DEPARTMENT OF ELECTRICAL ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

CERTIFICATE

I, Ruchi Bansal, Roll No. 2K12/C&I/012 student of M. Tech. (Control & Instrumentation), hereby declare that the dissertation/project titled "Implementation of Multi-objective Genetic and Simulated Annealing Algorithm on Process control systems" under the supervision of Dr. Bharat Bhushan of Electrical Engineering Department, Delhi Technological University in partial fulfillment of the requirement for the award of the degree of Master of Technology has not been submitted elsewhere for the award of any Degree.

Place: Delhi **(RUCHI BANSAL)**

Date: 28/07/2014

DR. BHARAT BHUSHAN
SUPERVISOR
Associate Professor
Electrical Engineering Department
Delhi Technological University
Delhi-110042

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to Dr. Bharat Bhushan for his guidance and assistance in the thesis. Without his consistent support, encouragement and valuable inputs, this project would not have become possible.

I would like to express my deep gratitude to Prof. Madhusudan Singh, the H.O.D of Electical Engineering Department without his moral support my project won’t be reached to this level.

I would also like to thank my batch-mates and friends who encouraged and helped me in completing the thesis work. Finally, I express my deep sincere thanks to my Parents and my husband Mr. Vaibhav Porwal who motivated and encouraged me for higher studies, without which it wouldn’t have been possible.

Ruchi Bansal

(2K12/C&I/012)

M.Tech (C&I)

Delhi Technological University, Delhi

ABSTRACT

In spite of new advanced control techniques have been developed, Proportional-Integral-Derivative (PID) controllers are still popular in industrial control systems. These controllers are extensively used in industries due to their simple structure, ease of implementation, efficiency and cost effectiveness. But tuning of parameters of these controllers is an important matter of concern taken into consideration.

There are various methods to tune these three parameters. The conventional methods used are Ziegler-Nichols method, Cohen-coon method etc. These conventional tuning methods fail to achieve required satisfactory performance where plants have time delay or non-linearity. Here, an effective method to tune PID parameters is presented by using Multi-objective Genetic Algorithm (MOGA) and Multi-objective Simulated Annealing Algorithm (MOSA). These intelligent algorithms are search algorithms which are effective technique for computing complex optimization problems.

Here, the performance of Multi-objective Genetic Algorithm and Multi-objective Simulated Annealing Algorithm is compared with single-objective Genetic Algorithm, Simulated annealing Algorithm and also with the conventional tuning methods. Performance analysis for indexes like ISE, IAE and IATE is also observed. Here, the aim is to show the flexibility in PID tuning using MOGA and MOSA. By doing so, the engineer focus on more than one objective unlike single-objective based objective functions. Simulation results on different systems are considered to show the effectiveness of the proposed method.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER-1

INTRODUCTION

The purpose of this chapter is to present the motivation behind the work done in the thesis. This chapter also provides the aim of Multi-objective optimization as well as the thesis organization.

1.1 MOTIVATION TO OPTIMIZATION

Now a day, optimization in simulation has become an essential issue so, researchers are constantly working on it. Most of the attempt is to apply optimization in simulation software. Here, the application of metaheuristics is immeasurably considered to solve optimization problem.

The more significance was given to how metaheuristics developed for combinatorial problems used for optimization. Some advanced implementations were given to simulation problems. In particular, the main emphasis is on how to check for simulation noise in metaheuristics so that their performance in real applications can be improved. Also the emphasis was given to convergence statements that definitely have some value, and that it is reassuring to know that if efforts would be involved, the search will finally be successful. Furthermore, it would be better to terminate the search and know at that particular moment how good a solution has been obtained. With regards to metaheuristics, little work has been done in the deterministic combinatorial optimization context to analyze convergence and even very little work in the simulation optimization problems.

Optimization has constantly been one of the most important issues in the field of engineering. Here, optimization of the solutions to any given problem is done. For a problem, all the possible solutions can be found out in a definite time if the no. of solutions is fixed. Then the performance can be compared to find the solution i.e optimal solution. But actually, the number of solutions to any problem is so large that any mechanism would not be able to generate all the solutions in the definite duration. Hence, researchers thought of optimizing the solutions in a finite duration. The problem of optimization has more than one solution with different accuracy. Here, the optimal solution is chosen as the best solution.

This value of the objective function is basically a measure of goodness of the solution. It shows how accurate is the solution to the problem. The output of the algorithm is some set of values which decides the optimal value to the problem. The different set of values can be obtained by varying these parameters. Fitness function depends on these variables and the most optimal values give the final output of the algorithm [43].

The goal of optimization is to seek improvement to approach some optimal point [16]. There are problems which cannot be defined like best scientist, cricket or poet. Here best means better than others. Most important goal for optimization is ‘improvement’.

1.2 INTRODUCTION TO METAHEURISTICS

There are many optimization methods to solve the complex problems but metaheuristics prove to be efficient when other methods of optimization prove to be unsuccessful or inefficient. These metaheuristic methods can be specifically applied to many practical problems of combinatorial nature [24]. The strength of metaheuristics lie in its efficacy. It does not ensure to provide the best solution but the optimal solution.

With the emergence of metaheuristics like Tabu search, Simulated annealing and Genetic algorithms, the main challenge has become to adapt the metaheuristics to solve particular class of problems [55]. Furthermore**,** good metaheuristic implementations to the problems provide optimal solution in very less time. The applicability of metaheuristics to solve complex optimization problems has an advantage over other local optimization methods. In this approach, the search for the optimal solution is improved as the algorithm proceeds in which search is guided by some pre-known principles [36]. Through this, metaheuristics can be viewed by specifying some of the elements. This can be considered as the advantage or the disadvantage to the method.

1.3 METAHEURISTIC APPROACH TO ALGORITHMS

The use of metaheuristics can be applied to the complex optimization problems. Researchers have given this approach a continuous attention in the past decades. The classical optimization techniques work out the specific problem immediately rather Metaheuristic algorithms acts as a general purpose method to investigate the search space. Metaheuristic algorithms are also known as “black-box” algorithms as they use inadequate knowledge about the particular problem to be tackled. These algorithms take inspiration from the concepts which are far from the field of optimization like Particle Swarm Optimization (PSO), Simulated Annealing (SA), Ant Colony Optimization (ACO) or Genetic Algorithm (GA) [40].

The metaheuristic approaches to the combinatorial problem together with the concepts and techniques from various methods are designed to develop new and highly efficient algorithms. Some basic concepts allow us to focus on the various characteristics of different methods used in hybrid algorithms. ACO, PSO and SA are all stochastic algorithms but PSO and ACO are population-based method while SA is a trajectory based method. In population-based method, a set of different solutions are explored at each iteration whereas in trajectory based method, a new single current solution at each iteration is determined.

1.4 THE AIM OF MULTI-OBJECTIVE OPTIMIZATION

Many problems are Optimisation problems, those whereby the configuration of a system must be determined which will maximize or minimize the performance of that system. In an optimisation system with a single performance metric (objective), the aim is to locate the configuration which is superior to all other configurations. For real-world problems, it is usually the case that there is more than a single objective of the system which it is desirable to optimize. Such systems are said to represent Multi-objective optimisation problems. As these objectives are generally competing it is no longer possible to find the single solution which is superior to all others in every objective and the aim then becomes to find all the solutions for which there exists no superior solution. For problems with only a single objective, Simulated annealing and Genetic algorithm optimisation techniques which is particularly desirable as there exists a proof that it is able to converge upon the optimal solution, even for problems with features which present difficulties for other popular optimisation techniques such as those where the location of the global optima is ambiguous [17]. As such, the motivation for this work is to investigate the extension of intelligent algorithms to Multi-objective Optimisation.

Optimisation problems often involve the simultaneous optimisation of competing objectives. One simple and spontaneous example of a class of multi-objective problem is that of mechanical or electronic system design. As in the case of minimization of the cost of manufacture of the system, while maximising its performance. Generally these are in competition (although many other objectives will often exist in such problems, such as maximising the efficiency and the reliability of such systems). In such an optimisation there will be variables which describe the construction of the system, called the parameters of the optimisation problem, such as the combination and layout of components in a system design problem, and a configuration of these parameters is referred to as a solution. The performance of a solution to the problem can be evaluated for each objective to be optimised, which provides a series of values describing its performance (two values in the two-objective cost/performance example).

While it is clear that some solutions are completely better than others, there exist many possible solutions which, when compared to each other, are not entirely better or worse. In these cases some objectives may be better while others are worse, leading to solutions which are not clearly ordered. Where one solution is completely better than another, the better solution is said to dominate the poorer and a dominating solution must be no worse for any objective and must show an improvement in at least one objective. As optimisation in the multi-objective case is generally a compromise between competing solutions, it is usual for there to be a set of solutions, none of which dominate each other (it is said that they are mutually non-dominating) and for which no feasible configurations dominate them. This set is called the Pareto set, and represents the optimal trade-offs between objectives available. The aim of Multi-objective Optimisation is therefore, given a definition of possible parameterisations and functions to evaluate the performance in each objective, to find the set of all parameter configurations which correspond to the members of the Pareto front, those configurations for which no configuration is possible for which the objective functions signify completely better quality.

1.5 OBJECTIVE OF MY PROJECT WORK

As the parameters of PID controller interrelate with each other so proper tuning of these parameters is a major issue of concern. The purpose of this project work is to design an optimal PID controller design using Intelligent Algorithms like Genetic Algorithm (GA) and Simulated Annealing (SA) Algorithm. Here, Multi-objective based PID tuning algorithms are proposed by GA and SA techniques to improve the performance of the PID controller. These computing techniques for a PID controller considerably reduce the overshoot and optimize settling time or rise time as compared to conventional tuning algorithm like Ziegler-Nichols tuning method.

In this project, time domain criteria are used for evaluating the PID controller. A set of good control parameters (proportional, integral and derivative gain) can provide a good step response that will result in performance criteria optimization in the time domain. These performance criteria in the time domain include the overshoot, rise time and settling time.

1.6 ORGANIZATION OF THESIS

CHAPTER-2 This chapter consists of the literature survey on various metaheuristic and evolutionary algorithms. The main focus is given on the papers related to Genetic algorithm and Simulated annealing algorithm.

CHAPTER-3 This chapter deals with the description of the concept of Genetic algorithm along with the operations used in the working of this algorithm.

CHAPTER-4 This chapter describes the concept of Simulated annealing algorithm along with its working phenomenon.

CHAPTER-5 This chapter consists of description of PID controllers, test problems, fitness functions along with its tuning with Genetic algorithm and Simulated annealing algorithm.

CHAPTER-6 This chapter presents results and discussion.

CHAPTER-7 This chapter brings out the main conclusion of this research work.

CHAPTER-2

LITERATURE REVIEW

2.1 INTRODUCTION

This chapter consists of a literature survey on different algorithms in this project. Various books and research papers related to PID tuning and intelligent algorithms have been studied, which form the backbone of the present work undertaken.

2.2 BRIEF OVERVIEW OF PAPERS

A conventional technique was proposed by Cohen–Coon [2] based on the concept of reaction curve of the process. Kitamori [3] also presented a different technique to tune the PID controller. But these methods could not satisfy the requirements of the designer. As a result another tuning technique was proposed by Tyreus and Luyben (T-L) [8] for more sophisticated plants.

Ziegler-Nichols (Z-N) [11] proposed a method a PID tuning which was related to quarter decay ratio but this technique was unsatisfactory for the processes where the overshoot is not desirable. Among the popular and well-known approaches are the integral of squared time weighted error rule (ISE) [11], integral of absolute error rule (IAE) [13], internal-model-control (IMC) based method [5], gain-phase margin method [14]. W.K. Ho computed the relation between gain and phase margins to set the parameters of PID controller. Also, the approach to reduce the model was developed by Wang to design PID controllers [19].

One more method of PID tuning was proposed by Z. Y. Zhao which was related to fuzzy gain technique [12]. B.Nagaraj et al. [15], soft computing techniques are applied to tune the PID controller. The results obtained from the techniques are compared with the traditional or conventional methods. The results obtained from the soft-computing techniques show improved performance. Hence, these techniques proved to be efficient and stable.

Also a neural networks tuning based PID controller with the help of fuzzy parameters is proposed in [20]. PID controllers are quite popular in industrial applications so development of most of the fuzzy controllers rotated around fuzzy PID controllers in the past years [18]. But designing of systematic fuzzy controller is not easy as the overall control structure of fuzzy is difficult to design. To design the effective fuzzy controller, the tuning of gains is also an important issue taken into consideration.

B. Jeffrey Schamburg et al. [23] presented a methodology for superheterodyne surveillance receivers. The optimization of the parameters is done using the Simulated annealing algorithm.

Chunna Zhao et al. [25], presented a comparative study of fractional order plants with the integer order PID controller. Fractional order controller provide more adequate description as they require more number of tuning parameters so these systems are more complex. Better performance of fractional order PID controller is shown.

Ashab Mirza et al. [29] presented PID controllers against bounded uncertainties. A model based on feedback linearization is used. The two loops (inner and outer) are used to linearize the non-linearities of the process. The robustness of the controller is shown through the results.

L. Galotto Junior et al. [30] presented a method called recursive least square is presented for tuning of PID controllers. The Genetic algorithm based tool was developed using Programmable Logic Controllers (PLC). Here, the identification of the plant is done based on the plant model and the parameters of the PID controller were found out. The method is used for plant identification and algorithm is used to tune the PID parameters. The communication is established with PLC and the data is acquired according to which simulation is done to evaluate the overall performance.

M. Willjuice Iruthayarajan et al. [31] proposed the designing of PID controller using Real coded Genetic algorithm (GA) and particle swarm optimization

(PSO). The performance measures like ISE, IAE and ITAE were minimized for two linear systems. The performance of RGA was compared with PSO on the basis of computation time, time-response specifications and statistical performance.

M. Shady Gadoue et al. [32] presented a comparative study of PI controller and fuzzy sliding mode controller. Here, GA is used to optimize the gains of PI controller. The above two controllers are compared on the basis of reference speed and load torque changes. The results show that the controller with fuzzy sliding strategy is better against the parametric variations.

Urvinder Singh et al. [35] presented a method based on Simulated Annealing algorithm to optimize the parameters of Yagi-Uda Antenna. The gain of the antenna is optimized which is a non-linear function of the designed variables. Simulation results are shown to evaluate the performance of the system.

Jin-Sung-Kim et al. [37] proposed the improved GA with difference in defining random number generation, objective function or fitness function. This newly developed Genetic algorithm is applied to tune PID parameters and the reverse osmosis plant is controlled. A new Root mean square error (RMSE) type function is designed which improves the performance of the algorithm.

B. Khaki et al. [39] presented an approach to optimize the parameters of the controller using Genetic and Simulated annealing algorithm. The model of static var compensator is tested and the objective function is designed for the controller on the basis of eigen values under the effect of disturbances.

Liu Fan et al. [41] presented an auto-tuning method to design the PID controller. The range of parameters like proportional, integral and derivative gain has been calculated using Z-N formula. Then by implementing GA to tune the PID parameters, the simulation is done on variety of plants.

Balram Sumanaet et al. [45], orthogonal simulated annealing (OSA) algorithm is proposed which is based on multi-objective optimization algorithm. This algorithm has incorporated a design called an orthogonal experiment design (OED) which focussed on multi-objective approach. This OSA algorithm performed outstandingly in comparison to other algorithms.

S. M. Giriraj Kumar et al. [47] discussed the comparative study of Simulated annealing (SA) based tuning of PID controller and conventional methods of PID tuning. The results show that SA based tuning is better for the complex processes used in industries. Transient characteristics as well as frequency domain characteristics are improved using SA algorithm. Also, the disturbance rejection is being improved with the better stability of the process.

Jun Liu et al. [49] presented the effectiveness of simulated annealing algorithm to optimize the design of transmission of the engine cooling fan. The efficiency of this algorithm is compared with the conventional based techniques.

M.F Nor Shah et al. [51], presented a technique called meta modelling approach to tune the parameters of the PID controller. The radial basis function neural network meta model is used to show the results for a single input and single output (SISO) evaporator system. The results obtained from genetic algorithm (GA) and ant colony optimization (ACO) for this non-linear plant.

R. Bindu et al. [56], presented the study of positional control of DC servomotors. In this paper, metaheuristic technique like Genetic algorithm is used to evaluate the optimal values of Proportional-integral-derivative (PID) controller parameters according to the particular system requirement.

Claudio Fabiano Motta Toledo et al. [57] presented a multi-population genetic algorithm (MPGA) to tune the PID controller. This algorithm is compared with the standard genetic algorithm. Neural networks are trained to provide the control criteria. This approach proves to be better when compared with the gradient techniques.

Pranay Lahoty et al. [63], presented a method for optimizing the gains of PID controller of an unstable system. The optimization algorithm like Genetic algorithm and Particle swarm optimization algorithm is used to achieve a quick and stable system. Performance analysis is done by evaluating the time domain specifications and objective function. The simulation results obtained using these two mentioned algorithms are also analyzed against the traditional Z-N tuning technique.

Mr.Jyotiprakash Patra1 et al. [64], the gain of the controller along with integral time and derivative time is calculated. Equations for the parameters of the controller are given using crave fitting technique. The ITAE performance index is minimized and accordingly the fitness function is designed. The results are tested on a first order plant with time delay. The results obtained from this technique are compared with the conventional or traditional methods. The results are also tested on the hardware implemented.

.

CHAPTER-3

GENETIC ALGORITHM

This chapter presents an overview of Genetic algorithm. Detailed information regarding its terminologies with its working is also provided by this text.

3.1 BIOLOGICAL INSPIRATION

Evolutionary algorithms are inspired from the biological analogues of the natural species. The natural evolution system provides inspiration to the artificial systems. A strong relation acts between the natural and artificial evolution systems. Likewise, Genetic algorithm is inspired from the reproductive systems of the natural species. The different individuals interact sexually and produce children to form new population. The new population or generation formed is far better than the previous generations as the newer generation adapt to the changing environment more aptly.

Population is basically a set of individuals comprised together. An individual represents a single entity in the population [41]. The chromosomes and DNA are responsible for the behaviour of individual’s traits or characteristics. Basically, species are made up of cells and the chromosomes from the structure of these cells. These chromosomes are splitted into genes and these genes are combined together in different ways to form different sort of features or characteristics of an individual. The capability of individuals is referred as the fitness of the individual.

According to the concept of Charles Darwin, the characteristics of the fittest individual pass on to the next generation and the characteristics of weaker individual ultimately die out. This implies the theory based on “survival of the fittest”.

3.2 INTRODUCTION TO GENETIC ALGORITHM

Genetic Algorithm also known as GA is a search algorithm which is similar to stochastic and adaptive optimization techniques. It imitates the process of evolution based on the concept of survival of the fittest or Darwinian Theory. It was first suggested by Sir John Holland with his colleagues and students in the year 1975. Here, gradual development acts as an optimizing process which leads to the progressive improvement of the objective or fitness function [32].

GA has been an effective technique for computing complex optimization problems. It works well for Multimodal and exhaustive search problems. Here, the goal is to search for the best possible values of the parameters which give the optimal value of the objective function. It does not trap into the local minima and finally converges to global minima [33]. GA use probabilistic transition rules and search for a population of points, not a single point. Their main characteristics are as follows:

(1) GA work with a coding of the parameters and not with the parameters themselves.
(2) GA use probabilistic and not deterministic transition rules to change the initial population through consecutive generations, which helps in the search process.

3.3 PRINCIPLES OF GENETIC ALGORITHM

Genetic Algorithm imitates the process of Evolution. Here, gradual development acts as an optimizing process. It is like an insight on continued existence and adaptation. GA begins with a set of solution called population and each individual represents a set of solution to the problem. An appropriate objective function or fitness function is used to determine the performance of each individual.

Selection/Reproduction, Crossover and Mutation are the three main phases of GA which leads to the progressive improvement of the objective function so that the successive generations would become better and better.

3.4 GENETIC OPERATORS

A new population is produced from the existing population by applying the three main Genetic operators. Reproduction or selection, crossover and mutation are generally used to improve the next generation. These three phases play an important role for the performance of the Genetic algorithm. The three operators are described in brief as follows:

3.4.1 Reproduction/Selection

The process involves the choosing of the two parents from the set of individuals for the operation of crossover. From the current population, the individuals are chosen to pass on further for the formation of next population [37]. An appropriate method is selected to select the parents so that the next generation is better than the previous generation [52]. Different selection methods are used for different problems. The principal behind all the selection methods is to transmit or select the fitter chromosomes further. The normally used methods for selection are explained in brief as follows:

1. Roulette Wheel selection
2. Stochastic Universal sampling
3. Rank selection
4. Tournament selection

3.4.1.1 Methods of selection

i) Roulette Wheel Selection

It is one of the important methods of selection in which probability of selecting the individual is directly proportional to that of individual's fitness value. Due to this reason, the individuals having the high fitness value get a better chance of being selected while the individuals having the low fitness value are discarded. Hence, the fitter individuals involve more in this process of selection [22].

Here, the Roulette wheel is rotated and the analysis is done when the wheel stops. It is obvious that the wheel is having more probability to stop where the individuals' fitness value is higher.

ii) Stochastic Universal Sampling

This method of selection is basically applied to increase the diversity of the solution. As in the case of Roulette Wheel Selection, the individuals with higher fitness value get the chance again and again but the lower fitness individuals are neglected. This is not the case with Stochastic Universal Sampling (SUS).

iii) Rank Selection

It is one of the selection methods where the individuals having extraordinarily large fitness value is not selected because the individuals are scaled in proportion to their ranks [38]. Here, the process of selection is more on the basis of rank that is evenly distributed instead of fitness values.

iv) Tournament Selection

This process of selection also contributes in solving the problem of selecting the individuals with higher fitness value. In this method, the candidates are selected to compete with each other. This selection method is motivated by the competitive method. Mostly, the individuals with higher fitness value wins but sometimes the lower fitness individual can also win [34]. There are also some other methods of selection like uniform selection, random selection etc.

3.4.2 Crossover

This process is basically the combination of two parents to reproduce the better or fitter offsprings. As already the process of selection fills the population with good individuals so, they mate with the hope to reproduce better children. In the operation of crossover, the selected string or chromosome is splitted and combines with other selected string or chromosome at the same crossover site. The probability of crossover is increased as crossover is not much random in nature. There are different types of crossover operators like single point crossover, two point crossover, arithmetic crossover and many more [66].

3.4.2.1 Common types of crossover

i) Single-Point Crossover

In this type of crossover, a point is selected which is known as crossover point. The new individual is formed by taking the bits from both the parents. The bits to the left of this crossover point from the first parent are combined with the bits of the right side of the other parent. This process can be shown in Figure 3.1:

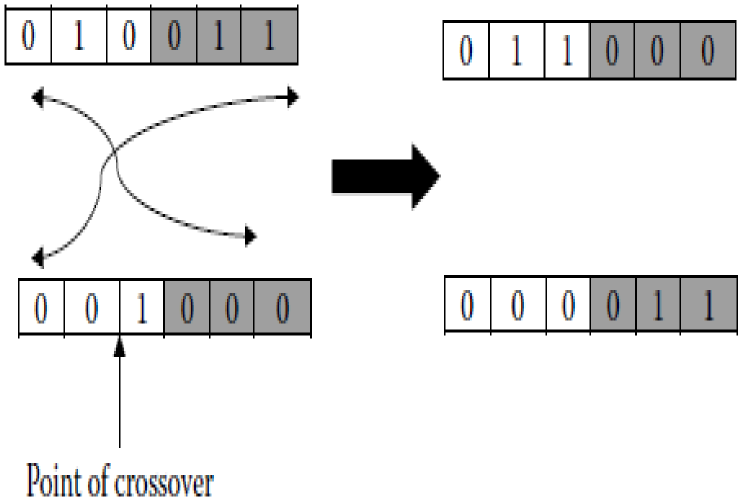

Fig 3.1: Single-point crossover

ii) Two-Point Crossover

This type of crossover is used when the former crossover does not produce good results. This two-point crossover is used successfully by taking two crossover points. By this method of crossover, more diversity is achieved. The process can be represented in Figure. 3.2:

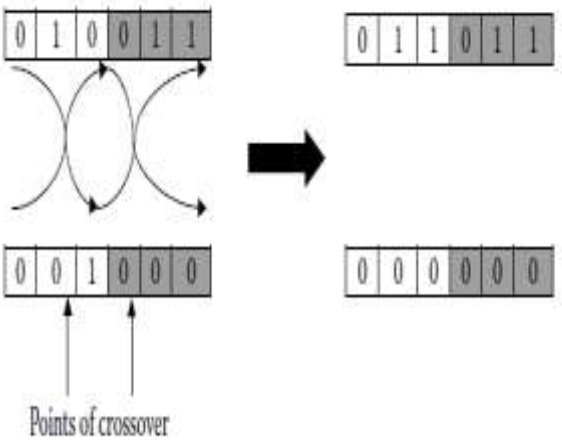

Fig 3.2: Two-Point Crossover

iii) Scattered Crossover

Unlike the two former methods discussed, scattered crossover is not a position oriented technique. In this type of crossover, random vector of binary values with 0 and 1 are created which corresponds to the parents. The bit 0 denotes one of the parents while the other bit denotes the other parent.

iv)Intermediate Crossover

In this type of crossover an intermediate value is taken from the two parents to form the new individual. This technique is generally applicable to the numeral representation of the string.

v) Heuristic Crossover

This process is comparable to the previous discussed method. Here, the position between the genes of the two parents helps in producing the new individual [42]. The newly formed gene is placed near to the fitter parent and far from the poorer quality parent.

3.4.3 Mutation

The operation of Mutation is applied after the process of crossover is completed. In the process of Mutation, bit 1 is flipped to 0 and bit 0 is flipped to 1 [62]. By doing so, the random characteristics are injected into the population [6]. This process is important to enhance the diversity of the population.

The good traits or characteristics remain and continue in the next generation while the bad trait finishes over time. The achieving of local minima can be avoided by the process of mutation. Maximum amount of mutation is 5% of total number of bits.

The process of mutation is shown in Figure 3.3:

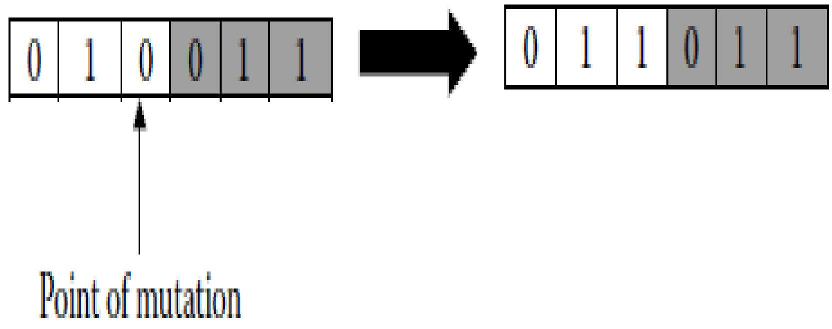

Fig 3.3: The mutation genetic operator

3.4.3.1 Types of Mutation

i)Uniform Mutation

This is commonly used operation in which the bit 0 is simply changed to 1 and bit 1 is changed to 0 [7].

ii) Gaussian Mutation

In this type of mutation, the solution is represented by numbers. Some randomly generated numbers are added to the present values at the chosen place.

iii) Variable Mutation Rate

As the name indicates the rate of mutation can be altered during the working or running time of the algorithm. This type of mutation helps in controlling the behaviour of the algorithm. As the mutation rate is the indication of randomness so it is kept high in the beginning so that it can explore new possibilities of points in the search space. But after sometime this rate is lowered so that the algorithm can be converged. Some special operators are also used for different specialized purposes. These operators add more flexibility and accuracy to the working of the GA.

3.4.4 Eliticism

Sometimes the high fitness individuals are ruined by the process discussed above that is mutation and crossover. To protect these individuals from being damaged, Eliticism plays an important role. In this method, the individual with highest fitness just pass on to the next generation.

If this individual is again at the highest fitness value in the next generation then again it continues to pass in the next generation otherwise it involves in the process of crossover and mutation [9]. In this generation whichever individual is having the highest fitness is called the elite member and passes to the successive generation.

3.4.5 Fitness Function

The value of the fitness function or objective function is minimized or maximized to optimize the solution of a specific problem. This value of the fitness function is basically an assess of goodness of the solution. It shows how accurate is the solution to the problem. The output of the algorithm is some set of values which decides the optimal value to the problem. The different set of values can be obtained by varying these parameters.

Fitness function depends on these variables and the most optimal values give the final output of the algorithm. The fitness function is very useful in examining how the algorithm is performing when the iterations are moved on. Three kinds of fitness are possible like best fitness, average fitness or the worst fitness.

3.4.6 Stopping Condition

The Genetic algorithm runs repeatedly for as many generations mentioned. The solution keeps on improving as the algorithm continues. However, conditions for stopping the algorithm need to be decided to finish the program. This is done by the stopping criterion, which decides when the steps of the algorithm will be ended. The best solution at that particular point would be considered as the final solution.

Stopping criterion can be set in several ways:

- The running of the algorithm can be stopped if the solutions obtained are adequately optimized. If there is no significant change in the further iterations then no more optimization is required hence the algorithm can be stopped.
- The algorithm can also be stopped if the algorithm meets the set time taken or the number of fixed generations.
- Stall time is the time when the algorithm is converged and there is much improvement in the consequent generations.

3.4.7 Fitness Scaling

In the process of scaling, the individuals are rescaled according to some principles so that few high fitness individuals do not supervise the newly formed generation. Here, fitness values plays a significant role as the scaling is done on the basis of fitness value guided by the respective principles. Scaling can be of different types like Rank scaling, Proportion scaling, Top scaling etc.

3.5 HOW GA WORKS?

The algorithm starts by creating initial random population. Then the algorithm produces a string of new populations. In each generation, the algorithm uses the individuals in the current generation to create the next generation population.

To create new population, the subsequent steps are performed by the algorithm:

- The fitness value is evaluated by scoring all the members of the current population
- Scaling of these fitness scores are done in order to convert them into more useful range of values.
- Then the individuals according to their fitness values are selected.
- Then the elite members are selected which directly move on to the next generation

- With the help of processes called crossover and mutation, the children are produced from the parents. Sometimes the children are produced by single-parent mutation.
- Now the children of the current population becomes the parents of the next generation [48]
- the algorithm stops when any one of the stopping criteria is met

The flowchart of MOGA-PID controller is shown in the Figure 3.4:

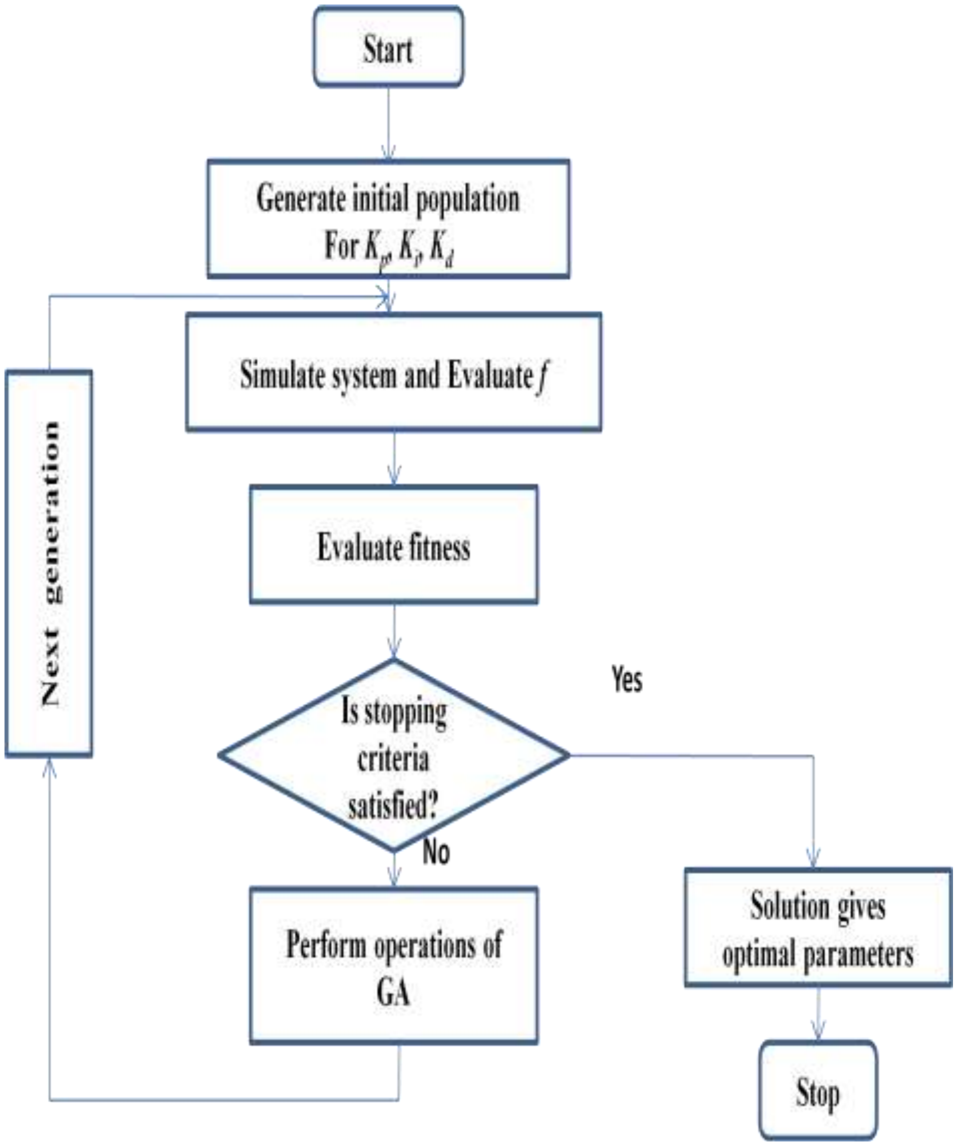

Fig 3.4: Flowchart of MOGA-PID controller

3.6 ADVANTAGES OF USING GA

The competence of GA does not rely on the features of the plant under consideration, due to their robustness. So, Genetic algorithm is applicable to a huge variety of plants use in industries and control engineering including non-linear ones [10]. Also, GA can consider various objectives together like overshoot, rise time, settling time etc. rather than restricted to single objective. GA does not converge over local optima like conventional technique hence possibly give effective results to the real-world problems.

3.7 DISADVANTAGES OF USING GA

Sometimes GA takes a long time to produce the results. Also, the operators and their selection method need to be chosen carefully otherwise the results can be worsening at each iteration or in other words, the successive generation can be worse than the previous generation.

CHAPTER-4

SIMULATED ANNEALING ALGORITHM

This chapter describes an overview of simulated annealing algorithm along with its concept and process of working.

4.1 INTRODUCTION TO SIMULATED ANNEALING

Simulated annealing is a technique based on the natural phenomenon of annealing in solids. This process involves achieving low-energy state when the metal is gradually cooled to adopt a crystalline state. When the metal is heated to a high temperature, its structure is changed as the particles inside the metal travel freely in the region. But, as the temperature is slowly lowered, the particles limit their motion and try to attain the ground state with lower energy.

This inspiration from the annealing process acts as a motivation for the Simulated annealing algorithm to use it in the simulation process. Hence, this process can be used as a search technique for optimization and attaining the minimum (optimum) solution to the problems [35].

The Simulated Annealing is based on the following characteristics annealing in metals that is, first heat the solid state metal to a high temperature and then slowly cool it down according to a particular schedule. This algorithm was first given by Metropolis in 1953 and then explained by Scott Kirkpatrick C Daniel Gelatt and Mario P.Vecchi in 1983 to deal with highly nonlinear problems.

Basically, Annealing is a thermal process for obtaining low energy states of a solid in a heat bath [28]. The process contains two steps:

1) Initially the temperature is increased to the maximum value at which the solid melts [23].

2) Then, the temperature is decreased slowly and carefully so that the particles arrange themselves in the ground state of the solid [53]. Basically, Ground state is a minimum energy state of the solid. The ground state of the solid is obtained only if the maximum temperature is high enough and the cooling is done slowly.

4.2 OVERVIEW OF SIMULATED ANNEALING ALGORITHM

Simulated annealing algorithm is inspired from the annealing process in metals that is, when the solid state metal is heated to a high temperature until it attains random state and then cools down slowly to attain thermal equilibrium so that the atoms arrange themselves in the minimum energy state of a perfect crystal. This minimum energy state is also called as ground state. To achieve this ground state the metal is heated enough carefully and being cooled down slowly. Here the average potential energy per atom is lowered and this can be used as minimization process. This process is carried out to eliminate imperfections from the crystal.

The slow cooling of the material and accepting worse solutions explores the larger search space and makes it more extensive search for an optimal solution [63]. The extent of the search is determined by a probability distribution and the algorithm accepts a worse point based on an acceptance probability [49]. This probability directly depends on the temperature and sometimes helps to identify a new search region for a better minimum. This algorithm can be used for stochastic/random, chaotic and non-linear optimization problems. Basically, this algorithm is based on Boltzmann probability distribution formula which says,

$$P(E) = e^{(-\Delta E/KT)} \tag{4.1}$$

Where E is the energy, T is the temperature and K is the Boltzmann constant.

Here, by controlling the parameter ‘T’, the convergence of the algorithm can be controlled [55]. Under Markov chain Monte Carlo Method also known as MC-MC Method, Metropolis-Hastings (MH) algorithm is the most powerful tool to solve variety of engineering problems. It is basically a sampling algorithm, used to generate new samples. These samples should be accepted or rejected depends on some rules or conditions which are based on Boltzmann distribution and this is Metropolis-Hastings (MH) algorithm. The selection of new states of SA is shown in the following figure 4.1:

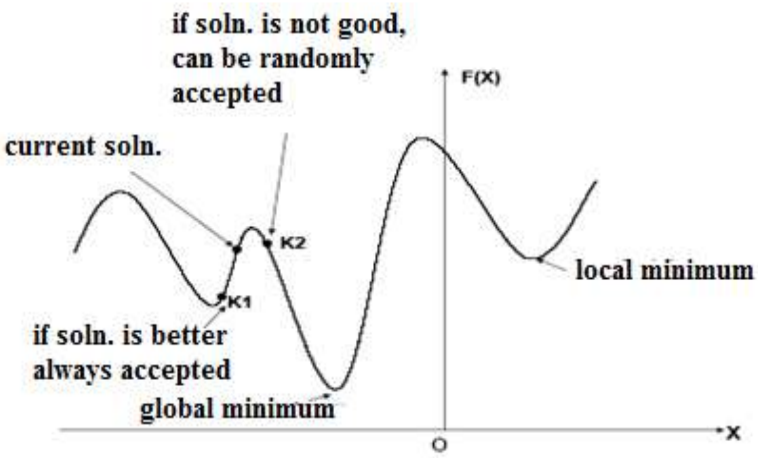

Fig 4.1: selection in new states of SA

In the beginning, the probability of accepting a worse state is high and then the probability decreases as the temperature reduces. For each value of temperature, the system must attain an equilibrium i.e., a number of new states must be tried before the temperature is reduced usually by 10% [54].

The Simulated annealing algorithm maintains both a state and a computational temperature, which is initially high and is reduced to (near) zero during the algorithm's execution. The configuration is usually a solution to the optimisation, and at each iteration of the algorithm this solution is perturbed in some manner to produce a new solution. The quality of a solution is said to be the energy of the state, analogous with real-world annealing. The quality of both solutions is evaluated, using the objective function, and a new state is selected from the two solutions. When the new solution is no worse than the previous solution, the new solution is selected as the state. Where the new solution is of lower quality than the existing solution, it may be accepted with a probability dependent upon both the current computational temperature and the magnitude of the difference in quality. Solutions are most likely to be accepted if they are only slightly worse, or if the temperature is high. Largely inferior solutions are unlikely to be accepted, and as the temperature approaches zero, it becomes vanishingly unlikely that any inferior solution will be accepted

The flowchart of SA Algorithm is shown in the Figure 4.2

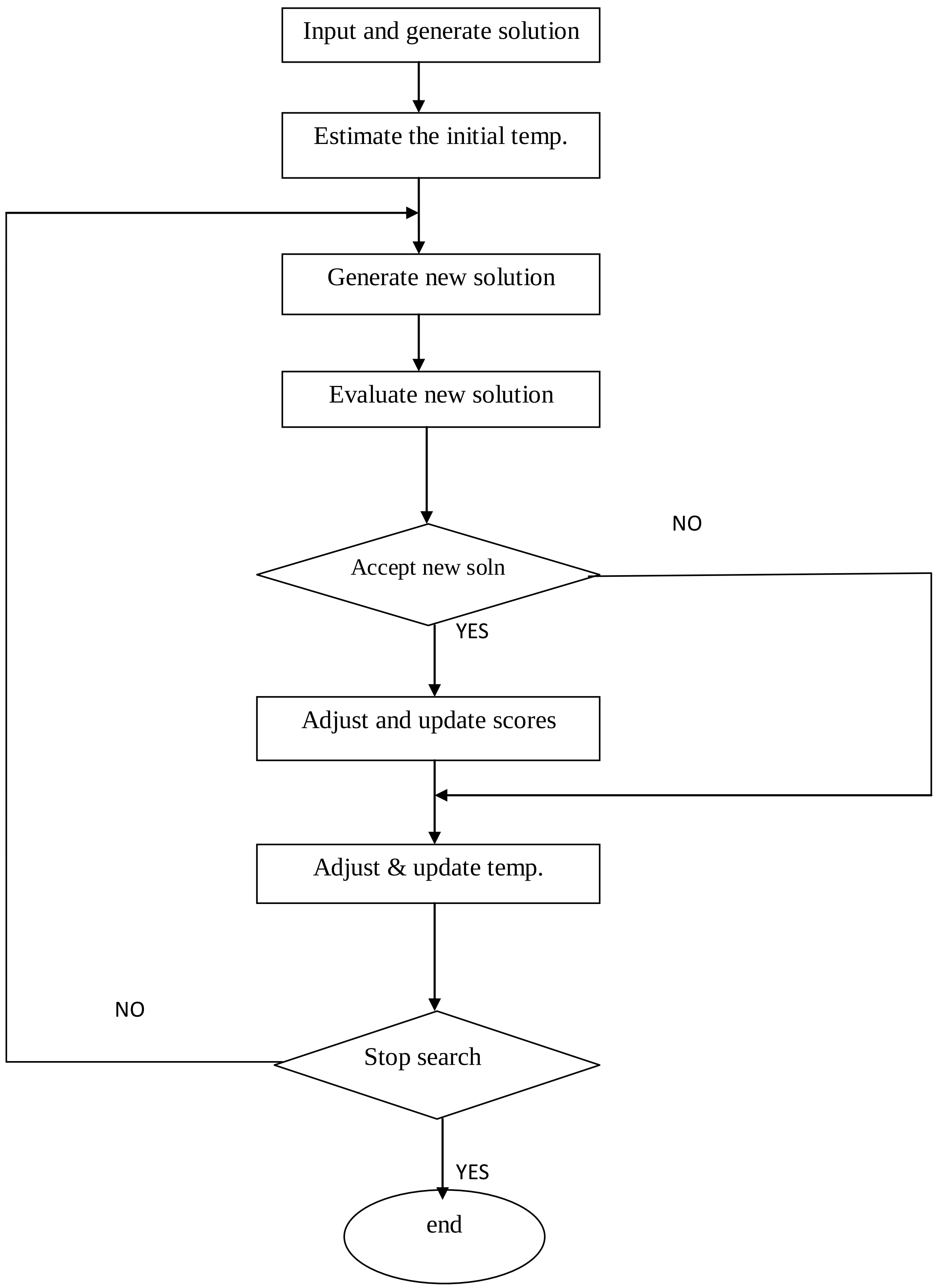

Fig 4.2: Flowchart of SA algorithm

4.3 HOW SA WORKS?

Simulated annealing algorithms are generally run on schedules shorter than those for which convergence on the global minimum is guaranteed and, as such, it is usual to maintain a record of the best performing solution found during an optimisation, not only the final solution. While the cooling schedule required to guarantee convergence is infeasibly slow, simulated annealing generally works well even on much shorter schedules and as such, it is worth extending simulated annealing for multi-objective optimisation.

Basically, simulated annealing algorithm uses the temperature parameter to manage or organize the search. Initially the temperature is kept at a high value and gradually it lowers with each iteration. A new point is generated in each iteration and its distance from the current point is calculated which is proportional to the temp, [39].

The current point is replaced and move to the next iteration if the new point has a better function value. It can also accept a worse point and move forward but its probability is directly dependent on the temperature. Sometimes this action helps to find the better minima. The outline of the algorithm is as shown in the Figure 4.3 [26]:

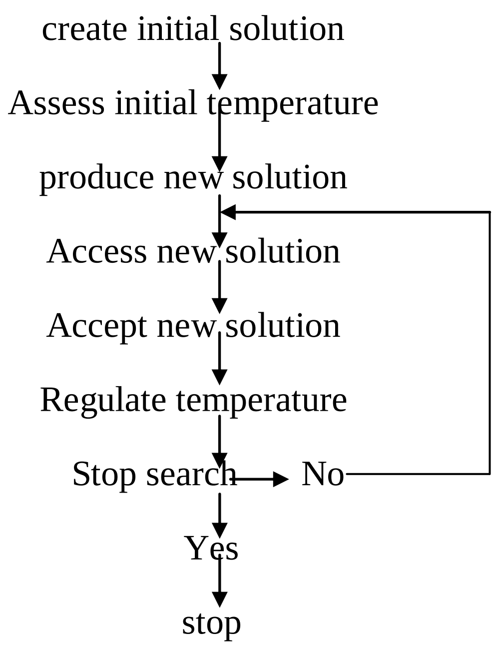

Fig. 4.3: Outline of SA algorithm

4.4 STRENGTHS OF SA

- Simulated annealing is a robust technique that can be dealt with highly nonlinear models, noisy data and chaotic with many constraints.
- Its main advantages over other local search methods are its flexibility and its capability to move towards global optimality.
- The algorithm is quite adaptable since it does not rely on any restrictive properties of the model.
- The performance of the algorithm can be enhanced by tuning it to any stochastic and non-linear system and the capability to tune a given algorithm for variety of problems should be considered an important attribute of an algorithm.

4.5 WEAKNESSES OF SA

- Simulated annealing is a metaheuristic algorithm so a number of options are needed to turn it into actual process of the algorithm.
- The compromise is done between the quality of the solutions and the time required to calculate them.
- To work out the algorithm, the precision of the numbers plays an important role in the quality of the results.

CHAPTER-5

PID CONTROLLERS AND ITS TUNING

This chapter introduces the basic PID controllers along with its tuning algorithm. The main contents of this chapter are Test problems on which the algorithm works and tuning of PID by Multi-objective Genetic and Simulated annealing algorithm.

5.1 INTRODUCTION TO PROPORTIONAL-INTEGRAL-DERIVATIVE (PID) CONTROLLERS

In spite of new advanced control techniques have been developed, Proportional-Integral-Derivative (PID) controllers are still popular in industrial control systems [29]. These controllers are extensively used in industries due to their simple structure, ease of implementation, efficiency and cost effectiveness.

When only proportional controllers are used in case of over damped systems (where $\zeta>1$) response of the closed loop system is very sluggish. Therefore, large values of k_c are preferred to make $\zeta<1$. Then the closed-loop response reacts faster but it becomes oscillatory. Also, if gain is large the sensitivity of controller's actuating signals towards deviations will be higher. Although, it cannot keep the process response at the desired set point and introduces offset, the response is much closer to the desired set point than would have been with no control at all. But, as the gain k_c increases, the offset decreases. The increase in the speed of systems response and the decrease in the offset, both very desirable features come at the expense of higher overshoots and longer oscillatory response. The response moves from sluggish over damped to faster but oscillatory under damped behaviour. Overshoot and decay ratio of the closed loop response both increase [4].

The Integral action causes the controller output to change as long as an error exists in the process output. Therefore, such a controller can eliminate even small errors. Increasing the integral action makes the response of the closed loop systems more sensitive. It increases the order of the dynamics for the closed loop response. Also, by increasing the order of a system the response becomes sluggish [21]. Thus, Integral action alone is expected to make the response of the closed loop system more sluggish.

The Derivative control does not change the order of the response. Response of the controlled process is slower and as the effective time constant increases, damping increases. This characteristic produces more robust behaviour by the controlled process. Thus, the overall effect of PID control makes the system stable.

The presence of integral action slows down the closed loop response of a process. To increase the speed of the closed loop response, the value of controller gain is increased. But, increasing enough k_c in order to have acceptable speeds, the response becomes more oscillatory and may lead to instability [42]. The introduction of the derivative mode brings the stabilizing effect to the system. Thus, acceptable response can be achieved by selecting an appropriate value for the gain k_c while maintaining moderate overshoots and decay ratios.

Due to its robust performance, PID controller can be used in wide range of operating conditions. Appropriate setting of the parameters is called the tuning of a control loop. It is basically the regulation of these control parameters to the optimum values for providing good dynamic response of the system. Apart from minimizing overshoot and steady state error, it can increase the stability of the system. Sometimes designing and tuning a PID controller can be tough in practice if multiple objectives such as short transient and high stability are to be achieved. However, improper tuning can make the system unstable and cause oscillatory behaviour.

According to the reports 80% of PID type controllers in the industry are unsuccessfully tuned, 30% of the PID loops operate in the manual mode and 25% of PID loops in reality operate under default industry settings [58]. PID control schemes provide the simple but effective solutions to most of the control engineering applications today. Since it is simple and easy to design, these controllers are commonly used in the heavy industries to regulate the time domain behavior of different types of dynamic plants.

There are various methods to tune these three parameters. The conventional methods used are Ziegler-Nichols (ZN) tuning which is based on measurement of ultimate gain and period time [64]. But this method could not be accepted by the processes where overshoot is undesirable. Cohen-Coon is another technique which was based on process reaction curve but this method has its own limitations. Techniques like pole-placement, Z-N step response are conventional approaches which were very popular among engineers since these approaches do not require much information about the plant. The drawback of these techniques was that the surety of achieving optimal gains was missing. Kitamori failed to achieve proper response when the system is nonlinear, higher-order system or having large delay.

At the same time, the conventional methods of PID tuning cannot satisfy the requirements of complex control system processes. Hence, meta-heuristic techniques have been introduced in recent years to prove their excellence in design of a PID controller. These methods have proved their effectiveness in giving better results by improving the time domain specifications and performance indices. Also, intelligent approaches have been incorporated by the researchers to make the tuning an easy work.

5.2 BASIC PID CONTROL ALGORITHM

Proportional, Integral and Derivative gains are the three interacting parameters whose suitable tuning provides stable response. The error as well as overshoot is minimized which is important for the system to stabilize. The basic structure of a plant controlled by PID controller is shown in Fig. 5.1.

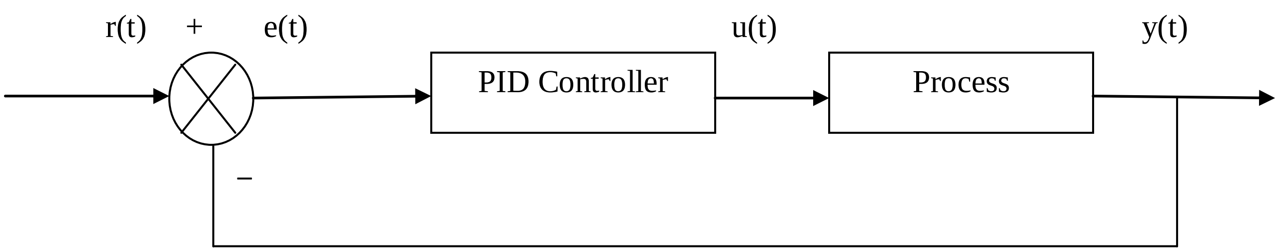

Fig 5.1: A process controlled by PID controller

Where r(t) is the reference point, y(t) is the output of the plant, e(t)=r(t)- y(t) is called the error signal and u(t) is the PID controller output for this particular error signal.

The transfer function of a PID controller has the following form [65]:

$$G(s) = K_p + K_i/s + K_d s \tag{5.1}$$

This form can also be given as

$$G(s) = K_p(1 + \frac{1}{T_i s} + T_d s) \tag{5.2}$$

Where K_p , K_i , and K_d is proportional gain, integral gain and derivative gain respectively.

Where $T_i = K_p / K_i$ and $T_d = K_d / K_p$ Here, T_i is the integral time constant and T_d is the derivative time constant.

For a PID- controlled system, there are basically three performance criteria which are described as:

1) Integral of the square error (ISE)

$$ISE = \int \in^2 (t)dt \tag{5.3}$$

2) Integral of the absolute value of error (IAE)

$$IAE = \int |\in (t)| dt \tag{5.4}$$

3) Integral of the time-weighted absolute error (ITAE)

$$ITAE = \int t|\in (t)| dt \tag{5.5}$$

Where ϵ(t) is the error or deviation from the desired point.

If large errors are to be suppressed strongly, ISE is better than IAE because the errors are squared and thus contribute more to the value of the integral. For the suppression of small errors, IAE is better than ISE because when small numbers are squared (smaller than one) they become even smaller. To suppress errors that persist for long times, the ITAE criteria will tune the controller better because the presence of large‘t’ amplifies the effect of even small errors in the value of the integral.

5.3 TEST PROBLEMS

To demonstrate the effectiveness of the presented two techniques, the following second-order and fourth-order plants are considered which are commonly observed in process control [12].

$$G_1(s) = \frac{e^{-0.5s}}{(s+1)^2} \tag{5.6}$$

$$G_2(s) = \frac{27}{(s+1)(s+1)^3} \tag{5.7}$$

Pade approximation is used to approximate the time delay of process $G_1(s)$. The PID controller designed by the two intelligent algorithms (Genetic and Simulated annealing algorithm) is tested on these plants or processes of very complex dynamic behaviour. If such processes are controlled by the designed robust PID controller then this proves the tuned controller to be efficient for most of the processes of today's application.

5.4. TUNING OF PID BY MOGA AND MOSA

The fitness function is a very significant factor while tuning the PID controller. A combination of different multi-objective fitness functions are studied to show the usefulness of fitness functions:

f_1= 0.05IAE+0.05t_s+0.90OS

f_2= 0.10IAE +0.60OS+0.20t_s+0.10t_r

f_3=0.05ISE+0.05t_s+0.90OS

f_4=0.10ISE +0.60OS+0.20t_s+0.10t_r

f_5=0.05ITAE+0.05t_s+0.90OS

f_6=0.10ITAE +0.60OS+0.20t_s+0.10t_r

f_7=0.10ISE+0.20t_s+0.70t_r

f_8=0.10ISE+0.70t_s+0.20t_r

f_9=0.10ISE+0.10t_s+0.80OS

f_{10}=0.10ITAE+0.10SSE+0.10t_s+0.70OS where t_s is the settling time within 5%, t_r is rise time, OS is the percentage overshoot, ISE is the integral square error, IAE is the integral absolute error and ITAE is the integral time absolute error.

Maximum focus is given to percentage overshoot in fitness function f_1 which means minimum overshoot is the necessity of the system. Similarly, other fitness functions can be interpreted. ISE, IAE and ITAE have been used in different objective functions. So as per the requirement of the system one can assign higher weight to a particular specification while considering the other necessary specifications simultaneously. It should be taken care of that the total summation of weights in a fitness function or objective function should be identical to one so that the overall performance of the system should be defensible. Hence, Multi-objective approach provides the facility of flexibility in PID tuning.

The performance of the MOGA-PID controller is also evaluated on the basis of its convergence rate. The factors or parameters on which the convergence rate depends are like population size, population type etc. Proper selection of these parameters is essential so that the algorithm is timely converged. Table 5.1 shows the parameters and operators of GA used during experimentation of the algorithm.

Table 5.1: Genetic Algorithm operators and parameters

Parameter	**Value/Method**
Population Size	20
Creation function	Feasible population
Selection method	Tournament selection
Mutation fn.	Adaptive feasible
Crossover fn.	Arithmetic crossover
No. of generations	65

Other than these functions used one more stopping criterion has been used that is function tolerance whose value is 1×10-6.

In case of MOSA-PID controller, the initial temperature plays a very important role as it can affect the speed and convergence of the algorithm. Table 5.2 shows the parameters of SA which are used during experimentation in this work.

Table 5.2: Simulated Annealing Algorithm operators and parameters

Parameter	Value/Method
Data type	Double
Annealing function	Fast annealing
Temperature function	Exponential temp.
Acceptance probability fn.	Simulated Annealing acceptance
Max. Iterations	200-250

Other than these functions used, one more stopping criterion is used that is function tolerance whose value is 1×10-6.

CHAPTER-6

SIMULATION RESULTS AND DISCUSSION

This chapter presents the simulation results performed with the use of two intelligent algorithms. The comparison of step responses has been shown for various fitness functions. The specifications like proportional gain, integral gain, derivative gain, maximum overshoot, rise time and settling time for different processes have been calculated and presented in tabular form.

SIMULATION RESULTS AND DISCUSSION

MOGA and MOSA based PID tuning have been applied on different order plants which are commonly used in process control. All the simulations have been performed on MATLAB. The following figures shows the step response of processes $G_1(s)$ and $G_2(s)$ controlled with conventional, single-objective and multi-objective based PID controller.

Fig. 6.1 shows the step response of process $G_1(s)$ controlled with conventional, single-objective and multi-objective based PID controller. It shows the controller tuning with GA using IAE for fitness function f_{1}.

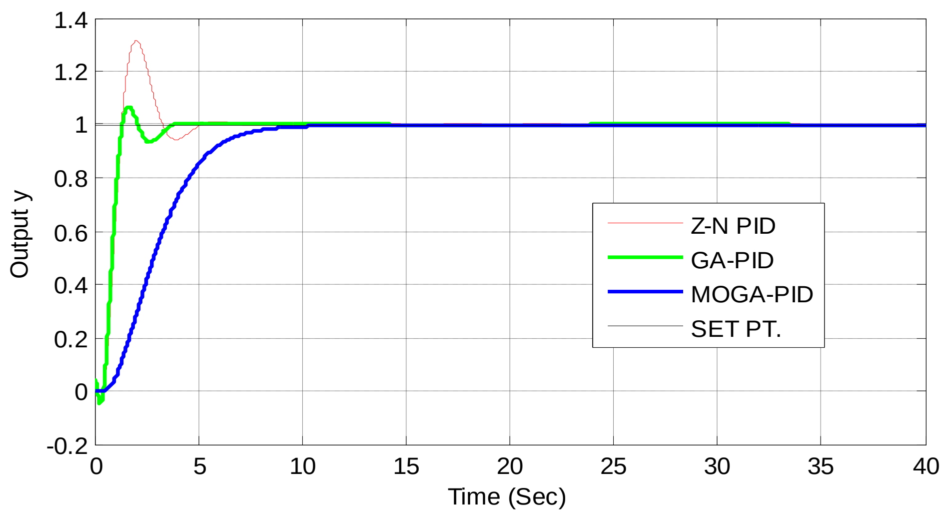

Fig. 6.1: Step response of $G_1(s)$ for f_1 using GA

Fig.6.2 shows the step response of process $G_1(s)$ controlled with conventional, single-objective and multi-objective based PID controller. It shows the controller tuning with SA using IAE for fitness function f_1.

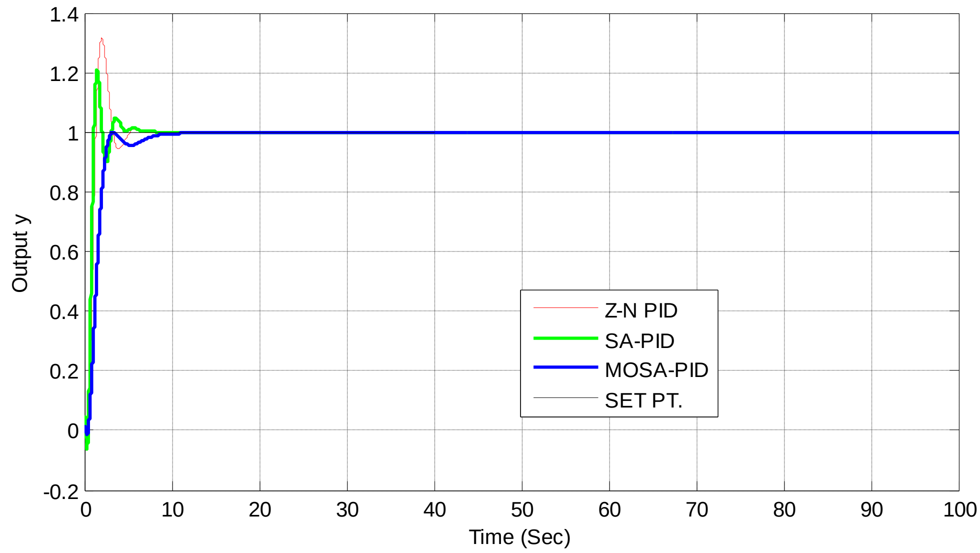

Fig. 6.2: Step response of $G_1(s)$ for f_1 using SA

Fig. 6.3 and Fig. 6.4 show the step response of process $G_1(s)$ controlled with conventional, single-objective and multi-objective based PID controller. It shows the controller tuning with GA and SA using IAE for fitness function f_2.

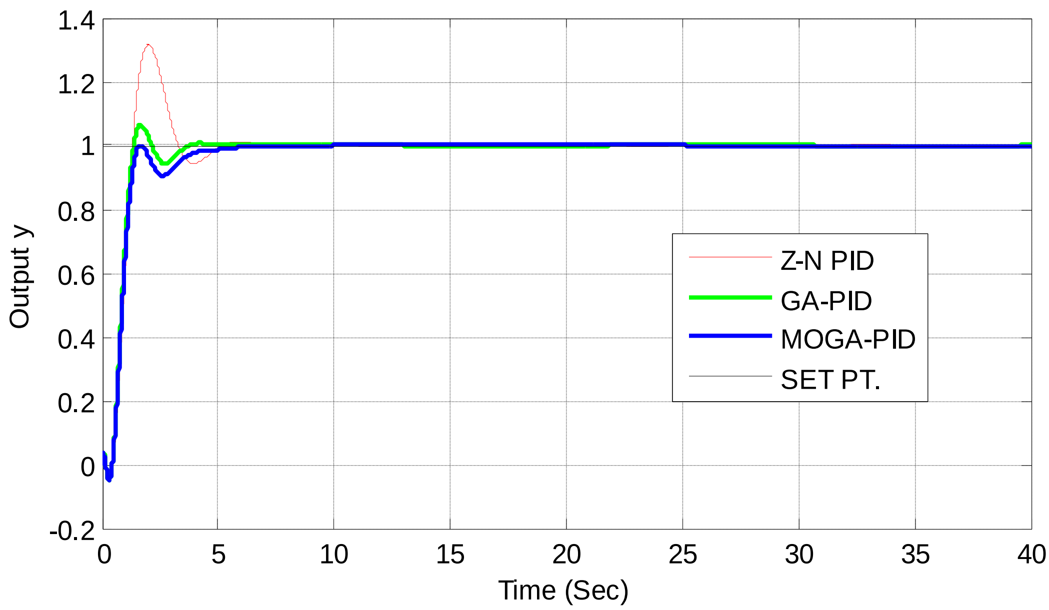

Fig. 6.3: Step response of $G_1(s)$ for f_2 using GA

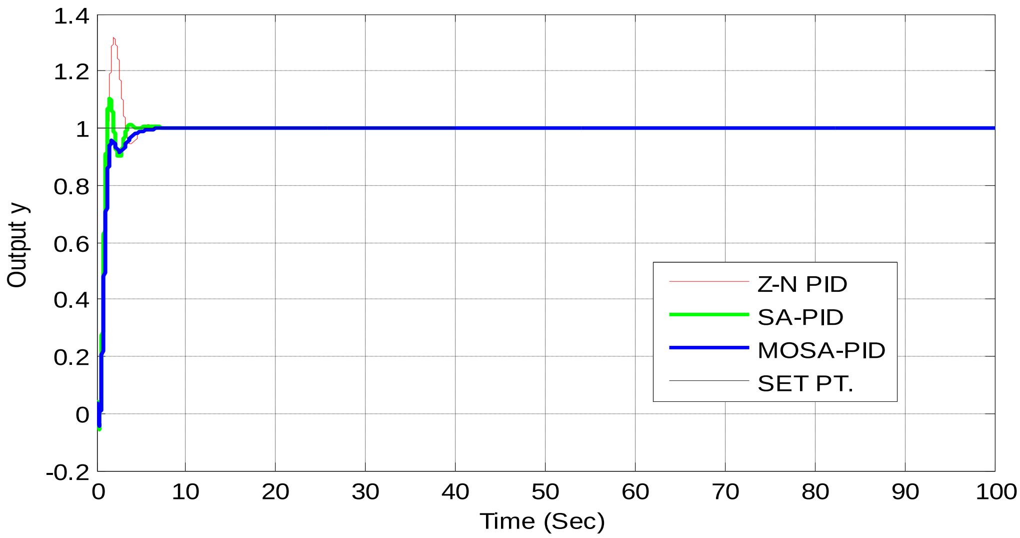

Fig. 6.4: Step response of $G_1(s)$ for f_2 using SA

Fig. 6.5 and Fig. 6.6 show the step response of process $G_1(s)$ controlled with conventional, single-objective and multi-objective based PID controller. It shows the controller tuning with GA and SA using ISE for fitness function f_3.

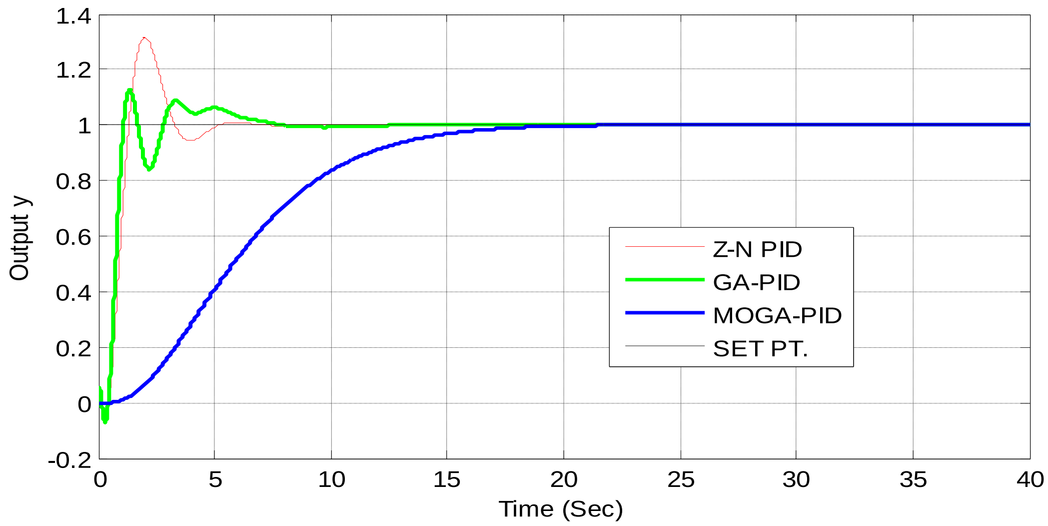

Fig. 6.5: Step response of $G_1(s)$ for f_3 using GA

Here, the step response of fitness function shows that in case of multi-objective fitness functions f_3, the overshoot is minimized in comparison to Z-N technique and single-objective based genetic algorithm. As more emphasis is given to the percentage overshoot, this specification is reduced maximum.

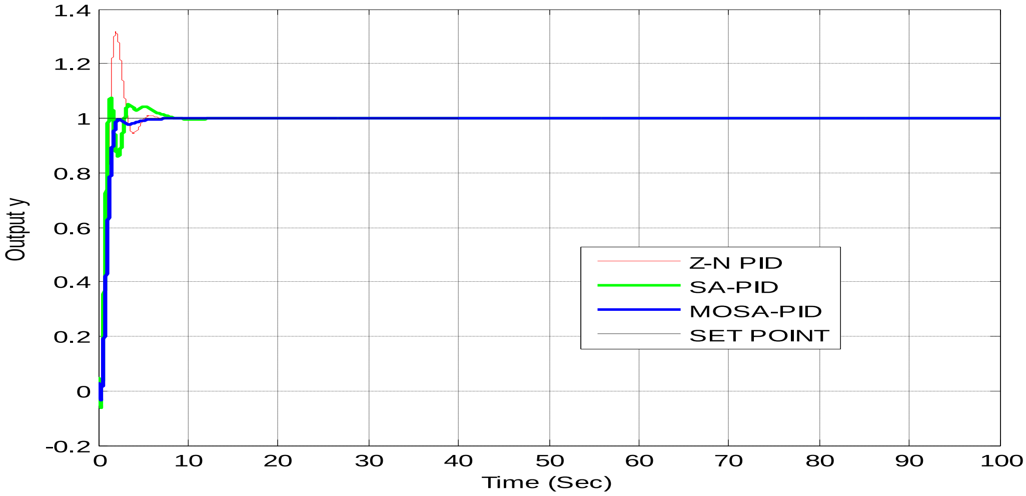

Fig. 6.6: Step response of $G_1(s)$ for f_3 using SA

Fig. 6.7 and Fig. 6.8 show the step response of process $G_1(s)$ controlled with conventional, single-objective and multi-objective based PID controller. It shows the controller tuning with GA and SA using ISE for fitness function f_4.

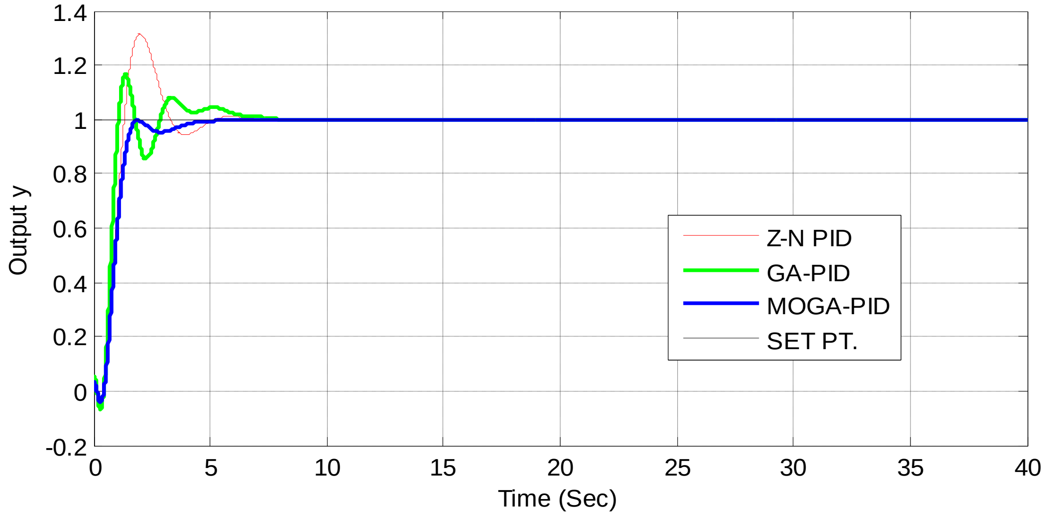

Fig. 6.7: Step response of $G_1(s)$ for f_4 using GA

Here, the step response of fitness function shows that in case of multi-objective fitness functions f_3, the overshoot is minimized in comparison to Z-N technique and single-objective based genetic algorithm. As more emphasis is given to the percentage overshoot, this specification is reduced maximum.

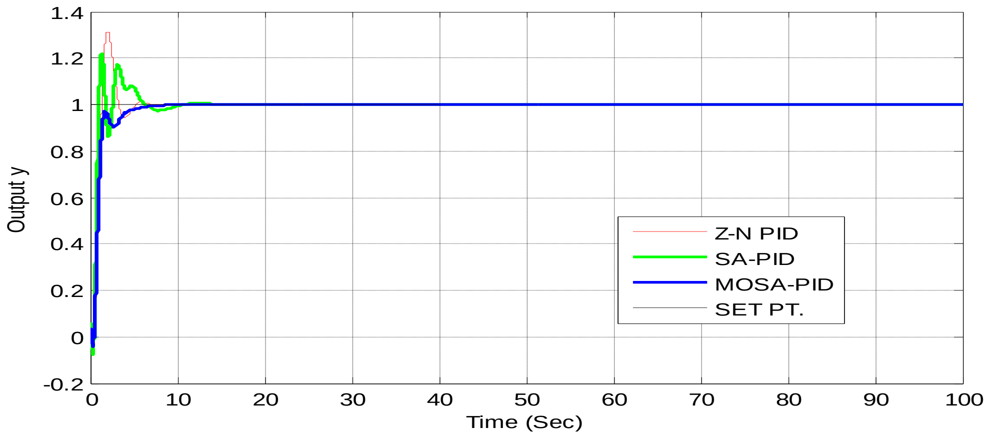

Fig. 6.8: Step response of $G_1(s)$ for f_4 using SA

Fig. 6.9 and Fig. 6.10 show the step response of process $G_1(s)$ controlled with conventional, single-objective and multi-objective based PID controller. It shows the controller tuning with GA and SA using ITAE for fitness function f_5.

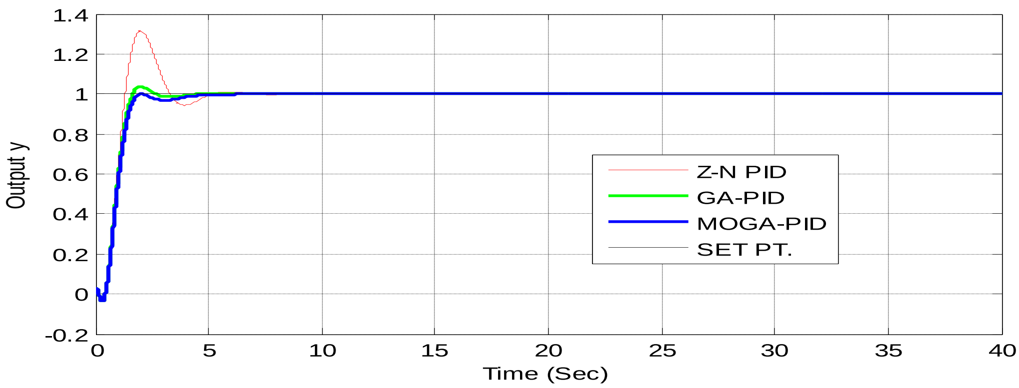

Fig. 6.9: Step response of $G_1(s)$ for f_5 using GA

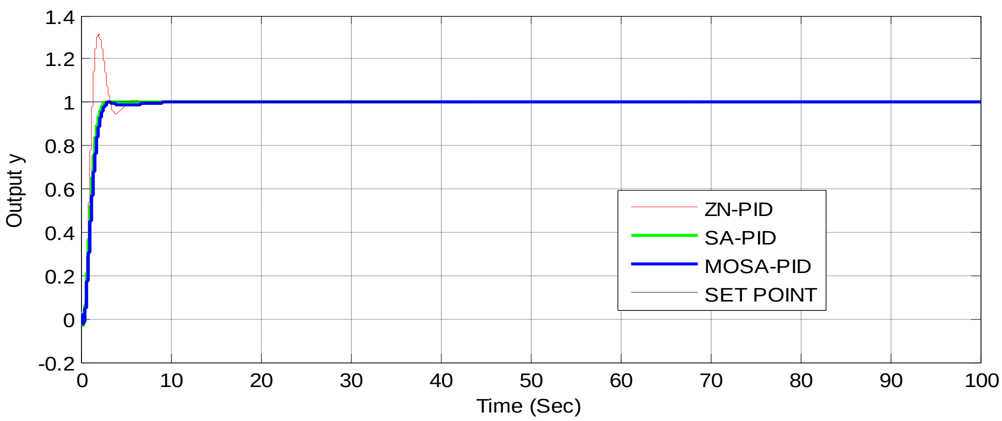

Fig. 6.10: Step response of $G_1(s)$ for f_5 using SA

Fig. 6.11 and Fig. 6.12 show the step response of process $G_1(s)$ controlled with conventional, single-objective and multi-objective based PID controller. It shows the controller tuning with GA and SA using ITAE for fitness function f_6.

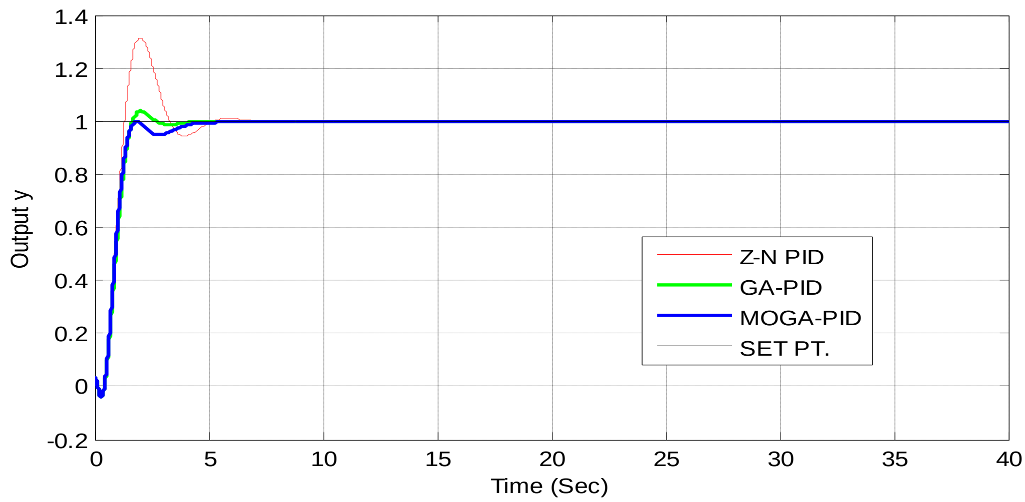

Fig. 6.11: Step response of $G_1(s)$ for f_6 using GA

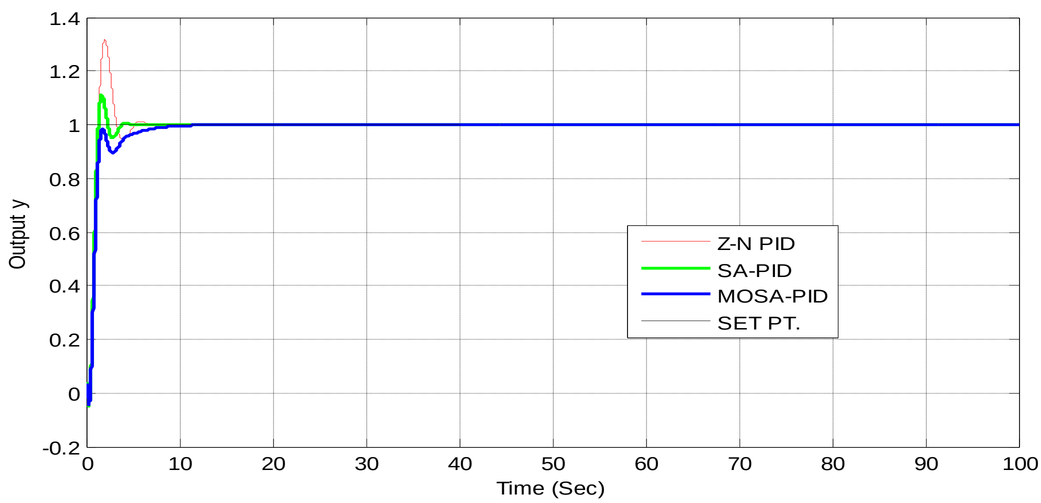

Fig. 6.12: Step response of $G_1(s)$ for f_6 using SA

Fig. 6.13 and Fig. 6.14 show the step response of process $G_2(s)$ controlled with conventional, single-objective and multi-objective based PID controller. It shows the controller tuning with GA and SA using IAE for fitness function f_1.

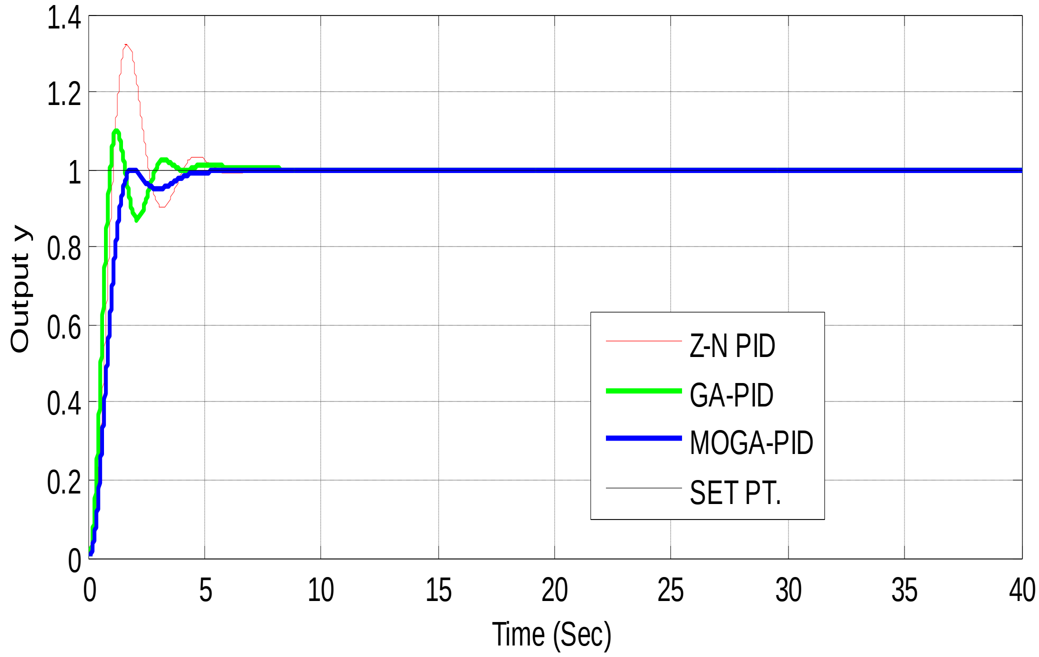

Fig. 6.13: Step response of $G_2(s)$ for f_1 using GA

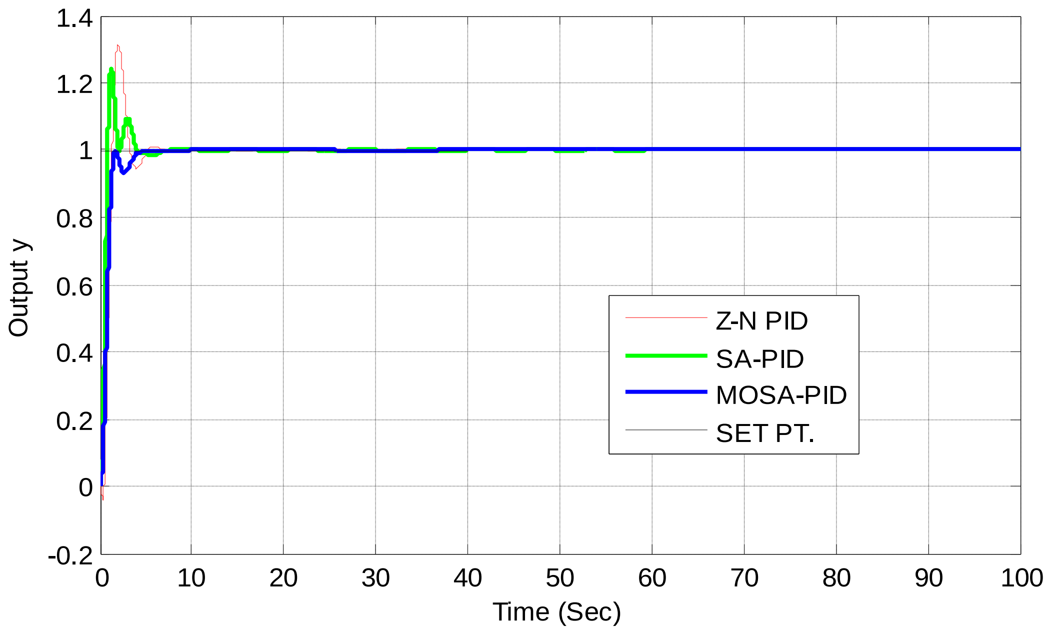

Fig. 6.14: Step response of $G_2(s)$ for f_1 using SA

Fig. 6.15 and Fig. 6.16 show the step response of process $G_2(s)$ controlled with conventional PID controller, single-objective PID controller and multi-objective based PID controller. It shows the controller tuning with GA and SA using IAE for fitness function f_2.

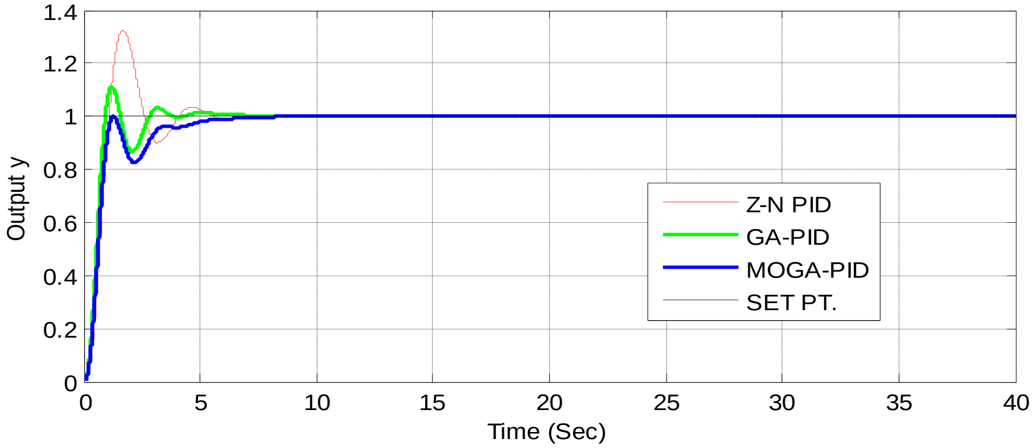

Fig. 6.15: Step response of $G_2(s)$ for f_2 using GA

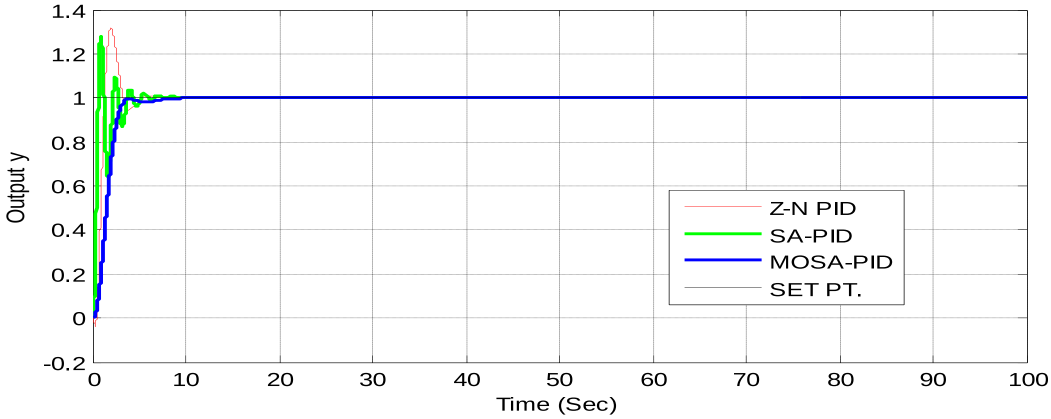

Fig. 6.16: Step response of $G_2(s)$ for f_2 using SA

Fig. 6.17 and Fig. 6.18 show the step response of process $G_2(s)$ controlled with conventional, single-objective and multi-objective based PID controller. It shows the controller tuning with GA and SA using ISE for fitness function f_3.

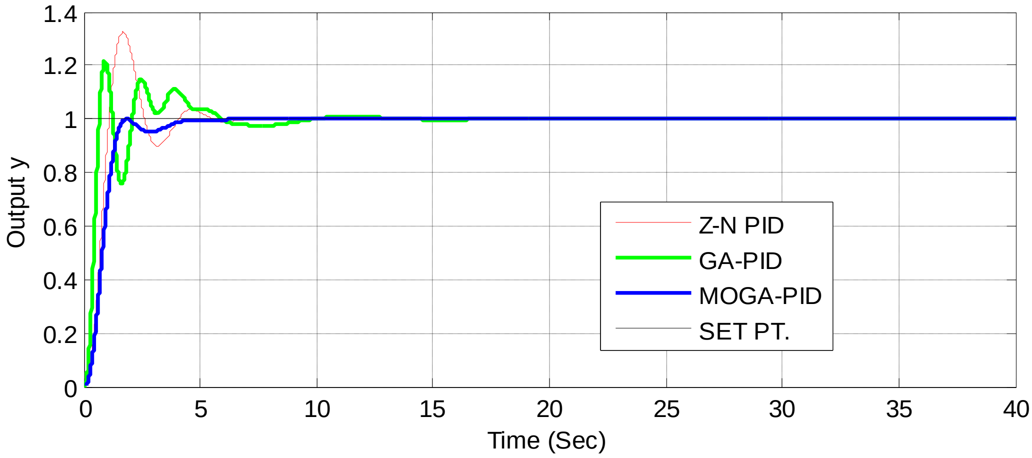

Fig. 6.17: Step response of $G_2(s)$ for f_3 using GA

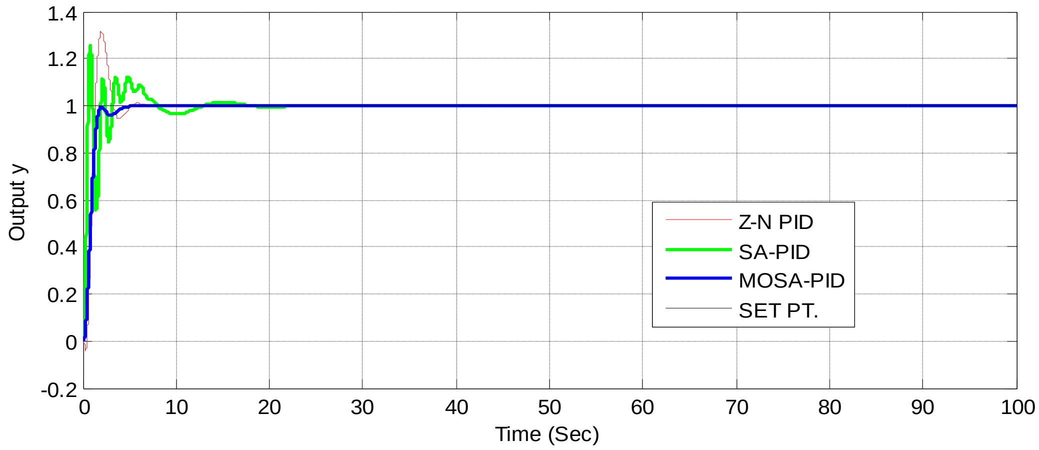

Fig. 6.18: Step response of $G_2(s)$ for f_3 using SA

Fig. 6.19 and Fig. 6.20 show the step response of process $G_2(s)$ controlled with conventional, single-objective and multi-objective based PID controller. It shows the controller tuning with GA and SA using ISE for fitness function f_4.

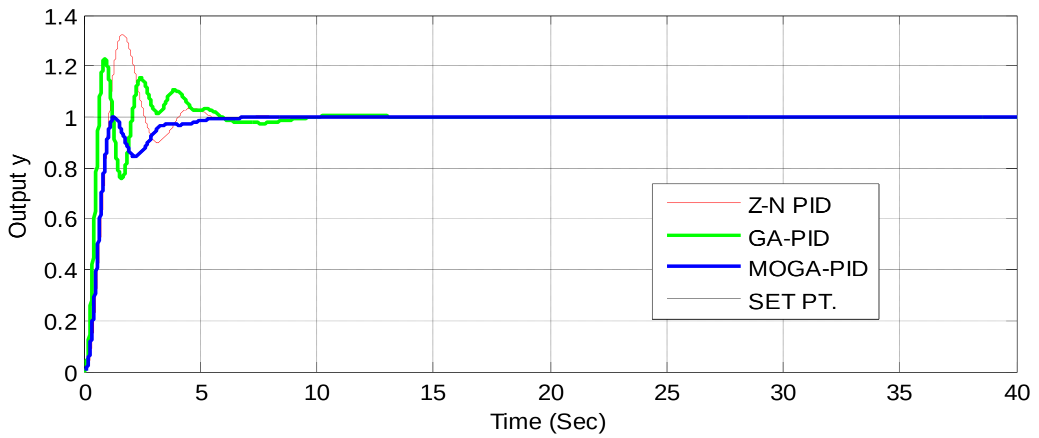

Fig. 6.19: Step response of $G_2(s)$ for f_4 using GA

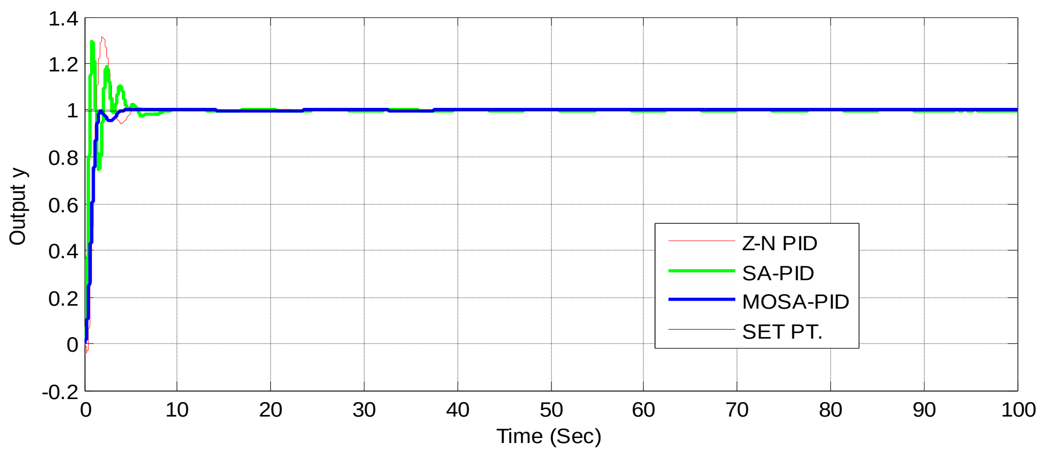

Fig. 6.20: Step response of $G_2(s)$ for f_4 using SA

Fig. 6.21 and Fig. 6.22 show the step response of process $G_2(s)$ controlled with conventional, single-objective and multi-objective based PID controller. It shows the controller tuning with GA and SA using ITAE for fitness function f_5.

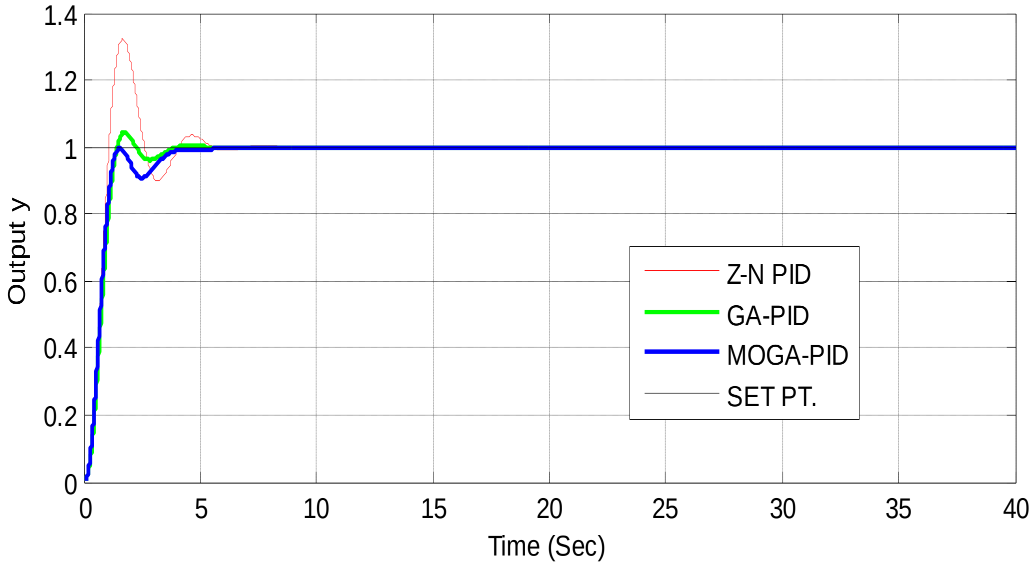

Fig. 6.21: Step response of $G_2(s)$ for f_5 using GA

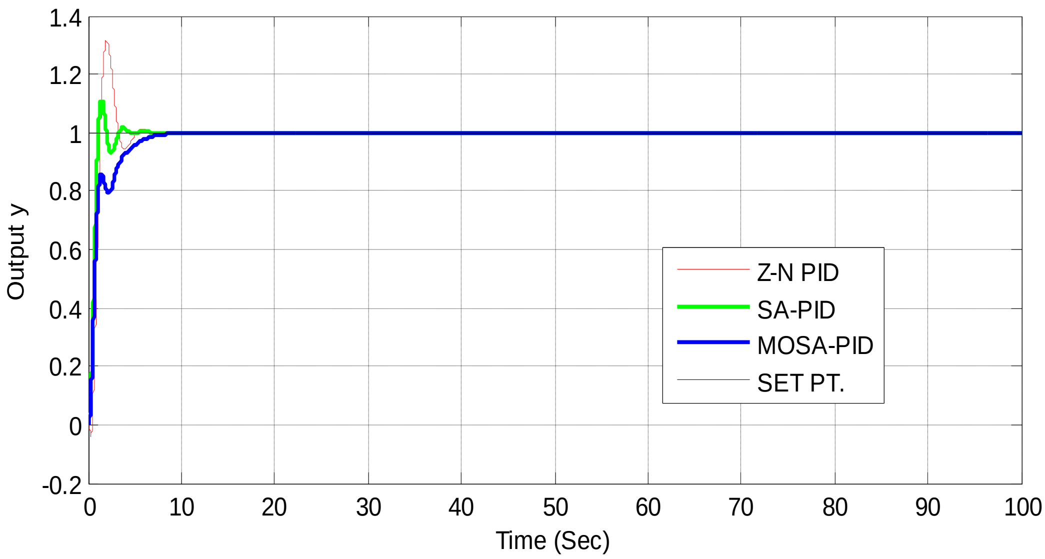

Fig. 6.22: Step response of $G_2(s)$ for f_5 using SA

Fig. 6.23 and Fig. 6.24 show the step response of process $G_2(s)$ controlled with conventional PID controller, single-objective PID controller and multi-objective based PID controller. It shows the controller tuning with GA and SA using ITAE for fitness function f_6.

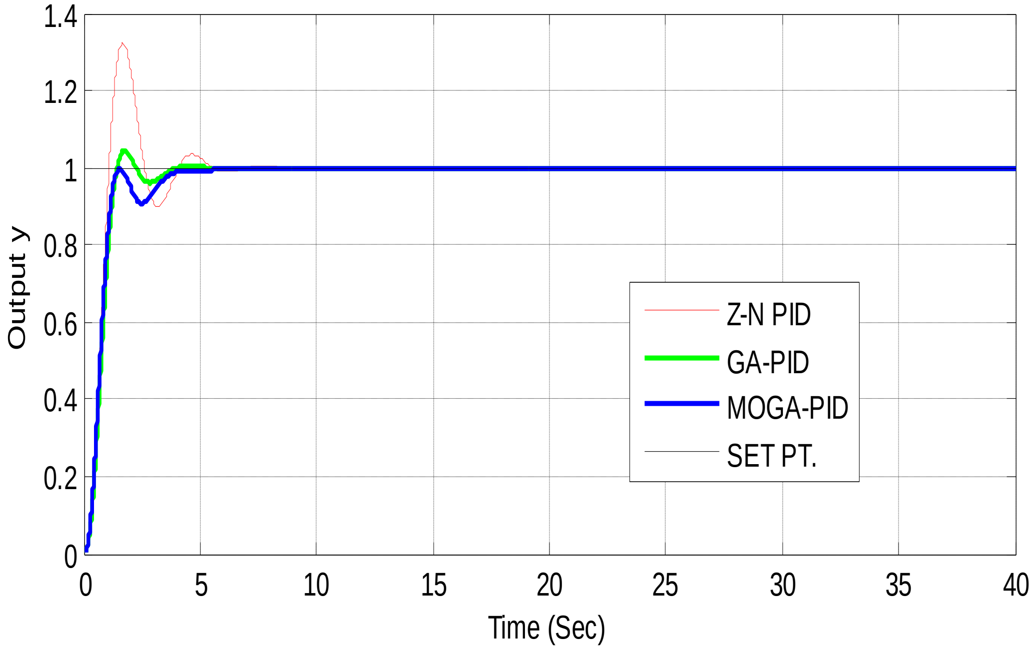

Fig. 6.23: Step response of $G_2(s)$ for f_6 using GA

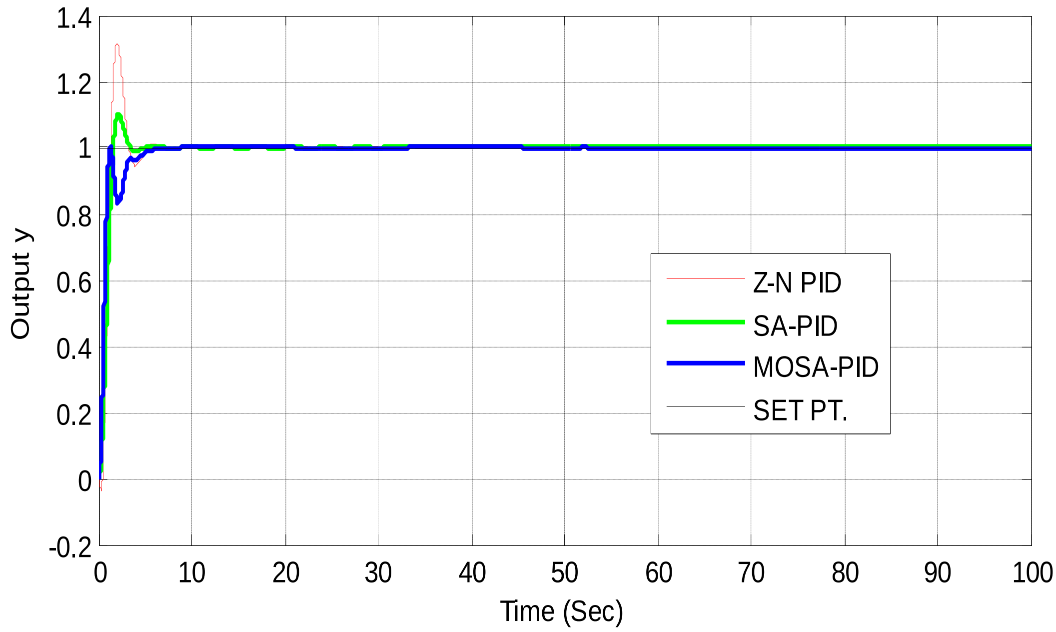

Fig. 6.24: Step response of $G_2(s)$ for f_6 using SA

From these graphs, it is clear that the step responses with MOGA and MOSA shows the minimization of the time domain specification like overshoot (%OS). Also the responses with ITAE show less oscillatory behaviour in comparison to IAE and ISE.

Next, we discuss about the summary of comparative study of different parameters using Z-N technique, single-objective based and Multi-objective based GA and SA.

Table.6.1 and Table.6.2 shows the comparative study of different parameters using GA and SA for fitness function f_1 and f_2 with IAE.

Table 6.1: Comparative study of different parameters for fitness fn. f_1

Process	Index	Z-N	GA-PID	MOGA-PID	SA-PID	MOSA-PID
G1	K_p	2.808	2.4184	0.4399	2.8257	1.4345
	T_i	1.64	2.1370	1.4296	1.5019	0.6179
	T_d	0.41	0.6303	.0293	1.9266	0.4333
	%OS	32	7.039	0	20.6322	0
	t_s(±5%)	4.16	3.0247	6.7020	2.8696	2.5978
	t_r	0.6472	0.6450	4.3810	0.4945	1.6402
G2	K_p	3.072	2.9194	1.9905	3.1995	2.1674
	T_i	1.352	1.7013	1.8751	3.5383	1.1400
	T_d	0.338	0.6917	0.4454	2.2857	1.0985
	%OS	32.8	10.0667	0	24.4071	4.644e-004
	t_s(±5%)	3.722	2.6315	1.5214	3.5504	3.1879
	t_r	0.6649	0.5853	1.0121	0.5186	0.9104

Table 6.2: Comparative study of different parameters for fitness fn. f_2

Process	Index	Z-N	GA-PID	MOGA-PID	SA-PID	MOSA-PID
G1	K_p	2.808	2.3839	2.2116	2.4594	2.0478
	T_i	1.64	2.0874	2.3520	1.2049	0.8869
	T_d	0.41	0.6266	0.6543	1.7750	1.3213
	%OS	32	6.5052	0.0066	9.2014	0.0114
	t_s(±5%)	4.16	2.8896	3.3798	2.9739	3.4355
	t_r	0.6472	0.6604	0.7329	0.5674	0.8651
G2	K_p	3.072	2.9984	2.5091	4.0564	0.9052
	T_i	1.352	1.6773	2.3102	1.8480	0.5976
	T_d	0.338	0.7061	0.6979	3.9044	0.1354
	%OS	32.8	11.4015	0.0036	27.7627	0
	t_s(±5%)	3.722	2.5885	3.1896	3.5880	2.9999
	t_r	0.6649	0.5656	0.6964	0.3887	2.0059

Table.6.3 and Table.6.4 shows the comparative study of different parameters using GA and SA for fitness function f_3 and f_4 with ISE.

Table 6.3: Comparative study of different parameters for fitness fn. f_3

Process	Index	Z-N	GA-PID	MOGA-PID	SA-PID	MOSA-PID
G1	K_p	2.808	2.2257	0.0455	2.1900	1.9314
	T_i	1.64	1.4110	0.2964	1.3542	0.8884
	T_d	0.41	0.9884	0.0634	1.9955	1.0006
	%OS	32	12.5947	0.0269	7.2134	0
	$t_s(\pm 5\%)$	4.16	5.4499	13.7321	2.7650	1.8109
	t_r	0.6472	0.4781	9.2642	0.5362	1.0504
G2	K_p	3.072	2.6684	2.0332	1.9329	1.9134
	T_i	1.352	0.7382	1.8337	2.9251	1.0618
	T_d	0.338	1.4874	0.4725	5.4096	0.8684
	%OS	32.8	21.7203	1.2483e-004	25.6061	0
	$t_s(\pm 5\%)$	3.722	4.5298	1.4708	6.6437	1.5872
	t_r	0.6649	0.3994	0.9770	0.3436	1.0584

Table 6.4: Comparative study of different parameters for fitness fn. f_4

Process	Index	Z-N	GA-PID	MOGA-PID	SA-PID	MOSA-PID
G1	K_p	2.808	2.4557	2.1096	2.3054	2.1230
	T_i	1.64	1.5598	2.2119	2.1490	0.8597
	T_d	0.41	0.8809	0.5842	2.4468	1.3430
	%OS	32	16.3876	0.0019	21.6784	0
	$t_s(\pm 5\%)$	4.16	3.8103	1.5453	5.3614	3.6500
	t_r	0.6472	0.4697	0.8531	0.4259	0.8248
G2	K_p	3.072	2.8295	2.4788	3.3250	2.0277
	T_i	1.352	0.7766	2.1933	4.1578	1.1532
	T_d	0.338	1.4028	0.6712	4.2014	1.0144
	%OS	32.8	23.0777	0.0135	29.2898	0.1150
	$t_s(\pm 5\%)$	3.722	4.4738	3.1383	4.3039	1.4572
	t_r	0.6649	0.3962	0.7163	0.3791	0.9691

In case of Multi-objective based fitness functions, maximum weight is given to the percentage overshoot, which is a time domain specification. Here, the aim is to reduce the overshoot which is also clear from the above tables.

Table.6.5 and Table.6.6 shows the comparative study of different parameters using GA and SA for fitness function f_5 and f_6 with ITAE.

Table 6.5: Comparative study of different parameters for fitness fn. f_5

Process	Index	Z-N	GA-PID	MOGA-PID	SA-PID	MOSA-PID
G1	K_p	2.808	2.1633	2.0610	1.8364	1.5843
	T_i	1.64	2.0698	2.1672	0.9075	0.7454
	T_d	0.41	0.5447	0.5615	0.9470	0.6883
	%OS	32	3.7186	4.372e-004	0.5545	0
	t_s(±5%)	4.16	1.5047	1.6127	1.9056	2.3532
	t_r	0.6472	0.8344	0.9037	1.1238	1.4557
G2	K_p	3.072	2.2239	2.2990	2.8245	1.8186
	T_i	1.352	1.7326	1.9868	1.6389	0.7845
	T_d	0.338	0.4744	0.5506	1.4905	1.3903
	%OS	32.8	4.2389	0.0027	11.3977	0.0070
	t_s(±5%)	3.722	1.3063	3.2012	2.8359	4.8529
	t_r	0.6649	0.8686	0.8318	0.6743	3.0812

Table 6.6: Comparative study of different parameters for fitness fn. f_6

Process	Index	Z-N	GA-PID	MOGA-PID	SA-PID	MOSA-PID
G1	K_p	2.808	2.1570	2.1416	2.5413	2.2029
	T_i	1.64	2.0585	2.2180	1.2031	0.8037
	T_d	0.41	0.5429	0.5957	1.4700	1.2994
	%OS	32	3.8034	0.0071	10.9445	0
	t_s(±5%)	4.16	1.5102	1.5001	2.0916	4.1059
	t_r	0.6472	0.8387	0.8214	0.6382	0.8156
G2	K_p	3.072	2.4685	2.4346	1.9183	2.4942
	T_i	1.352	1.7727	2.1267	1.2615	1.1211
	T_d	0.338	0.5338	0.6494	0.6917	1.7623
	%OS	32.8	5.2529	0.0214	9.9252	0.0261
	t_s(±5%)	3.722	2.8978	3.1397	2.8376	3.1235
	t_r	0.6649	0.7614	0.7410	0.9995	0.6977

It is noticeable from the results that MOGA-PID & MOSA-PID controller is far better than the conventionally tuned and single objective based PID controller for both transient domain specification like percentage overshoot (%OS) or steady state specification like settling time (t_s). A cautious study of the tables clearly shows that multi-objective based controller gives the freedom to tune according to the preferred requirement of the user. To do so, one just requires creating an appropriate fitness function in which a suitable weight is assigned to a particular specification. Figure 6.25 and Figure 6.26 show the step response & Table 6.7 shows the comparative study of different parameters for GA-PID controller system with IAE, ISE and ITAE for $G_1(s)$ and $G_2(s)$ respectively.

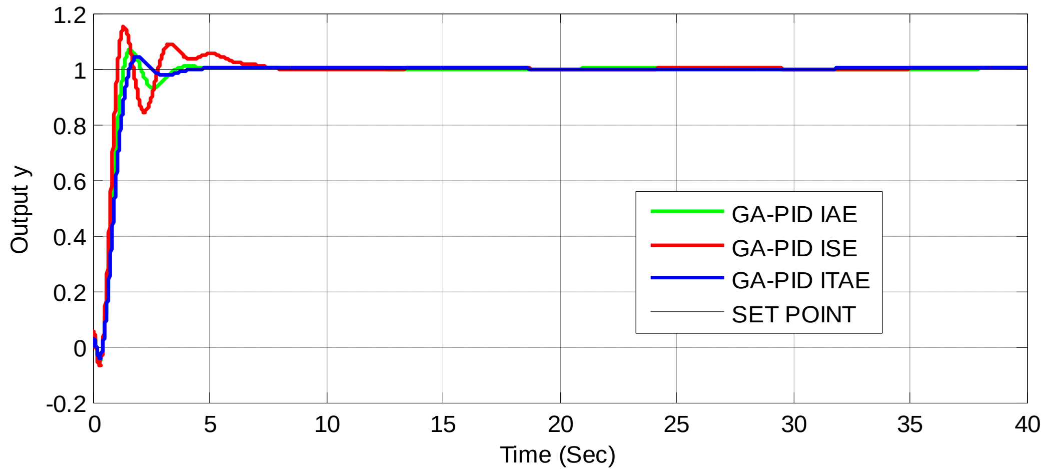

Fig 6.25: Step response of GA-PID controller system with IAE, ISE and ITAE for $G_1(s)$

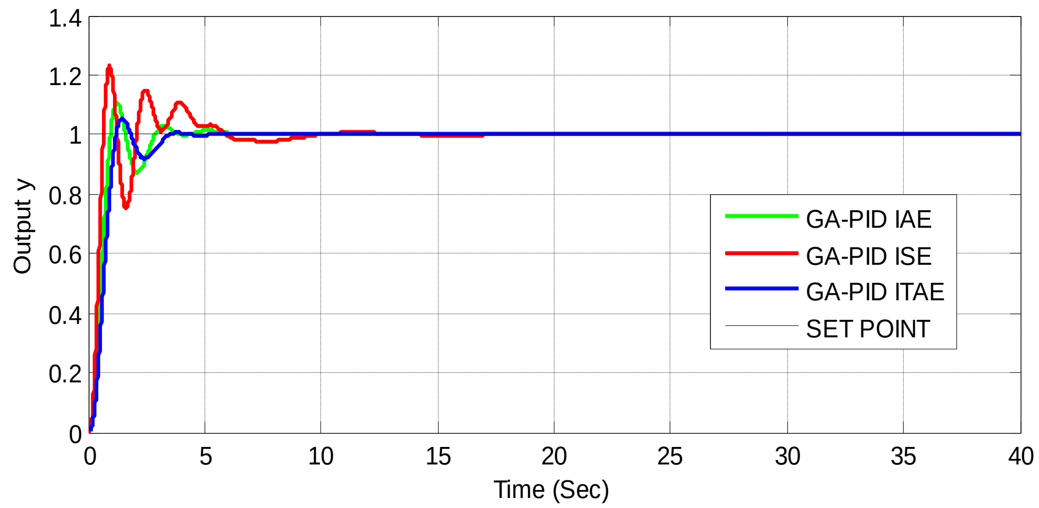

Fig 6.26: Step response of GA-PID controller system with IAE, ISE and ITAE for $G_2(s)$

Table 6.7: Comparative study of different parameters with IAE, ISE and ITAE

Process	Index	IAE	ISE	ITAE
G1	K_p	2.4184	2.2257	2.1633
	T_i	2.1370	1.4110	2.0698
	T_d	0.6303	0.9884	0.5447
	%OS	7.039	12.5947	3.7186
	t_s(±5%)	3.0247	5.4499	1.5047
	t_r	0.6450	0.4781	0.8344
G2	K_p	2.9194	2.6684	2.2239
	T_i	1.7013	0.7382	1.7326
	T_d	0.6917	1.4874	0.4744
	%OS	10.0667	21.7203	4.2389
	t_s(±5%)	2.6315	4.5298	1.3063
	t_r	0.5853	0.3994	0.8686

It is clear from the table that settling time is reduced maximum in case of ITAE in comparison to ISE and IAE. Also the overshoot has reduced substantially in case of ITAE. The overall response of ITAE performance index is better than other results.

Fig 6.27 and Fig.6.28 show the step response & Table 6.7 shows the comparative study of different parameters for SA-PID controller system with IAE, ISE and ITAE for $G_1(s)$ and $G_2(s)$ respectively

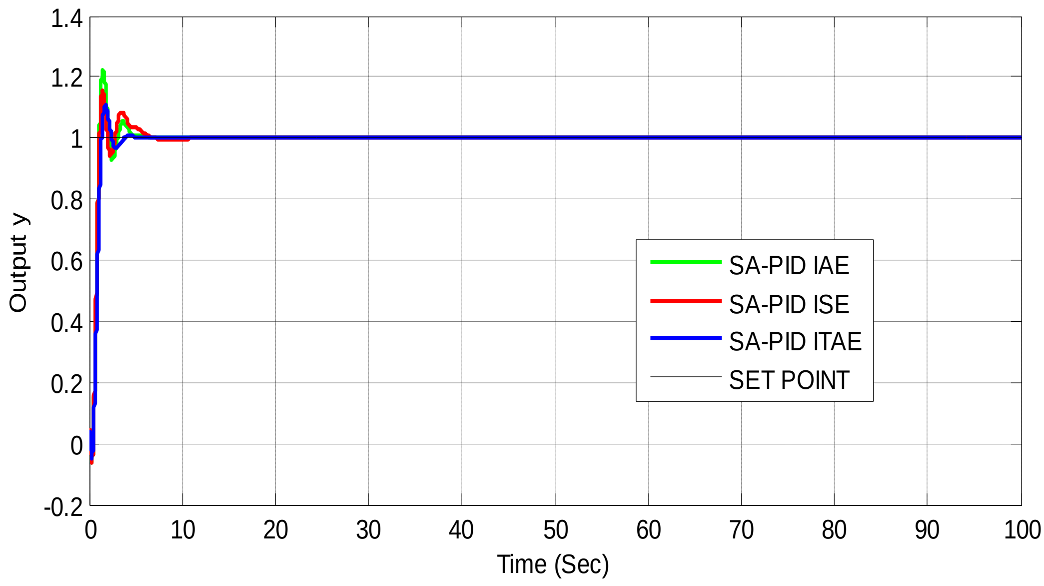

Fig 6.27: Step response of SA-PID controller system with IAE, ISE and ITAE for $G_1(s)$

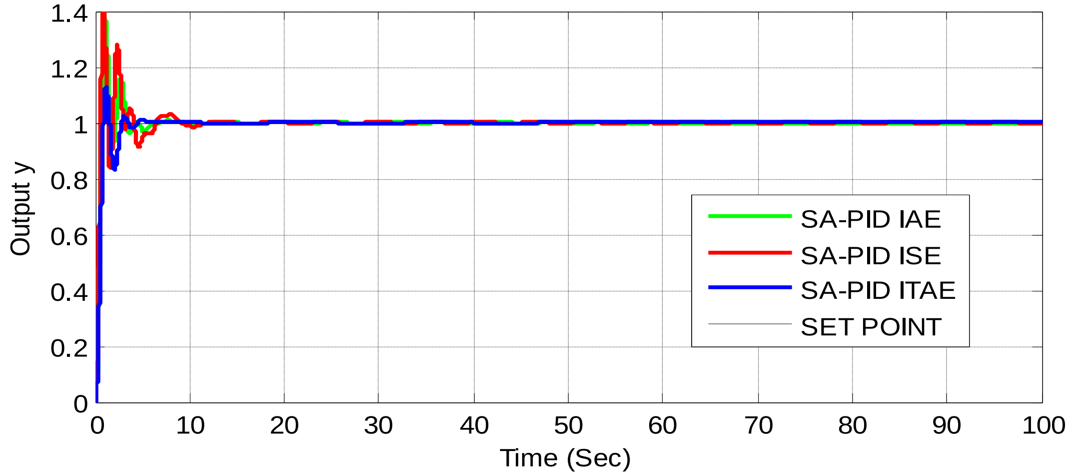

Fig 6.28: Step response of SA-PID controller system with IAE, ISE and ITAE for $G_2(s)$

Table 6.8: Comparative study of different parameters with IAE, ISE and ITAE

Process	Index	IAE	ISE	ITAE
G1	K_p	2.4594	2.1900	1.8364
	T_i	1.2049	1.3542	0.9075
	T_d	1.7750	1.9955	0.9470
	%OS	9.2014	7.2134	0.5545
	$t_s(\pm5\%)$	2.9739	2.7650	1.9056
	t_r	0.5674	0.5362	1.1238
G2	K_p	4.0564	1.9329	2.8245
	T_i	1.8480	2.9251	1.6389
	T_d	3.9044	5.4096	1.4905
	%OS	27.7627	25.6061	11.3977
	$t_s(\pm5\%)$	3.5880	6.6437	2.8359
	t_r	0.3887	0.3436	0.6743

It can be seen from the tables that responses with ITAE show less oscillatory behaviour in comparison to IAE and ISE. The parameters like settling time (t_s) and percentage overshoot (%OS) is less in the case of ITAE using algorithms.

Fig. 6.29 and Fig. 6.30 show the comparison of step response of system $G_1(s)$ controlled with MOGA-PID and MOSA-PID controller respectively in which different fitness functions have been used for the purpose of tuning. Table.6.9 shows the comparative study of MOGA-PID and MOSA-PID for various objective functions.

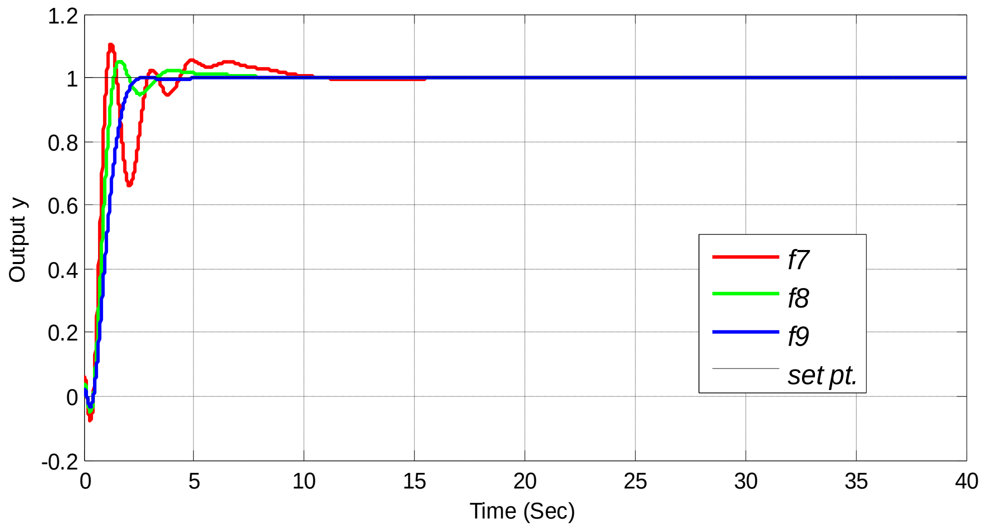

Fig 6.29: Step response of process $G_1(s)$ with MOGA-PID for various objective functions

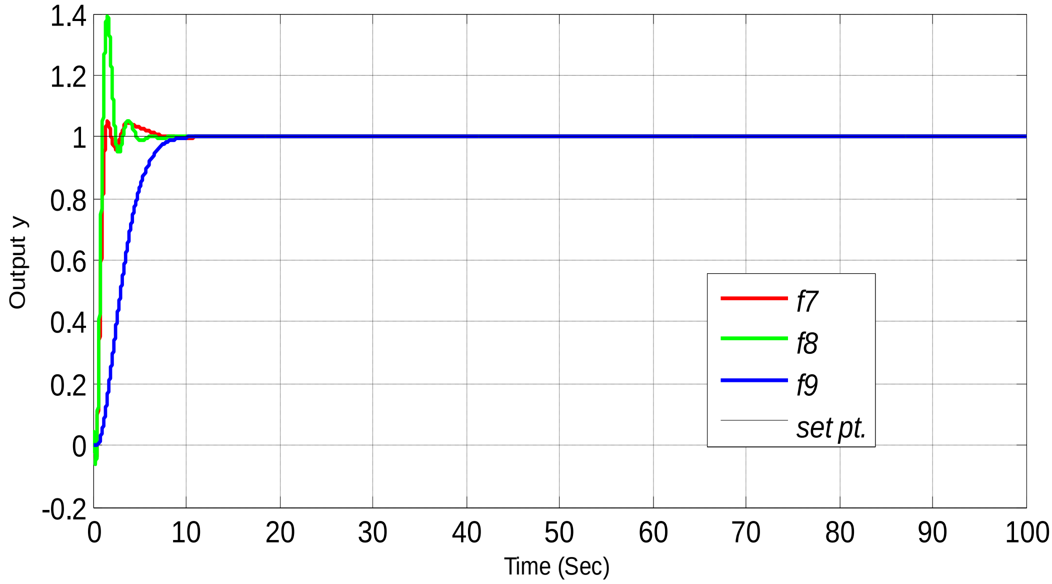

Fig 6.30: Step response of process $G_1(s)$ with MOSA-PID for various objective functions

It can be seen from the graphs that different fitness functions show the dissimilar response as different weights are assigned to a particular specification. Like when maximum weight is assigned to the rise time, which is a time domain specification, the target is to reduce the rise time. Similarly, when maximum weight is assigned to the percentage overshoot or settling time they are minimized. The other fitness functions can also be analysed in the same manner.

Table. 6.9 Comparative study of MOGA-PID and MOSA-PID for various objective functions

Process	Index	f_7	f_8	f_9
G1 With GA	K_p	1.8913	2.2962	1.8065
	T_i	1.5930	1.8994	2.0436
	T_d	1.3239	0.6790	0.5146
	%OS	10.5826	5.2392	3.37e-005
	t_s(±5%)	2.7499	1.2651	1.9627
	t_r	0.4412	0.6536	1.1572
G1 With SA	K_p	2.1923	2.1938	1.2537
	T_i	1.2022	1.3301	0.3393
	T_d	1.3177	1.6368	2.4997
	%OS	4.9415	4.8426	0.0196
	t_s(±5%)	1.4079	1.2529	11.3846
	t_r	0.7642	0.6459	0.5074

Since many industrial processes involve temperature control for their production. Applications like temperature process control of an electric heating furnace, resistance furnace temperature system requires to maintain the proper temperature. Mathematically, the transfer function of these type of systems is represented by a first-order system.

$$G(s) = \frac{Ke^{-\tau s}}{Ts+1} \tag{6.1}$$

Where τ is the delay time, K is the static gain and T is the time constant.

So, the example of Temperature control system is considered which is approximated by a first order plus time delay system is given as [66]:

$$G_3(s) = \frac{80e^{-20s}}{150s+1} \tag{6.2}$$

The fitness function used for this plant in given by f_{10}.

f_{10}=0.10ITAE+0.10SSE+0.10t_s+0.70OS

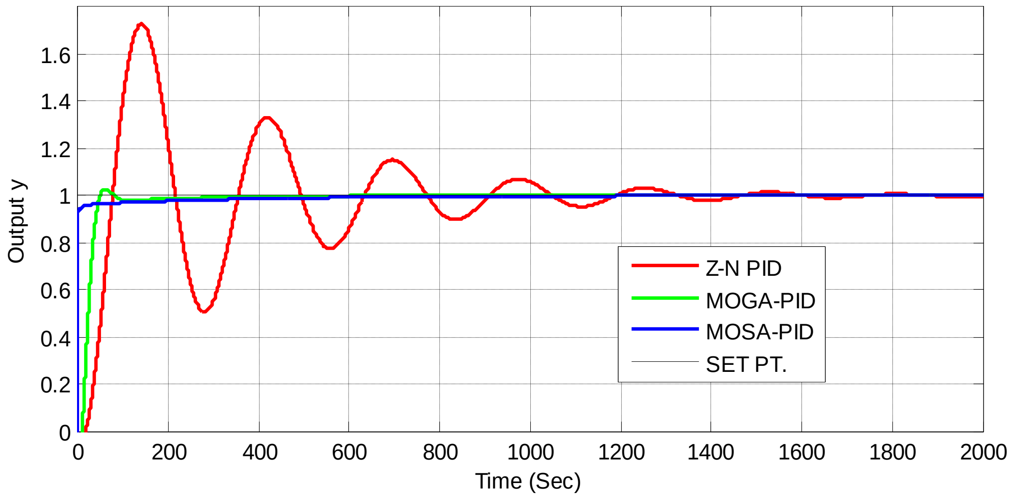

Fig. 6.31: Step response of $G_3(s)$ for f_{10} using GA and SA

Table. 6.10 Comparative study of MOGA-PID and MOSA-PID for f_{10}

Process	Index	Z-N PID	MOGA-PID	MOSA-PID
G3	K_p	0.016	0.0766	0.1860
	T_i	0.0009	4.02e-004	5.05e-004
	T_d	0.001	0.3641	1.854
	%OS	72.4011	2.6524	0
	t_s(±5%)	1.009e003	28.8772	0.471
	t_r	44.955	23.6181	0.343

A careful observation of the table 12 clearly shows that the overshoot is minimized in case of MOGA-PID and MOSA-PID. Also, the results obtained from Simulated annealing algorithm are better than Genetic algorithm. The conventional tuning technique like Z-N does not produce satisfactory results. Also, the rise time and settling time is reduced much in the case of MOSA-PID controller.

CHAPTER-7

CONCLUSIONS & FUTURE SCOPE OF WORK

After discussing the various aspects of this project, the next most important thing is to draw the conclusions from this research work. This chapter discusses the various conclusions drawn and further also outlines the scope of future research work in the same context.

7.1 MAIN CONCLUSIONS

In this thesis work, study has been performed between Multi-objective based PID controller, single-objective based PID controller and conventionally tuned PID controller. To demonstrate the effectiveness of the techniques used, different systems with a very complex dynamic behavior are considered. Step responses of different processes controlled with conventional and Multi-objective based PID controller is presented along with the controller using ISE, IAE and ITAE. Results show that the performance of Multi-objective based tuning of PID controller is much better in comparison to the single-objective based and conventional tuning approaches.

The controller shows flexibility in PID tuning using Multi-objective based approach. So this can be considered as an influential tuning scheme for PID controllers. It gives the freedom to the users to tune the controller according to the necessity. The main advantage of using Multi-objective approach is that it is independent from the intricacy of fitness function under consideration. By doing so, one can consider various objectives rather than limited to single-objective as in conventional tuning methods. Hence, the presented method is appropriate and effective to be applied in different processes.

7.2 FUTURE SCOPE OF WORK

As the fractional order systems are more complex, the controller requires more tuning parameters than that of integer-order controller. By doing so, the tuning efficiency of the controller can be improved. Fractional order PID controller has the better control effect than the integer-order controller. Fractional order PID controllers can be used to compare the step responses with the integer model PID controller.

Due to the interacting parameters (proportional, integral and derivative gain) it is difficult to tune the three parameters appropriately. So a type of Genetic Simulated annealing algorithm approach can be used to optimize the parameters of PID controllers. This approach combines the merits of Genetic Algorithm (GA) and Simulated Annealing Algorithm (SA). By combining the search ability of both the intelligent algorithms, global optimal solution of the problem can be achieved. It will eliminate the effect of premature convergence.

REFERENCES

[1] J. G. Ziegler and N. B. Nichols, “Optimum setting for Automatic Controllers,” ASME Transaction, vol. 64, pp. 759-768, 1942.

[2] G. H. Cohen and G. A. Coon, “Theoretical investigation of retarded control to tune PID controllers,” ASME Transaction, vol. 75, pp. 827-834, 1953.

[3] T. Kitamori, “A method of control system design based upon incomplete knowledge about controlled processes,” Transaction SICE Japan, vol. 15, pp. 549-555, 1979.

[4] G. Stephanopoulos, Chemical process control: an introduction to theory and practice. Prentice-Hall, 1984.

[5] M. Morari and E. Zafiriou, Robust Process Control, Prentice-Hall, Englewood Cliffs, New Jersey, 1989.

[6] D.E Goldberg, Genetic Algorithms in Search, Optimization and Learning, Addision- Wesely , 1989.

[7] D. Lawrence, Hand Book of Genetic Algorithms, Van Nostrand Reinhold, 1991.

[8] B.D Tyreus and W.L. Luyben, “Tuning PI controllers for integrator/dead time processes,” English Chem. Res. pp. 2628–2631, 1992.

[9] J.H. Holland, Adaptation in Natural and ArtifIcial Systems. University of Michigan Press, MIT Press, 1992.

[10] D.P. Kwok. "Optimal fuzzy PID control based on genetic algorithm", Proceedings of the 1992 International Conference on Industrial Electronics Control Instrumentation and Automation, vol.3 1992.

[11] M. Zhuang and D. P. Atherton, “Automatic tuning of optimum PID controllers,” IEEE Proceedings on Control and Applications, vol. 140, pp. 216-224, 1993.

[12] Z.-Y. Zhao, M. Tomizuka, and S. Isaka, “Fuzzy gain scheduling of PID controllers,” IEEE Transaction on Systems, Man and Cybernetics, vol. 23, no. 5. pp. 1392–1398, 1993.

[13] D. W. Pessen, “A new look at PID-controller tuning,” Journal of Dynamical Systems Measures and Control, vol. 116, pp. 553-557,1994.

[14] W. K. Ho, C. C. Hang, and L. S. Cao, “Tuning of PID controllers based on gain and phase margin specifications,” Automatica, Vol. 31, pp. 497-502, 1995.

[15] D. DasGupta and Z. Michalewicz, “Evolutionary Algorithms in Engineering Applications,” Berlin, Germany: Springer-Verlag, 1997.

[16] C.W. Steidley, "Stochastic methods in engineering design", Northcon Conference Proceedings, pp. 108-113, 1998.

[17] R.A Gallego, A. Monticelli and R. Romero, "Transmision system expansion planning by an extended genetic algorithm", IEE Proceedings - Generation Transmission and Distribution, Issue 3, vol. 145, pp. 329-335, 1998.

[18] Wei Li, “Design of a hybrid fuzzy proportional integral plus conventional derivation controller,” IEEE transactions on Fuzzy systems, vol. 6, no.4, pp. 449-461, 1998.

[19] Q. G. Wang, T. H. Lee, H. W. Fung, Q. Bi, and Y. Zhang, “PID tuning for improved and effective performance,” IEEE Transactions on Control Systems Technology, Vol. 7, pp. 457-465, 1999.

[20] C. J. Wu, G. Y. Liu, M. Y. Cheng, and T. L. Lee, “A neural network-based method for fuzzy parameter tuning of PID controllers,” J. Chinese Ins. Eng., vol. 25, no. 3, pp. 265-276, 2002.

[21] B. W. Bequette, Process Control: Modeling, Design, and Simulation, Prentice Hall PTR, 2003.

[22] R. L. Haupt, S. E. Haupt, Practical Genetic Algorithms, Wiley, 2004.

[23] J.B. Schamburg, D.E. Brown. "Simulated Annealing for Selection of Experimental Regions in Response Surface Methodology Applications", Proceedings of the Simulation Conference, vol. 1, 2004.

[24] Michel Gendreau, "Metaheuristics in Combinatorial Optimization", Annals of Operations Research, Issue 1, vol. 140, pp. 189-213, 2005.

[25] Chunna Zhao and Dingu Xue, Yang Quan Chen, “A fractional order PID Tuning Algorithm for a class of Fractional order plants”, proceedings of IEEE International conference on Mechatronics and Automation, 2005.

[26] Ibidapo-Obe, Oye, and Sunday Asaolu. "Optimization Problems in Applied Sciences : From Classical through Stochastic to Intelligent Metaheuristic Approaches", Industrial Innovation, 2005.

[27] F. Rothlauf, Representations for Genetic And Evolutionary Algorithms. Springer-Verlag Berlin/Heidelberg, 2006.

[28] R.C. Leou, "A new method for unit maintenance scheduling considering reliability and operation expense", International Journal of Electrical Power and Energy Systems, vol. 28, pp 471-481, 2006.

[29] A. Mirza, Sarfraz Hussain. "PID Controller with Effective Integral Control Ensures Robustness against Bounded Uncertainties", 2006 IEEE International Symposium on Industrial Electronics, vol. I, pp. 142-147, 2006.

[30] L. Galotto Junior, J.O.P Pinto, J.A Botture and G. Lambert-Torres, "Recursive Least Square and Genetic Algorithm Based Tool for PID Controllers Tuning", International Conference on Intelligent Systems Applications to Power Systems, pp 1-6, 2007.

[31] M. Willjuice Iruthayarajan, S. Baskar, "Optimization of PID parameters using Genetic Algorithm and Particle Swarm optimization", International conference on Information and Communication Technology in Electrical Sciences, pp. 81-86, 2007.

[32] Shady M. Gadoue, D.Giaouris and J.W. Finch "Genetic Algorithm Optimized PI and Fuzzy Sliding Mode Speed Control for DTC Drives", World Congress on Engineering, Vol. 1, 2007.

[33] Xiang Zhong Mang and Baoye Song, "Fast Genetic Algorithms Used for PID Parameter Optimization", proceedings of IEEE International Conference on Automation and Logistics, pp. 2144-2148, 2007.

[34] S. N. Sivanandam and S. N. Deepa, Introduction to Genetic Algorithms. Springer-Verlag Berlin Heidelberg, 2007.

[35] Urvinder Singh, Munish Rattan and Narwant Singh,"Optimization of gain and impedance of Yagi-Uda antenna using simulated annealing", 19th International Conference on Applied Electromagnetics and Communications, pp 1-4, 2007.

[36] Sigurdur Olafsson, Li Xiaonam and Wu Shuning, "Operations research and data mining", European Journal of Operational Research, pp. 1429-1448, 2008.

[37] Jin-Sung Kim, Jin-Hwan Kim, Ji-Mo Park, Sung-Man Park, Won-Yong Choe and Hoon Heo, "Auto Tuning PID Controller based on Improved Genetic Algorithm for Reverse Osmosis Plant", International Journal of Intelligent Technology, vol.2, no. 11, 2008.

[38] S. Sumathi, T. Hamsapriya, and P. Surekha, Evolutionary Intelligence. Springer-

Verlag Berlin Heidelberg, 2008.

[39] B. Khaki, B. Mozafari, A. Mehrtash, R. Sirjani and Parastar, "Optimal controller design of SVC for system stability improvement", IEEE International Conference on Sustainable Energy Technologies, pp. 1277-1281, 2008.

[40] Rahul Swarnkara, & A. Jennifer, Hardinga, "Modelling and optimisation of a product recovery network", International Journal of Sustainable Engineering, Issue 1, vol. 2, p. 40-55, 2009.

[41] Liu Fan, Er. Meng Joo, "Design for auto-tuning PID controller based on genetic algorithms", 4th IEEE Conference on Industrial Electronics and Applications, pp. 1924-1928, 2009.

[42] A. O"Dwyer, Handbook of PI and PID Controller Tuning Rules. Imperial College Press, 2009.

[43] Anupam Shukla, Ritu Tiwari and Rahul Kalu, "Real life applications of soft computing", 2010.

[44] B.Nagaraj, Dr. N. Murugananth, "A comparative study of PID controller Tuning using GA, EP, PSO and ACO", IEEE conference, 2010.

[45] Balram Sumana, Nazish Hodaa, Shweta Jhab, "Orthogonal Simulated annealing for Multiobjective optimization", Computers & chemical Engineering, Elsevier, Issue 10, vol. 34, pp. 1618-1631, 2010.

[46] S.M GirirajKumar, Bodla Rakesh, N.Anantharaman, "Design of Controller using Simulated Annealing for a Real Time Process", International Journal of Computer Applications (0975 – 8887) Volume 6– No.2, September 2010.

[47] S. M. Giriraj Kumar, Bodla Rakesh, N.Anantharaman, "Design of controller using simulated annealing for a real time process", International Journal of computer applications, vol. 6, 2010.

[48] Farhad Jafari, Ali Dastfan, R. Tahmasebi and M. Rahideh, "Single-phase PWM rectifier parameters optimized by using the intelligent method", IEEE International Conference on Computational Technologies in Electrical and Electronics Engineering, pp. 677-681, 2010.

[49] Jun Liu, Jiang Zhu, "Intelligent Optimal Design of Transmission of Cooling Fan of Engine", Third International Conference on Information and Computing, vol. 4, pp 101-104, 2010.

[50] Mohammad Dashti J, Kambiz Shojaeee G., S.M.H Seyedkashi, M. Behnam T, “Novel Simulated Annealing Algorithm in order to optimal adjustment of Digital PID controller”, 11th International Conference Control, Robotics and Vision, 2010.

[51] M.F Nor Shah, M.A.Zainal, A.Faruq, S.S.Abdullah, “Metamodelling Approach for PID controller optimization in an Evaporator process”, 4th International conference on modelling simulation & applied optimization (ICMSAO), pp. 1-4, 2011.

[52] Ali, Mohammed Obaid; Koh, S. P.; Chong, K. H. and Hamoodi, Asmaa Salih. "Design a PID Controller of BLDC Motor by Using Hybrid Genetic-Immune", Modern Applied Science, vol. 5, no.1, pp.75, 2011.

[53] Sadegheih, A.. "Optimal design methodologies under the carbon emission trading program using MIP, GA, SA, and TS", Renewable and Sustainable Energy Reviews, 2011.

[54] S. M Girirajkumar, A. Atal Kumar and N. Anantharaman, "Application of Simulated Annealing for Tuning of a PID Controller for a Real Time Industrial Process", Journal of Computer Science & Control Systems, 2011.

[55] Aman Kumar, K. Aswani Dhingra, "Optimization of Scheduling Problems: A genetic algorithm survey", International Journal of Applied Science and Engineering Research, Issue 2, vol. 1, pp 11-25, 2012.

[56] R. Bindu, K. Mini Namboothiripad, “Tuning of PID Controller for DC Servo Motor using Genetic Algorithm”, International Journal of Emerging Technology and Advanced Engineering, ISSN: 2250-2459, Volume 2, Issue 3, March 2012.

[57] Claudio Fabiano Motta Toledo, Joao Migwel. G. Lima, Marcio da Silva Arantes, “A Multi-population Genetic Algorithm Approach for PID controller Auto-tuning”, 2012.

[58] Gholamreza Farahani, “Computing optimum coefficients of PID controller”, 38th Annual conference on IEEE Industrial Electronics society, 2012.

[59] Wei Jiang and Xuchu Jiang, “Design of an Intelligent Temperature Control System based on the Fuzzy self-tuning PID”, International Symposium on safety science and Engineering, Science Direct, 2012.

[60] Shoubin Wang, Na Li, Fan Yang, “Resistance Furnace Temperature system on

Fuzzy PID controllers", Journal of Information and computational science, vol. 9, pp 2627-2634, 2012.

[61] Dr. K. Babulu, D. Kranthi Kumar, "Fuzzy self-adaptive PID controller Design for electric heating furnace", International Journal of Engineering Inventions", ISSN: 2278-7461, vol. 1, Issue 5, pp. 10-21, 2012.

[62] Shi Dequan, Gao Guili, Gao Zhiwei, Xiao Peng, "Application of expert Fuzzy PID method for temperature control of heating furnace", International workshop on Information and Electronics Engineering, Science Direct, pp. 257-261, 2012.

[63] Pranay Lahoty, Girish Parmar, "A Comparative Study of Tuning of PID Controller using Evolutionary Algorithms", International Journal of Emerging Technology and Advanced Engineering ISSN 2250-2459, Volume 3, Issue 1, January 2013.

[64] Mr.Jyotiprakash Patra1 , Dr.Partha Sarathi Khuntia and Mrs. Sumitra Samal, "Analysis and Comparison of different performance index factor for Conventional PID & GA plus PID Controller", International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS), pp. 242-250, 2013.

[65] Arroyo, José M., and Francisco J. Fernández. "Application of a genetic algorithm to n K power system security assessment", International Journal of Electrical Power & Energy Systems, 2013.

[66] Papavassiliou, . "Cognitive Methods and Evolutionary Computing", Evolutionary Dynamics of Complex Communications Networks, 2013.

[67] Yogendra Kumar Soni, Rajesh Bhatt, "Simulated Annealing optimized PID controller design using ISE, IAE, IATE and MSE criteria", International Journal of Advanced Research in Computer Engineering &Technology, vol. 2, 2013.

[68] Yogesh Chaudhari, "Design and implementation of intelligent controller for a continuous stirred tank reactor system using Genetic algorithm, International journal of advances in engineering and technology, 2013.

[69] Mohit Jain, Vijander Singh and Asha Rani, "Performance analysis of Multi- [70] Liu, Fang, "Nature inspired computational intelligence for financial contagion modelling", Brunel University Brunel Business School PhD Theses, 2014.

objective Genetic Algorithm tuned PID controller for Process control", ISSN: 2319-2526, Vol. 2, issue-5, 2013.

[71] Shing Chih Tsai, Sheng Yang Fu, "Genetic Algorithm based simulation optimization considering a single stochastic constraint", European of Operational Research, Elsevier, pp. 113-125, 2014.

APPENDIX

LIST OF PUBLICATIONS

[1] Ruchi Bansal and Bharat Bhushan, "**Optimum Gain settings of PID controller using Multi-objective Genetic Algorithm**", International Journal of Applied Engineering Research, ISSN 0973-4562, Vol. 9, number 6, pp. 679-686, 2014.

Research India Publications

http://www.rippublication.com

[2] Paper communicated in IEEE conference "Innovative Applications of Computational Intelligence on Power, Energy and Controls with their Impacts on humanity" titled "**Designing of Multi-objective Simulated Annealing Algorithm Tuned PID controller for a Temperature control system**".

.