu, D. Yan, and D. Levy, “Parameter estimation of vertical takeoff and landing aircrafts by using a pid controlling particle swarm optimization algorithm,” *Applied Intelligence*, vol. 44, no. 4, pp. 793–815, 2016.

[86] S. Chatterjee and V. Mukherjee, “Pid controller for automatic voltage regulator using teaching–learning based optimization technique,” *International Journal of Electrical Power & Energy Systems*, vol. 77, pp. 418–429, 2016.

[87] X. Li, Y. Wang, N. Li, M. Han, Y. Tang, and F. Liu, “Optimal fractional order pid controller design for automatic voltage regulator system based on reference model using particle swarm optimization,” *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 5, pp. 1595–1605, 2017.

[88] I. Podlubny, “Fractional-order systems and fractional-order controllers,” *Institute of Experimental Physics, Slovak Academy of Sciences, Kosice*, vol. 12, no. 3, pp. 1–18, 1994.

[89] S. Das, I. Pan, S. Das, and A. Gupta, “Improved model reduction and tuning of fractional-order piλdμ controllers for analytical rule extraction with genetic programming,” *ISA transactions*, vol. 51, no. 2, pp. 237–261, 2012.

[90] A. Sikander, P. Thakur, R. Bansal, and S. Rajasekar, “A novel technique to design cuckoo search based fopid controller for avr in power systems,” *Computers & Electrical Engineering*, vol. 70, pp. 261–274, 2018.

[91] H. Ramezanian, S. Balochian, and A. Zare, “Design of optimal fractional-order pid controllers using particle swarm optimization algorithm for automatic voltage regulator (avr) system,” *Journal of Control, Automation and Electrical Systems*, vol. 24, no. 5, pp. 601–611, 2013.

[92] M. Zamani, M. Karimi-Ghartemani, N. Sadati, and M. Parniani, “Design of a fractional order pid controller for an avr using particle swarm optimization,” *Control Engineering Practice*, vol. 17, no. 12, pp. 1380–1387, 2009.

[93] S. Chen, H. Xu, D. Liu, B. Hu, and H. Wang, “A vision of iot: Applications, challenges, and opportunities with china perspective,” *IEEE Internet of Things journal*, vol. 1, no. 4, pp. 349–359, 2014.

[94] J. A. Stankovic, “Research directions for the internet of things,” *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 3–9, 2014.

[95] S. R. Pokhrel and C. Williamson, “Modeling compound tcp over wifi for iot,” *IEEE/ACM transactions on networking*, vol. 26, no. 2, pp. 864–878, 2018.

[96] J. Kua, S. H. Nguyen, G. Armitage, and P. Branch, "Using active queue management to assist iot application flows in home broadband networks," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1399–1407, 2017.

[97] S. R. Pokhrel, H. L. Vu, and A. L. Cricenti, "Adaptive admission control for iot applications in home wifi networks," *IEEE Transactions on Mobile Computing*, 2019.

[98] A. Oustaloup, J. Sabatier, and P. Lanusse, "From fractal robustness to the crone control," 1999.

[99] A. Oustaloup, L. Le Lay, and B. Mathieu, "Identification of non integer order system in the time-domain," in *IEEECESA96, SMC IMACS Multiconference*, 1996.

[100] J. Trigeassou, T. Poinot, J. Lin, A. Oustaloup, and F. Levron, "Modeling and identification of a non integer order system," in *1999 European Control Conference (ECC)*. IEEE, 1999, pp. 2453–2458.

[101] I. Petras and L. Dorcak, "Fractional-order systems and piλdμ-controllers," *IEEE Transactions on Automatic Control*, vol. 44, no. 1, pp. 208–214, 1999.

[102] D. Xue, C. Zhao, and Y. Chen, "Fractional order pid control of a dc-motor with elastic shaft: a case study," in *2006 American control conference*. IEEE, 2006, pp. 6–pp.

[103] C. A. Monje, B. M. Vinagre, V. Feliu, and Y. Chen, "Tuning and auto-tuning of fractional order controllers for industry applications," *Control engineering practice*, vol. 16, no. 7, pp. 798–812, 2008.

[104] D. Valério and J. S. Da Costa, "Tuning of fractional pid controllers with ziegler–nichols-type rules," *Signal processing*, vol. 86, no. 10, pp. 2771–2784, 2006.

[105] R. Munasinghe, *Classical Control Systems: Design and Implementation*. Alpha Science International, 2012.

[106] R. C. Dorf and R. H. Bishop, *Modern control systems*. Pearson, 2011.

[107] R. Lahcene, S. Abdeldjalil, and K. Aissa, "Optimal tuning of fractional order pid controller for avr system using simulated annealing optimization algorithm," in *2017 5th International Conference on Electrical Engineering-Boumerdes (ICEE-B)*. IEEE, 2017, pp. 1–6.

[108] B. Bourouba, S. Ladaci, and H. Schulte, "Optimal design of fractional order $pi\lambda d\mu$ controller for an avr system using ant lion optimizer," *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 200–205, 2019.

[109] H. Nenavath and R. K. Jatoth, “Hybridizing sine cosine algorithm with differential evolution for global optimization and object tracking,” *Applied Soft Computing*, vol. 62, pp. 1019–1043, 2018.

[110] R. V. Rao and G. Waghmare, “A new optimization algorithm for solving complex constrained design optimization problems,” *Engineering Optimization*, vol. 49, no. 1, pp. 60–83, 2017.

[111] R. Rao, “Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems,” *International Journal of Industrial Engineering Computations*, vol. 7, no. 1, pp. 19–34, 2016.

[112] R. Rao and K. More, “Optimal design and analysis of mechanical draft cooling tower using improved jaya algorithm,” *International Journal of Refrigeration*, vol. 82, pp. 312–324, 2017.

[113] A. Ali and S. Majhi, “Pid controller tuning for integrating processes,” *ISA transactions*, vol. 49, no. 1, pp. 70–78, 2010.

[114] Z. Bingul and O. Karahan, “Comparison of pid and fopid controllers tuned by pso and abc algorithms for unstable and integrating systems with time delay,” *Optimal Control Applications and Methods*, vol. 39, no. 4, pp. 1431–1450, 2018.

[115] Z.Bingul and O.Karahan, “A novel performance criterion approach to optimum design of pid controller using cuckoo search algorithm for avr system,” *Journal of the Franklin Institute*, vol. 355, no. 13, pp. 5534–5559, 2018.

[116] S. Ekinci and B. Hekimoğlu, “Improved kidney-inspired algorithm approach for tuning of pid controller in avr system,” *IEEE Access*, vol. 7, pp. 39 935–39 947, 2019.

[117] C. Gong, “Jaya algorithm-optimized pid controller for avr system,” in *International Conference on Intelligent and Interactive Systems and Applications*. Springer, 2018, pp. 382–393.

[118] H. R. Pota, “Mimo systems-transfer function to state-space,” *IEEE transactions on education*, vol. 39, no. 1, pp. 97–99, 1996.

[119] V. Hajare and B. Patre, “Decentralized pid controller for tito systems using characteristic ratio assignment with an experimental application,” *ISA transactions*, vol. 59, pp. 385–397, 2015.

[120] P. Lanusse, J. Sabatier, and A. Oustaloup, "Extension of pid to fractional orders controllers: a frequency-domain tutorial presentation," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 7436–7442, 2014.

[121] A. Zaafouri, C. B. Regaya, H. B. Azza, and A. Châari, "Dsp-based adaptive backstepping using the tracking errors for high-performance sensorless speed control of induction motor drive," *ISA transactions*, vol. 60, pp. 333–347, 2016.

[122] C. S. Besta and M. Chidambaram, "Tuning of multivariable pi controllers by blt method for tito systems," *Chemical engineering communications*, vol. 203, no. 4, pp. 527–538, 2016.

[123] V. Hajare, B. Patre, A. Khandekar, and G. Malwatkar, "Decentralized pid controller design for tito processes with experimental validation," *International Journal of Dynamics and Control*, vol. 5, no. 3, pp. 583–595, 2017.

[124] C. Rajapandiyan and M. Chidambaram, "Controller design for mimo processes based on simple decoupled equivalent transfer functions and simplified decoupler," *Industrial & Engineering Chemistry Research*, vol. 51, no. 38, pp. 12 398–12 410, 2012.

[125] H. R. Mokadam, B. M. Patre, and D. K. Maghade, "Tuning of multivariable pi/pid controllers for tito processes using dominant pole placement approach," *International Journal of Automation and Control*, vol. 7, no. 1-2, pp. 21–41, 2013.

[126] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE transactions on evolutionary computation*, vol. 15, no. 1, pp. 4–31, 2010.

[127] M. Ali, M. Pant, and A. Abraham, "A modified differential evolution algorithm and its application to engineering problems," in *2009 International Conference of Soft Computing and Pattern Recognition*. IEEE, 2009, pp. 196–201.

[128] H. Nenavath and R. K. Jatoth, "Hybrid sca-tlbo: a novel optimization algorithm for global optimization and visual tracking," *Neural Computing and Applications*, vol. 31, no. 9, pp. 5497–5526, 2019.

[129] H. Wang, G. Zeng, Y. Dai, D. Bi, J. Sun, and X. Xie, "Design of a fractional order frequency pid controller for an islanded microgrid: A multi-objective extremal optimization method," *Energies*, vol. 10, no. 10, p. 1502, 2017.

[130] S. Tavakoli, I. Griffin, and P. J. Fleming, "Tuning of decentralised pi (pid) controllers for tito processes," *Control engineering practice*, vol. 14, no. 9, pp. 1069–1080, 2006.

[131] Y. Shen, Y. Sun, and S. Li, “Adjoint transfer matrix based decoupling control for multivariable processes,” *Industrial & Engineering Chemistry Research*, vol. 51, no. 50, pp. 16 419–16 426, 2012.

[132] Z. Kovacic and S. Bogdan, *Fuzzy controller design: theory and applications*. CRC press, 2018.

[133] Y.-S. Zhou and L.-Y. Lai, “Optimal design for fuzzy controllers by genetic algorithms,” *IEEE transactions on industry applications*, vol. 36, no. 1, pp. 93–97, 2000.

[134] H. Nenavath, R. K. Jatoth, and S. Das, “A synergy of the sine-cosine algorithm and particle swarm optimizer for improved global optimization and object tracking,” *Swarm and Evolutionary Computation*, vol. 43, pp. 1–30, 2018.

[135] R. M. R. Allah, “Hybridization of fruit fly optimization algorithm and firefly algorithm for solving nonlinear programming problems,” *International Journal of Swarm Intelligence and Evolutionary Computation*, vol. 5, no. 2, pp. 1–10, 2016.

[136] S. Mirjalili, “Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm,” *Knowledge-based systems*, vol. 89, pp. 228–249, 2015.

[137] S. Mirjalili and A. Lewis, “The whale optimization algorithm,” *Advances in engineering software*, vol. 95, pp. 51–67, 2016.

[138] J. D. Deaton and R. V. Grandhi, “A survey of structural and multidisciplinary continuum topology optimization: post 2000,” *Structural and Multidisciplinary Optimization*, vol. 49, no. 1, pp. 1–38, 2014.

[139] A. E. Hassanien, T. Gaber, U. Mokhtar, and H. Hefny, “An improved moth flame optimization algorithm based on rough sets for tomato diseases detection,” *Computers and Electronics in Agriculture*, vol. 136, pp. 86–96, 2017.

[140] L. Jourdan, M. Basseur, and E.-G. Talbi, “Hybridizing exact methods and metaheuristics: A taxonomy,” *European Journal of Operational Research*, vol. 199, no. 3, pp. 620–629, 2009.

[141] M. M. Majeed and S. R. Patri, “A hybrid of woa and mgwo algorithms for global optimization and analog circuit design automation,” *COMPEL-The international journal for computation and mathematics in electrical and electronic engineering*, 2019.

[142] B. Rosner, R. J. Glynn, and M.-L. Ting Lee, “Incorporation of clustering effects for the wilcoxon rank sum test: a large-sample approach,” *Biometrics*, vol. 59, no. 4, pp. 1089–1098, 2003.

[143] K. J. Åström and T. Hägglund, “The future of pid control,” *Control engineering practice*, vol. 9, no. 11, pp. 1163–1175, 2001.

[144] A. Ateş and C. Yeroglu, “Optimal fractional order pid design via tabu search based algorithm,” *ISA transactions*, vol. 60, pp. 109–118, 2016.

[145] A. O’Dwyer, *Handbook of PI and PID controller tuning rules*. Imperial college press, 2009.

[146] S. B. Chiranjeevi *et al.*, “Implementation of fractional order pid controller for an avr system using ga and aco optimization techniques,” *IFAC-PapersOnLine*, vol. 49, no. 1, pp. 456–461, 2016.

[147] B. Hekimoğlu, “Sine-cosine algorithm-based optimization for automatic voltage regulator system,” *Transactions of the Institute of Measurement and Control*, vol. 41, no. 6, pp. 1761–1771, 2019.

[148] G.-Q. Zeng, J. Chen, M.-R. Chen, Y.-X. Dai, L.-M. Li, K.-D. Lu, and C.-W. Zheng, “Design of multivariable pid controllers using real-coded population-based extremal optimization,” *Neurocomputing*, vol. 151, pp. 1343–1353, 2015.

[149] R. Zhang, S. Wu, and F. Gao, “Improved pi controller based on predictive functional control for liquid level regulation in a coke fractionation tower,” *Journal of Process Control*, vol. 24, no. 3, pp. 125–132, 2014.

[150] B. N. Getu, “Water level controlling system using pid controller,” *International Journal of Applied Engineering Research*, vol. 11, no. 23, pp. 11 223–11 227, 2016.

[151] A. K. Vincent and R. Nersisson, “Particle swarm optimization based pid controller tuning for level control of two tank system,” in *14th ICSET-2017 IOP Conference Series: Materials Science and Engineering*, 2017, p. 263.

[152] M. Lee and J. Shin, “Constrained optimal control of liquid level loop using a conventional proportional-integral controller,” *Chemical Engineering Communications*, vol. 196, no. 6, pp. 729–745, 2009.

[153] B. Şimşek, G. Ultav, V. A. Küçük, and Y. T. İç, “Pid control performance improvement for a liquid level system using parameter design,” *International Journal of Applied Mathematics, Electronics and Computers*, vol. 4, pp. 98–103, 2016.

[154] N. B. Almutairi and M. Zribi, “Sliding mode control of coupled tanks,” *Mechatronics*, vol. 16, no. 7, pp. 427–441, 2006.

[155] M. M. Noel and B. J. Pandian, “Control of a nonlinear liquid level system using a new artificial neural network based reinforcement learning approach,” *Applied Soft Computing*, vol. 23, pp. 444–451, 2014.

[156] Z. Aydogmus, “Implementation of a fuzzy-based level control using scada,” *Expert Systems with Applications*, vol. 36, no. 3, pp. 6593–6597, 2009.

[157] P. Aravind, S. Ramachandran, and M. Saranya, “Pso based pid controller design for a liquid flow process,” *International Journal of Current Engineering and Technology*, vol. 4, pp. 4252–4256p, 2014.

[158] T. Teng, J. Shieh, and C. Chen, “Genetic algorithms applied in online autotuning pid parameters of a liquid-level control system,” *Transactions of the Institute of Measurement and Control*, vol. 25, no. 5, pp. 433–450, 2003.

[159] P. Sethi and S. R. Sarangi, “Internet of things: architectures, protocols, and applications,” *Journal of Electrical and Computer Engineering*, vol. 2017, 2017.

[160] M. Wang, G. Zhang, C. Zhang, J. Zhang, and C. Li, “An iot-based appliance control system for smart homes,” in *2013 fourth international conference on intelligent control and information processing (ICICIP)*. IEEE, 2013, pp. 744–747.

[161] H. Xu, W. Yu, D. Griffith, and N. Golmie, “A survey on industrial internet of things: A cyber-physical systems perspective,” *IEEE Access*, vol. 6, pp. 78 238–78 259, 2018.

[162] A. Agrawal, S. C. Kor, U. Nandy, A. R. Choudhary, and V. R. Tripathi, “Real-time blast furnace hearth liquid level monitoring system,” *Ironmaking & Steelmaking*, vol. 43, no. 7, pp. 550–558, 2016.

[163] C. Anil and R. P. Sree, “Tuning of pid controllers for integrating systems using direct synthesis method,” *ISA transactions*, vol. 57, pp. 211–219, 2015.

[164] A. K. Jana, *Chemical process modelling and computer simulation*. PHI Learning Pvt. Ltd., 2018.

[165] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in engineering software*, vol. 69, pp. 46–61, 2014.

[166] H. Faris, I. Aljarah, M. A. Al-Betar, and S. Mirjalili, "Grey wolf optimizer: a review of recent variants and applications," *Neural computing and applications*, vol. 30, no. 2, pp. 413–435, 2018.

[167] N. Mittal, U. Singh, and B. S. Sohi, "Modified grey wolf optimizer for global engineering optimization," *Applied Computational Intelligence and Soft Computing*, vol. 2016, 2016.

[168] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: a nature-inspired algorithm for global optimization," *Neural Computing and Applications*, vol. 27, no. 2, pp. 495–513, 2016.

[169] A. Maier, A. Sharp, and Y. Vagapov, "Comparative analysis and practical implementation of the esp32 microcontroller module for the internet of things," in *2017 Internet Technologies and Applications (ITA)*. IEEE, 2017, pp. 143–148.

[170] G. Wang, Q. Shang, S.-e. Yu, and S. Yang, "Design of pc communication with hart field instrumentation," in *2012 International Conference on Computer Science and Electronics Engineering*, vol. 3. IEEE, 2012, pp. 299–302.

[171] P. Serikul, N. Nakpong, and N. Nakjuatong, "Smart farm monitoring via the blynk iot platform: case study: humidity monitoring and data recording," in *2018 16th International Conference on ICT and Knowledge Engineering (ICT&KE)*. IEEE, 2018, pp. 1–6.

List of Publications

1. Jailsingh Bhookya, J. Ravi Kumar (2021) “Sine-Cosine-Algorithm Based Fractional Order PID Controller Tuning for TITO Systems", *Int. Journal of Bio-Inspired and Computations*, Vol. 17(2), 113-120. (**SCIE** - InderScience)

2. Jailsingh Bhookya, J. Ravi Kumar, (2020) “ImprovedJaya Algorithm based FOPID/PID Controller for AVR System", *COMPEL - The international journal for computation and mathematics in electrical and electronic engineering*, Vol.4, No.39. (**SCIE** - Emerald)

3. Jailsingh Bhookya, J. Ravi Kumar, (2020) “PID Controller Design for Decentralized TITO Process Using Modified Differential Evolution Algorithm", *Int. Journal of Robotics and Automation*. (**SCIE** - Acta Press - In Production)

4. Bhookya, J., Jatoth, R. K. (2019) “Optimal FOPID/PID controller parameters tuning for the AVR system based on sine cosine algorithm", *Evolutionary Intelligence*, 12(4), 725-733. (**Scopus/ESCI** - Springer)

5. Jailsingh Bhookya, J. Ravi Kumar, (2019) “Fractional Order PID Controller Design for Multivariable Systems using TLBO", *Journal of Chemical Product and Process Modeling*, Vol.15(2). (**Scopus** - De Gruyter - ahead of print)

6. Jailsingh Bhookya, J. Ravi Kumar, (2018) “Performance Comparison of Optimal PID Controller Design for TITO System", *Journal of Engineering and Applied Sciences*, 13 (Special issue 3), 3330-3336. (**Scopus** - MedWell)

7. Jailsingh Bhookya, J. Ravi Kumar, “Implementation of PID Controller for Liquid Level System using mGWO algorithm and IoT application", *Journal of Industrial Information Integration*. (**SCIE** - Elsevier - Revising)

8. Jailsingh Bhookya, J. Ravi Kumar, “A novel Hybrid MFO-WOA Algorithm for Global Optimization and FOPID/PID Controller Design with Experimental Validation in IoT Applications", *IEEE Transactions on Emerging Topics in Computational Intelligence*. (**SCI** - IEEE - Under Review)