RESEARCH ARTICLE OPEN ACCESS

Optimization of PID Parameter In Control System Tuning With Multi-Objective Genetic Algorithm.

Md Amanullah*, Pratibha Tiwari (Assistant Professor)**
*Department of Electrical Engineering (Control & Instrumentation), SHIATS, Deemed to be University, Allahabd-211007)
** (Department of Electrical Engineering, SHIATS , Deemed to be University, Allahabd-211007)

ABSTRACT
Way of playing advancement is the out-standing design of the study of PID control and frequently research work has been guided for this aspiration. The Proportional plus Integral plus Derivative (PID), controllers are most sweepingly used in control theory as well as industrial plants owing to their ease of execution and sturdiness way of playing. The aspiration of this deed representation capable and apace tuning approach using Genetic Algorithm (GA) to obtain the optimized criterion of the PID controller so as to acquire the essential appearance designation of the technique below meditation. The make perfect achievement about multiple plants have in relation to the established tuning approach, to consider the ability of intended approach. Mostly, the whole system's performance powerfully depends on the controller's proficiency and thus the tuning technique plays a key part in the system's behavior.
Keywords - PID controller; Optimization; Genetic Algorithm; Tuning methods;

I. INTRODUCTION

Now a day's world wide PID controller sweepingly used for an optimum solution gives a superior efficiency. For obtaining the better efficiency the absolute output needed to match set output. For this aspiration requirement of a controller. PID controller is the widely used in the process industry like petrochemical, paper, pulp, oil & gas, as well as missile control systems, because of its easy design and robust implementation in a broad range of operational condition. Unluckily, it was completely complex tune properly the gains of the PID controllers because various industrial plants are frequently loaded down with difficulties such as high order, time delays, and nonlinearities. During the long time various heuristic program procedures have been proposed during the tuning of PID controllers.

The first technique put-upon the excellent tuning rules intended by Ziegler and Nichols. In general, it is frequently complex to assign optimal or near optimal PID criterion by the Ziegler-Nichols method in various industrial plants. During these reasons, it is extremely beneficial to increase the capabilities of PID controllers by adding new features. That is also energetic for several of the plants where oscillations and overshoot is generally not In demand. This led Tyreus and Luyben to recommended new famous conventional tuning method for further conservative process loops. As for Cohen–Coon recommended a new tuning method which was based on a process reaction curve. In Kitamori also recommended a new technique about PID tuning. These conventional procedures are very Famous amid control engineers because one and only can use them, especially as no or small observation about the plant under control is available. These procedures provide stable, healthy and completely great Achievers in spite of this the gains are not at all assured of being optimal. Even, those conventional tuning process frequently breaks down to accomplish suitable Achievement in the case of plants having nonlinearity, higher order or time delay. Thus, intelligence techniques have been introduced by the researchers according to established the tuning an easier one. As for a latest scheme of PID tuning is recommended based on the Fuzzy gain programming approach. A neural networks tuned PID controller with the help of fuzzy criteria is presented.As for this paper presents a PID tuning approach founded with respect to Multi-objective Genetic Algorithm (MOGA) and his performance is matched by conventional techniques of tuning. The MOGA tuned PID (MOGA-PID) controller is well-tried on several sophisticated techniques frequently based on control system literatures. Hence, the next chapters explain by the sophisticated control techniques used in this paper for trial the performance about Multi-Objective Genetic Algorithm (MOGA) PID controller.

II. Investigation of the Plant Problems

To explain the influence about the presented technique three systems are considered. $G_1(s)$ is a certain time delay second order system, $G_2(s)$ is a third order system and $G_3(s)$ is a fourth order system. As for process control, these systems are almost generally observed, represented as:

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

Equation (1) is a second order system

$$G_2(s) = \frac{4.288}{(s+0.5)(s^2+1.64s+8.456)} \quad (2)$$

Equation (2) is a third order system

$$G_3(s) = \frac{27}{(s+1)(s+3)^3} \quad (3)$$

Equation (3) is a fourth order system

Due to the time delay definition with respect to process $G_1(s)$, a second order pade approximation. The pade approximation is authentic only at low frequencies, and accommodates better frequency-domain approximation than a time domain approximation.

III. About Genetic Algorithm:-

John Holland was the father of genetic algorithm who discovered in early 1970. Genetic algorithms (Gas) are adaptive heuristic search algorithm based on the evolutionary ideas of natural expansion. As such the denote and intellectual victimization of a random search used to optimization problems. Although randomized, GAs are absolutely not in random, instead the deed of historical knowledge to direct the search into the reason of excellent performance in the search space. The key technique of the GAs is designed to fabricate processes in general system mandatory for evolution, mainly those follow on the ethics first laid down by Charles Darwin of “survival of the fittest.” GA has been studied as a satisfactory and effective method for finding out difficult optimization problems. By exactly avoiding local minima, it converges to global minima. It originates from an initial population obtaining a number of chromosomes where for each one correlate to a result of the given problem. Then the achievements of each original are calculated by using a correct fitness function. Essentially, GA consists of five significant steps: initial population, fitness function, Selection, Crossover and Mutation. These are also known as GA drivers. The application of five basic operators confesses the formation of new children, which may be located excellent than their parents. This algorithm is repeated for several generations and finally stops when meeting product that is denoted the optimum result of the problem.

(a) Initial population:-

Its starts with Randomly Originated states, these states are satisfactory to the problem. The population size of its create on the nature of the problem, even so typically consist of various hundreds or thousands of feasible solutions. Commonly, the population is originated randomly; admit the perfect range of possible solutions. Infrequently, the solutions can be "sown" in the range, where optimal solutions are possible to be established.

(b) Fitness function:-

A fitness function is a certain type of objective function that is familiar with summarize, because a single figure of merit, however close to a given design solution is to complete the set of goals. In the range of genetic programming and genetic algorithms, a single design solution is denoted by a string of numbers (specified as a chromosome). After every overall testing or simulation, the concept is to remove the 'n' worst design solutions, and to create 'n' new ones as from the best design solutions. Each and every design solution have to be rewarded a figure of merit, to illustrate how close it comes to meet the overall requirement, and it is developed by put into use the fitness function to the test or simulation, solutions are obtained from that solution.

Two main parts of fitness functions survive: the one where the fitness function does not change while optimizing a fixed function or testing with a fixed set of test cases and another one where the fitness function is changeable, while niche separation or develop the set of test cases.

(c) Selection:-

Two pairs are selected at random to reproduce. They are selected based on their fitness function score. One may be selected more than one, whereas one may not be selected at all. Make a copy the selected programs to the new population. The regeneration process may be subdivided into two parts, first is Fitness Evaluation and second is Selection. The fitness function is how is operated the evolutionary process and its view is to identify how well a string (particular) solves the problem, admitting as an evaluation of the respective performance of each population member. Basically four most common methods of the selection:

1. Tournament Selection
2. Normalized geometric selection
3. Roulette Wheel selection
4. Stochastic Universal is sampling

(d) Crossover:-

For each pair to be mated, a crossover point is preferred at random from within the bit string. The offspring is developed by interchange between the parents at the crossover point. Population is different early in the process, these reasons the crossover to be large in the beginning. However, it will settle down in future generations. Hence, There are several types of crossover operators like single point crossover, two point crossover, arithmetic crossover etc.

(e) Mutation:-

Mutation is a genetic operator familiar with a cure for genetic difference from one generation of a population about genetic algorithm chromosomes to the next. It is related to biological mutation. Mutation change more than one gene values in a chromosome from its initial state. In mutation, the solution may completely change from the last solution. Hence, GA may come to a better solution by using mutation. Mutation occurs during evolution by a user-definable probability. This has probably been set low. If it is set too high, the search will become an original random search.

Genetic Algorithm performs the following steps:

1. Originate an initial population, randomly or heuristically.
2. Calculate and save the fitness for each particular in the current population.
3. Specify the selection probability at single that it is reciprocal to its fitness
4. Originate the next current population by most probabilities selecting the individuals from the earliest current population, in order to goods product of genetic operators.
5. Repeat step 2 until a sufficient result is obtained.

A flow chart of the general scheme of the implementation of the GA is shown in Figure 1.

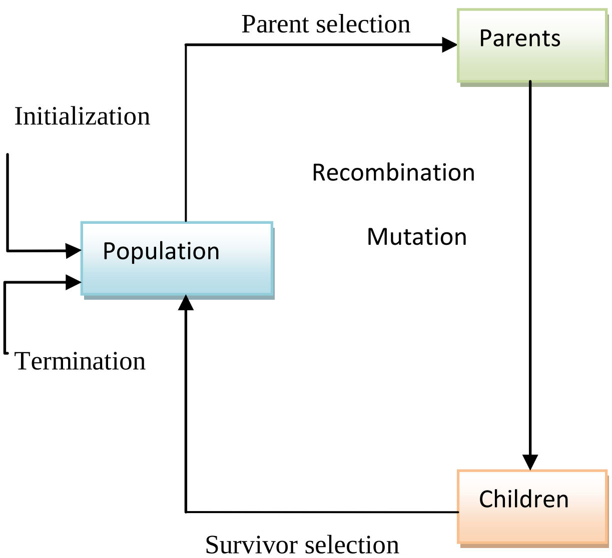

Figure 1:- flow chart of the general scheme of the implementation of the Genetic Algorithm.

IV. Fundamental of PID control Action:-

The PID controller has been brodly used since it invented in 1910. The combination of proportional control action, integral control action and Derivative control Action is Termed proportional plus Integral plus derivative control Action. It improves both the transient and steady state response characteristics. It is similar to lead lag compensator or band reject filter, it reduces the rise time. It increases bandwidth and also increases stability of the system. The peak overshoot depends on properly tuned values of T_i and T_d. It eliminates the steady state error between input and output. It increases the TYPE and ORDER by the system is One. The transfer function of a PID controller as given below.

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \quad (4)$$

Where, K_p = proportional gain, K_i = Integral gain, K_d = Derivative gain.

We can specify another Equivalent Form of the PID control has the following form.

$$G_c(s) = K_p[1 + 1/(T_i s) + T_d s] \quad (5)$$

Where, $T_i = \frac{K_p}{K_i}$ and $T_d = \frac{K_d}{K_p}$; T_i and T_d are called as Integral and Derivative time constants, respectively. The basic block diagram plant controlled by PID controller is shown in given below.

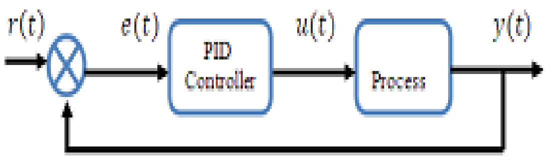

Figure 2:- Block diagram of the simplest closed loop system

Where, $r(t)$ = set point, $e(t)$ = error signal, $u(t)$ = controller output, $y(t)$ = plant output.

Typical structure:-

The typical operation of the PID control is shown in Figure 3. The signal error, $e(t)$, enters the PID control block and the resulting action signal is the sum of the error signal modified by the proportional, integral and derivative actions.

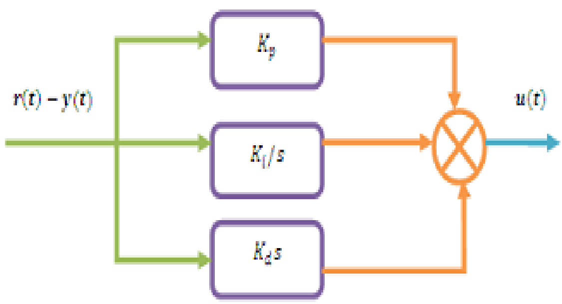

Figure 3:- Block Diagram of Typical PID control Scheme.

$e(t) = r(t) - y(t)$ is called as error signal and

$u(t)$ is the controller output for this specific error signal.

$$u(t) = K_p e(t) + K_i \int e(t) d(t) + K_d \frac{d(t)}{dt} e(t) \quad (6)$$

Also, the equation (6) can be rewritten in Laplace form as:

$$u(s) = e(s)(K_p + K_i/s + K_d s) \quad (7)$$

Finally, under the above strategy the transfer function of the PID controller or the control law is established by

$$C(s) = \frac{u(s)}{e(s)} = K_p + \frac{K_i}{s} + K_d s \quad (8)$$

Tuning of PID by MOGA:-

The fitness criterion is a mathematical performance of the problem's high level demand. That is, our fitness criterion tries to optimize for the integral of the square error (ISE) for these step input and also to optimize for maximum awareness. The fitness function is a more significant consideration while tuning the PID controller by MOGA. Several multi-objective fitness functions are considered in this study represented as follows:

$$f_1 = 0.05ISE + 0.05t_s + 0.90OS \quad (9)$$

$$f_2 = 0.10ISE + 0.70t_r + 0.20t_s \quad (10)$$

$$f_3 = 0.10ISE + 0.20t_r + 0.70t_s \quad (11)$$

$$f_4 = 0.10ISE + 0.80OS + 0.10t_s \quad (12)$$

$$f_5 = 0.10ISE + 0.60t_r + 0.10OS + 0.20t_s \quad (13)$$

Where, t_s is the settling time within 5 percent, t_r is rise time, *OS* is percentage overshoot and *ISE* is Integral of the square error which can be defined as follows:

Commonly, the PID controller scheme method using the integrated absolute error (IAE), integral of squared-error (ISE), integrated of time-weighted-squared-error (ITSE*)* is frequently occupied with control system design on account of it can be classified experimentally in the frequency domain.

$$IAE = \int_0^t r(t) - y(t) dt = \int_0^t e(t)\, d(t) \quad (14)$$

$$ISE = \int_0^t e^2(t) d(t) \quad (15)$$

$$ISTE = \int_0^t te^2(t) d(t) \quad (16)$$

It is the preferred of the control engineer so that which individual criterion of the control system requirement additional consideration. So as per the demand a higher weight can be allowed, although given the other requirement specification at the same time. Yet, the total sum of the weights in an objective function must be equal to one, in order to the total performance of the system may be confirmed. It shows the resilience in PID tuning while applying MOGA.

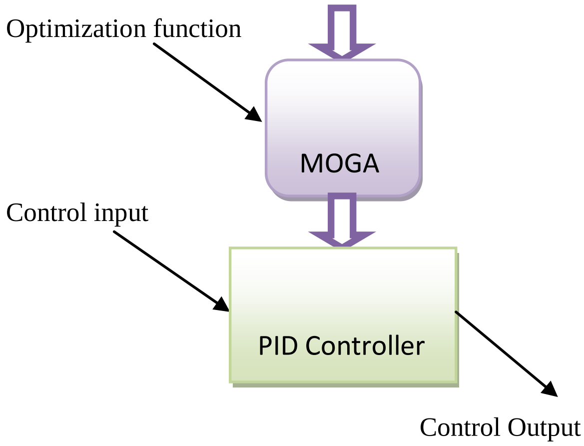

Figure 4:- MOGA based PID Tuning Scheme.

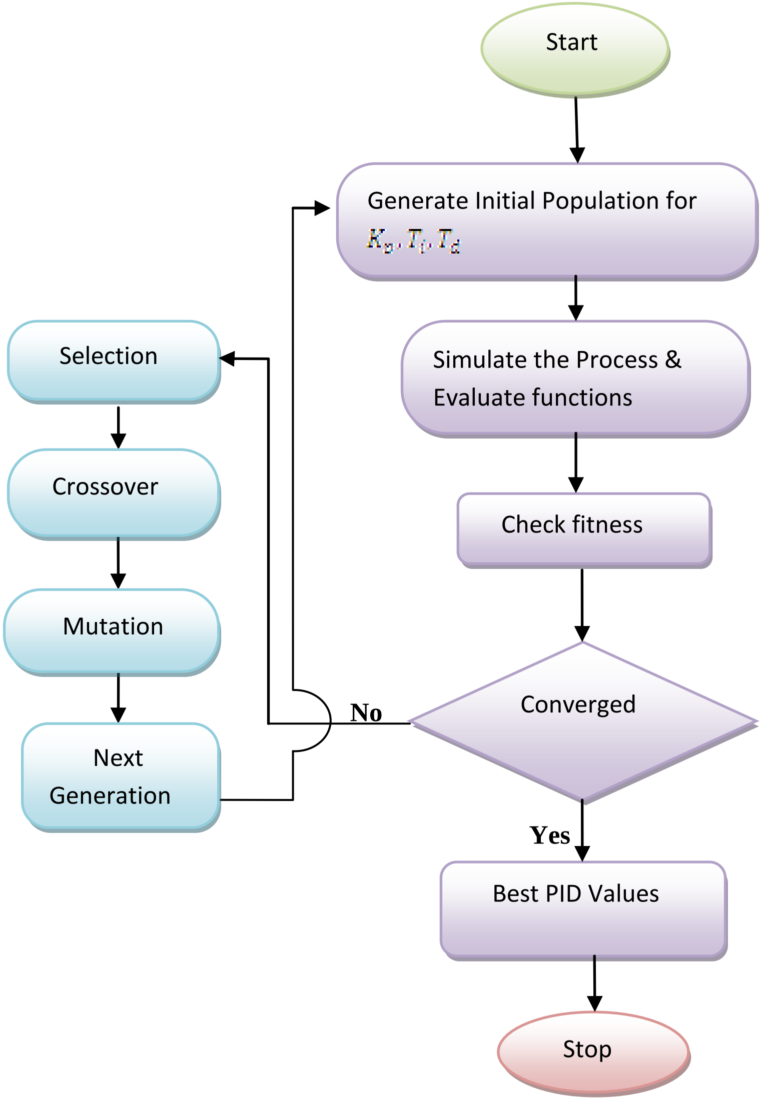

Figure 5:- Flow chart of MOGA-PID Controller.

Eliminate the negative error components by the help of ISE. $ISE = \sum_{k=1}^{q} e^2(k)$ In this simulation, the objective is to decrease the cost function. During that motive the objective function is preferred for the Integral Square Error (ISE). The ISE squares the error to eliminate negative error components. Multi Objective Genetic Algorithm (MOGA) PID Controller performance depends on the convergence rate.the criteria like population type, population size, creation, function, selection and much more are also affecting the convergence rate. Hence, its order to decline precocious convergence of Genetic Algorithm, so this extremely strategies to select the suitable operators and criteria for it. Table I display the criteria and operators of GA that are accomplish by accurate experimental research of this work.

GA Parameter	Value/Method
Population type	Double Vector
Population size	20
Creation function	Feasible population
Selection	Tournament
Mutation	Adaptive feasible
Crossover	Arithmetic crossover
Generation	65

Accepting above two stopping criterion have been used in this paper, the first one is the number of generations and the second one is function tolerance which is 1×10^{-6}.

V. Simulation and Result

In order to that the authenticate ability of the optimization strategy, a simulation criterion test is implemented in MATLAB Simulink software. The results are MOGA based PID tuning and Compared with Z-N, Kitamori, T-L have been well-tried on a different types of plants which are generally initiated in the process control system. Entire these simulations are performed on a laptop having an Intel® Core (TM) 2Duo CPU processor operating at 2.20GHZ, 32 bit RAM and Installed with MATLAB® 7.10.0 (R2010a). Figure 6,7,8 shows the step response of the process $G_1(s), G_2(s), G_3(s)$ respectively controlled by several conventional and MOGA- PID controller.

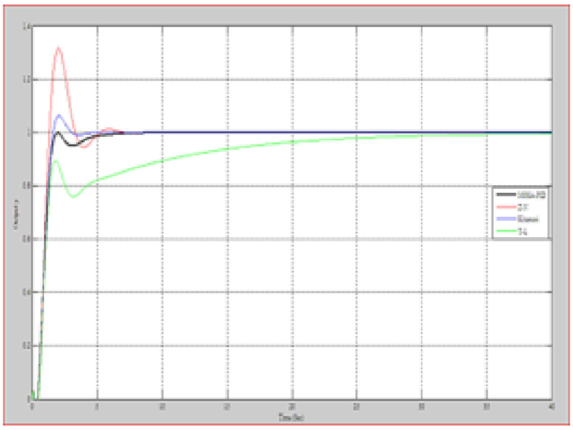

Figure 6: - shows the Step response of the process G_1(S)

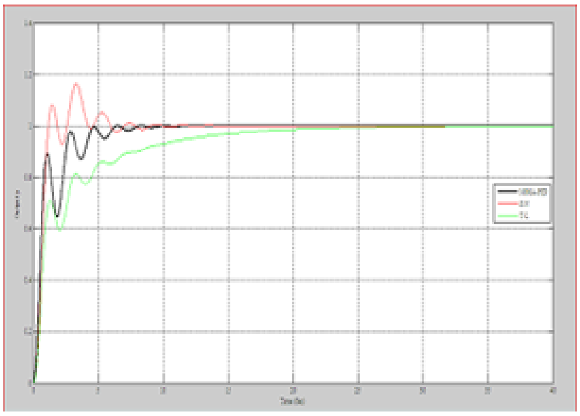

Figure 7:- shows the Step response of the process G_2(S)

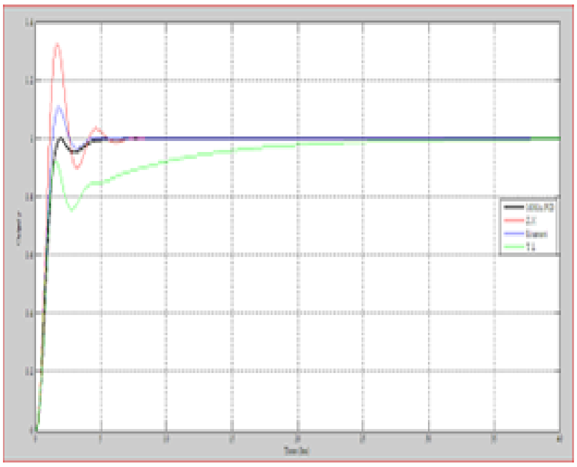

Figure 8:- shows the Step response of the process G_3(S)

Table 2:-Overview of Relative Study

Process	Index	Kitamori	Z-N	T-L	MOGA-PID
G_1	K_p	2.211	2.807	2.1327	2.1245
	T_i	2.039	1.65	7.195	2.3328
	T_d	0.518	0.41	0.518	0.5556
	$\%OS$	6.7	32	0	0
	$t_s \pm 5$	2.38	4.17	16.04	1.58
G_2	K_p	2.211	2.17	1.658	2.3504
	T_i	2.039	1.04	4.531	2.6757
	T_d	0.518	0.257	0.3269	0.4470
	$\%OS$	6.7	18	0	0
	$t_s \pm 5$	2.38	5.46	12.01	4.22
G_3	K_p	2.356	3.071	2.3271	1.9826
	T_i	1.648	1.352	5.9489	1.8418
	T_d	0.415	0.339	0.4291	0.4532
	$\%OS$	11	32.9	0	8.7×10^{-14}
	$t_s \pm 5$	2.2	3.721	13.08	1.51

Table 2: Method for tuning applications. Denoted the overall performance study of the MOGA-PID controller and several conventional PID controllers. In this relative study of the results achieved by Kitamori's and Ziegler-Nichols PID controller. The parameters of Tyreus and Luyben PID controller have been determined by $K_p=K_u/2.1, T_i=2.2P_u$ and $T_d=P_u/6.2$ where K_u is the ultimate gain and P_u is the ultimate period of sustained oscillation. While tuning of PID by MOGA, f_1The fitness function is considered. THE behavior of MOGA-PID controller is better than the conventionally tuned PID controller. Fig. 9 shows the correlation of step response of the system $G_1(s)$ controlled with MOGA-PID controller in which various fitness functions have been exploited for the objective of tuning.

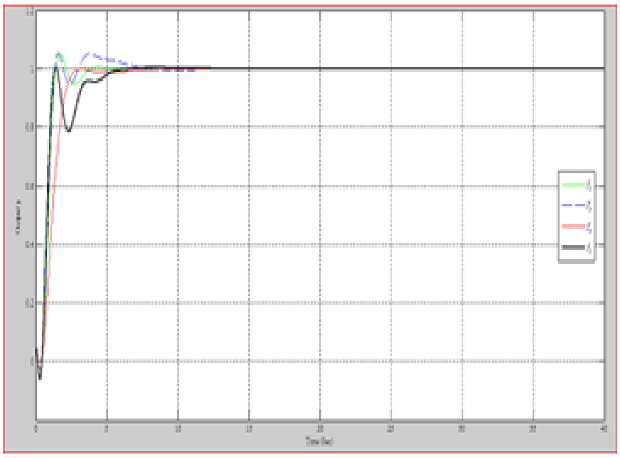

Figure 9:-MOGA-PID step response for several Objective function

Table 3:-Comparison of MOGA-PID for several objective function

Process	Index	f_2	f_3	f_4	f_5
G_1	K_p	2.3347	2.2168	1.6374	2.1067
	K_i	1.1160	1.3538	0.7779	0.8931
	K_d	1.4531	1.6871	0.7454	1.8781
	%OS	5.22	5.25	0	0.4274
	$t_s \pm 5\%$	1.31	1.21	2.23	3.31
	t_r	0.67	0.61	1.39	0.58
	Objective function	0.8157	1.0515	0.3211	1.1391

Hence, the exact surveillance of the table 3 fair denoted by the MOGA-PID controller gives the information to the control engineer tuning in demanding needs. Here, suitable weight is appointed to a particular specification in this case target is degrade the overshoot it is clear from the Table 3.

VI. Conclusion

The Designed with MOGA-PID controller has much faster response than the conventional tuned PID controller.Result showing that the achievement of MOGA based PID controller is much better.it is a effective tuning strategy for PID controllers. The great utility of using MOGA is that it is absolutely autonomous from the complicated nature of the objective function under consideration. The MOGA designed PID is much better in terms of the rise time and settling time than the Z-N Technique.in this manner when tuning a PID controller, the controlling process can be considered several objectives slightly cramped to the single objective for the conventional tuning methods.

References

[1] K. J. Astrom and T. Hangglund, "Automatic tuning of PID Controllers," Instrument Society of America, 1988.
[2] J. G. Ziegler and N. B. Nichols, "Optimum setting for Automatic Controllers," ASME Trans., vol. 64, pp. 759-768, 1942.
[3] Tyreus, B.D. and W.L. Luyben, "Tuning PI controllers for integrator/dead time processes," Ind. Eng. Chem. Res. pp. 2628–2631, 1992.
[4] G. H. Cohen and G. A. Coon, "Theoretical investigation of retarded control," ASME Trans., vol. 75, pp. 827-834, 1953.
[5] T. Kitamori, "A method of control system design based upon partial knowledge about controlled processes," Trans. SICE Japan, vol. 15, pp. 549- 555, 1979.
[6] A. O Dwyer, Handbook of PI and PID Controller Tuning Rules. Imperial College Press, 2009.
[7] Z.-Y. Zhao, M. Tomizuka, and S. Isaka, "Fuzzy gain scheduling of PID controllers," IEEE Trans. on Systems, Man and Cybernetics, vol. 23, no. 5. pp. 1392–1398, 1993.
[8] 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.
[9] R. L. Haupt, S. E. Haupt, Practical Genetic Algorithms, Wiley, 2004.
[10] S. Sumathi, T. Hamsapriya, and P. Surekha, Evolutionary Intelligence. Springer-Verlag Berlin Heidelberg, 2008.
[11] S. N. Sivanandam and S. N. Deepa, Introduction to Genetic Algorithms. Springer-Verlag Berlin Heidelberg, 2007.339999
[12] G. Stephanopoulos, Chemical process control: an introduction to theory and practice. Prentice-Hall, 1984.
[13] F. Rothlauf, Representations for Genetic And Evolutionary Algorithms. Springer-Verlag Berlin/Heidelberg, 2006.
[14] B. W. Bequette, Process Control: Modeling, Design, and Simulation. Prentice Hall PTR, 2003.
[15] T. Jain and M.J. Nigam, Optimisation of PD-PI Controller Using Swarm Intelligence. Journal of Theoretical and Applied Information Technology, 4(11): p. 1013-1018, 2008.
[16] P. B. De Moura Oliveira, Modern Heuristics Reviewfor PID Control Optimization: a Teaching Experiment. in Proceeding of International Conference on Control and Automation. 2005. Budapest, Hungary.
[17] C. R. Reeves, Modern Heuristic Techniques for Combinatorial Problems. 1995, London: McGraw- Hill.
[18] M. Vanitha and K. Thanushkodi, Solution to Economic Dispatch Problem by Differential Evolution Algorithm Considering Linear Equality and Inequality Constrains. International Journal of Research and Reviews in Electrical and Computer Engineering. 1(1): p. 21-26, 2011.
[19] R. Storn and K. Price, Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. Journal of Global Optimization. 11(4): p. 341–359, 1997.
[20] D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning. 1989: Addison-Wesley Publishing Co. Inc.