International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
International Conference & workshop on Advanced Computing 2013 (ICWAC 2013) – www.ijais.org

PID Tuning using Elite Multi-Parent Crossover Genetic Algorithm

Reshmi P. Pillai
Ramrao Adik Institute of Technology
Nerul
Navi Mumbai

Sharad Jadhav
Ramrao Adik Institute of Technology
Nerul
Navi Mumbai

M.D.Patil
Ramrao Adik Institute of Technology
Nerul
Navi Mumbai

ABSTRACT
Proportional-Integral-Derivative (PID) controllers have been widely used in process industry for decades from small industry to high technology industry.But they still remainpoorly tuned by use of conventional tuning methods. Conventional technique like Zeigler-Niclos method does not give an optimized value for PID controller parameters.In this paper we optimize the PID controller parameter using Genetic Algorithm(GA), which isa stochastic global search method that replicates the process of evolution. Using genetic algorithms to perform the tuning of the controller will result in theoptimum controller being evaluated for the system every time. The GA is basicallybased on an iterative process of selection, recombination, mutation and evaluation. Multi-parent Crossover Algorithm with Discrete Recombination is implemented in this paper along with recommendation for further work. This algorithm uses different replacement strategy as compared to Elite Multi-Parent Crossover Evolutionary Optimization Algorithm (EMPCOA) therby increasing population diversity thus reducing the number of iterations required. Elitism is also known to increase speed and ensures the good solution once found is passed on to the next generation.

Keywords
PID tuning; Genetic Algorithm; Multi-parent crossover; Elite crossover; Discrete recombination.

1. INTRODUCTION
The PID controller was patented in 1939 by Albert Callender and Allan Stevenson of Imperial Chemical Limited of Northwich, England. The PID controller is widely used in most industrial processes despite continuous advances in control theory. The main reason is due to their simplicity of operation, ease of design, inexpensive maintenance, low cost, and effectiveness for most linear systems. Recently, motivated by the rapidly developed advanced microelectronics and digital processors, conventional PID controllers have gone through a technological evolution, from pneumatic controllers via analog electronics to microprocessors via digital circuits [1]. Most conventional PID tuning methods require considerable technical experience to apply tuning formulas to determine the PID controller parameters. The conventionaltuning methods require the process model to be reduced if it istoo complicated originally [2]. In practical applications, most of the industrial process exist to be non-linear, variability of parametersand uncertainty of model are very high, thus using conventional PID tuning methods theprecise control of the process cannot be achieved. Due to this, PID controllersare rarely tuned optimally and thus required improved tuning technology. The above problemscan be well addressed by the application of non-conventional methods for tuning of the PIDcontroller.Most practical PID remains poorly tuned leading to deteriorated process performance [3]. The conventional tuning methods require considerable technical experience and are time consuming and do not work well for non-linear, higher order and time-delayed systems and the ones that do not have a precise mathematical model [4]. Non-conventional methods are especially useful for solving problems ofcomputationally complicated and mathematically untraceable.Hence the need arises for an optimization algorithm like Genetic Algorithm (GA).

Genetic Algorithm is a stochastic search and optimization method that mimics theprocess of natural *evolution* [5]. John H. Holland formally introduced GA in his book,*Adaptation in Natural and Artificial Systems* in the 1975 at the University of Michigan, AnnArbor, United States. GA is one of the *Evolutionary Algorithms* methodologies.The key aspectdistinguishing an evolutionary search algorithm from such traditional algorithms is that it ispopulation-based. Through the adaptation of successive generations of a large number ofindividuals, EA performs an efficient directed search. Evolutionary search is generally betterthan random search as EA inspired by the evolution process in nature and try to solveproblems by evolving sets of search points. GA imitates natural evolution with survival of the fittest approach. It performs on coding of parameters hence does not depends on the continuity of parameter nor the existence of the derivatives of the functions, thus allowing it to handle multi parameters or multi-model type of optimization problems. GA can also work for non-deterministic systems or the systems that can be only partially modeled. GA uses random choice and probabilistic decision to guide the search, where the population improves toward near optimal points from generation to generation [6].The main advantage of the GA formulation is that fairly accurateresults may be obtained using a very simple algorithm. The GA is basically based on aniterative process of selection, recombination, mutation and evaluation. GA has parallel search techniques, which emulate natural genetic operations. Due to its high potential foroptimization, GA has received great attention in control systems such as the search of optimal PID controller parameters.

Organization of this paper is as follows, section 2. Gives basic idea of a PID controller, 3.Introduces Genetic Algorithm, 4.Explains the steps of the GA being implemented, 5. Explains the design procedure and section 6.presents the results and 7. Presents the conclusion and future scope .

2. PID CONTROLLER

A PID controller aims at minimizing the error between a measured processvariable of the controlled system and a reference, by calculating the error and generating acorrection signal to the system from the error.The block diagram of a conventional PID controller is shown in the Fig(1), where $r(t)$ is the reference value, $Y(t)$ is the output of the controlled system, $e(t)$ is the error between $r(t)$and $Y(t)$, whereas $u(t)$ is the output control signal of the PID controller.A conventional PID controller consists of threecomponents: the proportional part, the integral part and the derivative part as shown in Fig (1).The proportional term produces an output value that is proportional to the current error value.The contribution from the integral term is proportional to both the magnitude of the error and the duration of the error.Derivative control is used to reduce the magnitude of the overshoot produced by the integral component and improve the combined controller-process stability.The output control signal of a PID controller is described as follows,

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \qquad (1)$$

where, u(t) is the output control signal, e(t) is an error signal, and K_p, K_i, and K_d refersto the proportional gain, the integral gain and the derivative gain, respectively.
K_p, K_i, and K_d should satisfy following equations,

$$K_i = K_p T_i \qquad (2)$$

$$K_d = K_p T_d \qquad (3)$$

where, T_i and T_d refers to the integration time and derivative time, respectively.

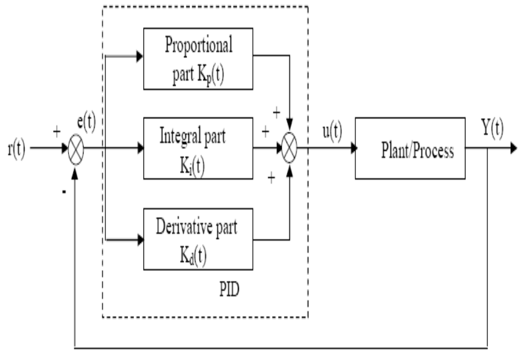

Fig 1. PID Controller Block Diagram

The individual effects of these three terms on the closed-loop performance are summarized in Table (1). Note thatthis table serves as a first guide for stable open-loop plants only. For optimum performanceK_p, K_i and K_d are mutuallydependent in tuning.

Table 1. Effects of Independent P.I and D tuning [7]

Closed loop response	Rise Time	Over-shoot	Settling Time	Steady State Error	Stability
Increase K_p	Decrease	Increase	Small Increase	Decrease	Degrade
Increase K_i	Small Decrease	Increase	Increase	Large Decrease	Degrade
Increase K_d	Small Decrease	Decrease	Decrease	Minor change	Improve

The quality of PID tuning rules is of considerable practical importance because asmall percentage improvement in the operation of a plant can translate into large economicsavings or other benefits.

3. GENETIC ALGORITHM

GA is a probabilistic optimization algorithm with a high probability of finding a good solution in a given search space. *Genetic Algorithm* can handle multiple variables and only requires the ability todevelop a mathematical model to configure a set of inputs (the variables) in order for the model to produce an optimal output.After initialization of population, each string (individual) in the population isevaluated to determine the performance of the string.Then, the higher-ranking strings aremate. The process of crossover is performed by combining strings containing partial solutions. The algorithm favors fittest strings as parents, thus better strings will have more number of offspring.The GA exploits the regions of the solution space, because successive generations ofreproduction and crossover produce increasing numbers of strings in those regions. In this paper the offspring replaces the weakest spring, thus maintaining the population size same [8].Lastly, mutations modify a small fraction of the strings. Mutation alone does notgenerally advance the search for a solution, but it does provide insurance against thedevelopment of a uniform population incapable of further evolution [9].

Guo Tao’s Algorithm (GTA) is a linear non-convex multi-parent crossover operator (GTX)which is used in optimization of nonlinear continuous functions [10]. The multi-parent crossover utilizes more number of candidate solutions and the replacement strategy implemented is supposed to be minimizing selection pressure. But major limitation of GTA is that it may ignore better solutions in population. Tomake use of better solutions in population elite preservation strategy is introduced by XiaoyiChe, Youxin Luo and Zhaoguo Chen implemented in the elite multi-parent crossoverevolutionary optimization algorithm (EMPCOA) [11]. The selection scheme and replacementstrategy implemented in EMPCOA gives global optima with increase in an execution time.This is mainly due to the decrease in population diversity and therefore requires more numberof iterations to converge.

Aimed at these shortages of GTA and EMPCOA we are motivated to implement the Multi-parent Crossover Algorithm with DiscreteRecombination [8] with better parts of both algorithms like, fixed population size of GTA andelite preservation strategy of EMPCOA with multi-parent crossover. Here, we aim to reducethe number of iterations and execution time with improvement in transient (performance)response.

General steps involved in GA are i. Representation, ii. Objective function, iii.Population initialization, iv. Parent selection mechanism, v. Variation operator, crossover (recombination), vi. Variation operator, mutation and vii.Termination condition.

3.1 Work Flow of GA

A GA is typically initialized with a randomly generated population consisting of candidate individuals. Each individual in the population is usually represented by a real-valued number or a binary string. Such strings are called as chromosomes. A set of chromosome or individual is a whole population. Performance of each individual is measured and assessed by the objective function. The objective function assigns each individual a corresponding number called its fitness value. If the termination criteria are not met with the current population then a new individuals are created withgenetic operators. A survival of the fittest strategy is appliedon individuals. The fittest parents are found out by reproduction or selection operator. Newindividuals are generated by performing operations such as crossover and mutation on theindividuals whose fitness has just been measured. The fitness of the offspring is thencomputed. The offsprings are inserted into the population replacing the parents or low-fitnessindividuals producing a new generation. This cycle is performed until the termination criterionis reached. Such a single population GA is powerful and performs well on a wide variety ofproblem. Every iteration of GA loop is referred to as a generation. When termination criteriagets satisfied GA stops.

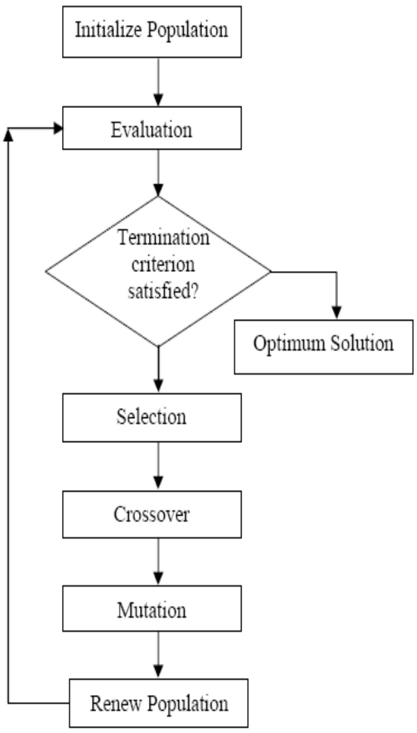

Fig 2. Work flow of Genetic Algorithm

3.2 Scope of work

- Implementation of the Multi-parent Crossover Algorithm to obtain optimal PID parameters
- To compare the performance with existing GAs like EMPCOA.
- Performance analysis with different error criteria.
- Performance comparison in terms of population size.
- Comparison of control effort with different error criteria.
- Performance evaluation based on termination criteria.

4. PID TUNING USING GA METHOD

The GA being implemented is Multi-parent Crossover Algorithm with DiscreteRecombination [8].The main features of the algorithm implemented are i. It uses elite preservation strategy, ii. It makes use of multi-parent crossover to create new offspring, iii. The performance of the proposed algorithm is better than the existing algorithms like GTA and EMPCO in terms of number of iterations and computational time and iv. It gives better transient response as compared to the existing algorithms like GTA andEMPCOA.

The steps of the algorithm are

Step 1 Produce initial population $P_0 = X_1;X_2:::X_N$randomly at searching space S, N isnumber of individuals in populations and t = 0;

Step 2 Arrange the individuals in population P from good to bad according to the fitnessof parent. Then still record as P_t=X_1,X_2...X_N after the arrangement, X_1 is thebest individual X_N is the worst one;

Step 3 Termination criteria is when the fitness difference between X_{worst} and X_{best}become less than or equal to fitness limit (ε) (i.e. the fitness of the worst individualis almost same as the best one), then go to step 7;

Step 4 Choose K (K≤m) best individuals X_1,X_2...,X_k from population P_t, and then, choose (m-K) individuals X_{k+1}, X_{k+2}...,X_m from the rest (N-K) individualsrandomly. A subspace V is formed from these m (m≤N) individuals. Then perform multi-parent crossover as given in (4);

$$V = x_c \in S, x_c = \sum_{i=1}^{m} a_i x_i \tag{4}$$

$$\sum_{i=1}^{m} a_i = 1, -0.5 \le a_i \le 1.5 \tag{5}$$

Step 5 Compare x_c with X_{worst};x_c replaces X_{worst}if better (x_c,X_{worst}) condition is true else discard x_c;

Step 6 Go to Step 2

Step 7 Output is the best solution and end.

Here we implement the algorithm for optimizing PID parameters. The following section describes how gains K_p, K_i and K_d are represented in form of chromosome or individual. The implementation of PID parameters optimization procedure using GA starts with the chromosome representation.

K_p	K_i	K_d

Fig 3. Chromosome representation

As illustrated in Figure (3), the chromosome is formed by three values that correspond to the three gains to be adjusted in order to achieve a satisfactory behavior. The gains K_p, K_i and K_d are strings of chromosome as shown in Figure (3).A set of chromosomes or individual forms a generation. Block diagram for optimization of PID parameters using GA is presented in Fig 4.

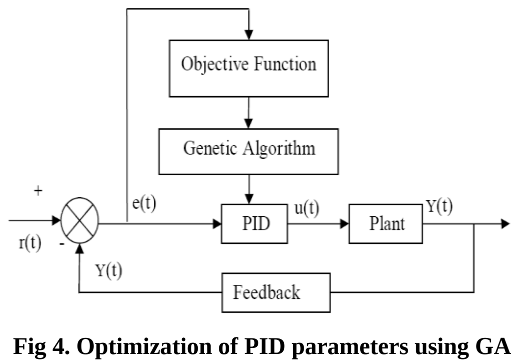

Fig 4. Optimization of PID parameters using GA

An objective function could be created to find a PID controller that gives the smallest overshoot, fastest rise time or quickest settling time. There are several variables used as the standard to measure systems performance. In general, unit step input is used to test the systems, and the output signals is characterized by some standard performance measures likes settling time, percent overshoot, rise time and peak time. All these measures are defined in the time domain response. Each chromosome in the population is passed into the objective function one at a time. The chromosome is then evaluated and assigned a number to represent its fitness, which is its fitness value. The GA uses the chromosomes fitness value to renew population consisting of the fittest members. When the chromosome enters the evaluation function, it is split up into its three terms. The newly formed PID controller is placed in a unity feedback loop with the system transfer function. This will result in reducing the compilation time of the program. The system transfer function is defined in another file and imported as a global variable. The controlled system is then given a step input and the error can be assessed using error performance criterion integral absolute error (IAE).

5. DESIGN PROCEDURE

- The initial population of 10 chromosomes is generated using random number generation.
- The objective function is used as error performance criterion. The error criteria used is Integral of absolute error (IAE). The magnitude of this error will be used to assess the fitness of each chromosome.
- The termination criteria for the algorithm is based on fitness limit. The algorithm terminates when the difference between the fitness of best solution X_{best} and worst solution X_{worst} is less than or equal to fitness limit $\varepsilon = 1$.
- A subspace is formed using elite preservation strategy and then multi-parent crossover takes place using 5 parents. The 2 best chromosomes are selected and the 3 random parents are selected from the remaining 8 chromosomes.
- In the proposed algorithm the offspring solution (x_c) is compared and replaced with X_{worst}.

6. RESULT

We implement the proposed algorithm on a system with transfer function given as follows.

$$G(s) = \frac{25}{s^2 + 6s + 25} \qquad (3)$$

The step response of the closed loop system is shown in Figure (5). Weobserved the transient performance of the plant for the set of optimum gains of PID (i.e. K_p, $K_i$and K_d).

From the graph shown in Fig 5 we observe the following,

Parameters	Open loop	Closed loop
Rise Time	0.371s	0.0899s
Settling time	1.19s	1.48s
Peak Time	0.787s	0.189s
Peak Amplitude	1.09	1.15

The proposed algorithm takes approximately 146.287960 sec. and 30iterations to reach the optimum value. Optimized PID parameters obtained throughthe proposed algorithm are, miminum value of function zn_best= -1.9388, K_p = 8.4913, K_i = 9.3399 and K_d = 0.3571.

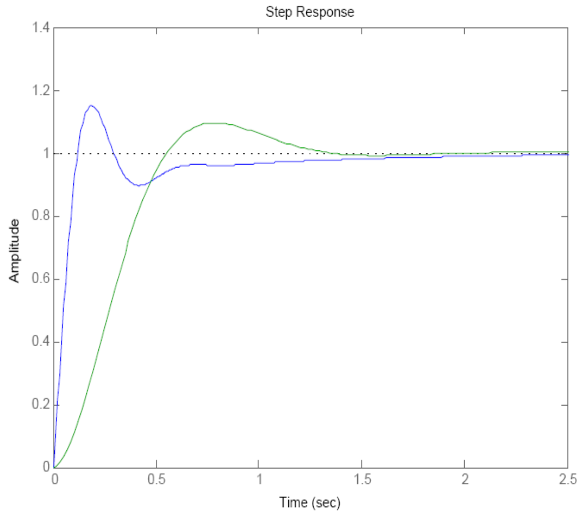

Fig 5.Step response of the open loop and closed loop system with the proposed algorithm for the illustrative example

7. CONCLUSION

The proposed algorithm maintains population diversity which in turn reduces the number of iterations, hence reduces the execution time and gives better transient response. The proposed algorithm works better than EMPCOA in terms of transient response, number of iterations and execution time. This is mainly due to the change in replacement strategy. Further study of this topic could include comparison with the existing EMPCOA, performance analysis with different error criteria, performance comparison in terms of population size and termination criteria and comparison of control effort with different error criteria.

8. ACKNOWLEDGMENTS

It's with great pleasure that I express my gratitude to my guide Prof.S.P.Jadhav and Prof. Mukesh D. Patil for their guidance and inspiration. They helped me work in right direction to attain desired objective. I thank all those who have contributed directly or indirectly to this project.

9. REFERENCES

[1] K. S. Tang,Kim Fung Man, Guanrong Chen and Sam Kwong, “An Optimal Fuzzy PID Controller”, IEEE Transactions On Industrial Electronics, vol 48, no 4, pp. 757-765, 2001

[2] Ho W. K., Gan O. P., Tay E. B. and Ang E. L., “Performance and gain and phase margins of well-known PID tuning formulas”, IEEE Transactions on Control Systems Technology, vol. 4, no. 4, pp. 473–477, 1996.

[3] Aidan O'Dwyer. 2006 PI and PID controller tuning rules: an overview and personal perspective. Proceedings of the {IET} Irish Signals and Systems Conference.

[4] K. S. Tang,Kim Fung Man, Guanrong Chen and Sam Kwong. 2001An Optimal Fuzzy PID Controller. IEEE Transactions On Industrial Electronics.

[5] Holland J.H. , "Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence", Ann Arbor, MI: University of Michigan Press 1975, Second edition, Cambridge, MA: The MIT Press, 1992.

[6] Teo Lian Seng, Marzuki Bin Khalid and Rubiyah Yusof. 1999. Tuning of a Neuro-Fuzzy. IEEE} Transactions On Systems, Man, And Cybernetics—Part B: Cybernetics

[7] Kiam Heong Ang, Gregory Chong and Yun Li, "PID Control System Analysis, Design and Technology", IEEE Transactions On Control Systems Technology, vol 13, no 4, pp. 555-576, 2005.

[8] Dazhi Jiang, Yulin Du, Jiali Lin, Zhijian Wu. 2010Multi-parent Crossover Algorithm with Discrete Recombination. IEEE.

[9] R. Michalski, J. Carbonell and T. Mitchell, "Machine Learning: An Artificial Intelligence Approach", Los Altos, Morgan Kaufmann, 1986.

[10] Guo Tao and Kang Li-Shan, "A new Evolutionary Algorithm for Function Optimization", Wuhan University Journal of Natural Sciences, vol.4, no.4, pp.409- 414, 1999.

[11] Xiaoyi Che, Youxin Luo and Zhaoguo Chen, "Optimization for PID control parameters on hydraulic servo control system based on the elite multi-parent crossover evolutionary algorithm" , International Conference on Measuring Technology and Mechatronics Automation, IEEE Computer Society, pp. 845- 848, 2010.