Design of GA based PID Controller for three tank system with various Performance Indices

P Srinivas, K Vijaya Lakshmi, V Naveen Kumar

Abstract: Proportional–integral-derivative (PID) controllers tuning is a challenging aspect for researchers and process operators. This paper proposes PID controller tuning of a three tank level process using genetic algorithm. Genetic algorithms are based on the principle of genetics and evolution and stand up as a powerful tool for solving search and optimization problems. Selection, crossover and mutation are three basic genetic operators for functioning of the algorithm. Optimization of PID parameters is evaluated using genetic algorithm with various performance indices i.e. ISE, ITAE, IAE and IME. Also the responses of three tank level process using genetic PID controller with various performance indices are analyzed and compared. Analysis is performed through computer simulation using Matlab/Simulink toolbox. The comparative study shows that the application of genetic algorithm based PID controller with ISE index gives the best performance compared to other indices for three tank level process.

Index Terms—Genetic algorithm, PID controller, Performance indices, Three tank process.

I. INTRODUCTION

The liquid level control in multi tank process is of paramount importance especially in food processing, Pharmaceutical, Petrochemical and other industries. In industries level control process with large dead time are difficult to control [1].In industries the final product quality depends on the accuracy of PID controller. The controller is used to maintain the process variable to the set point automatically. The conventional PID controller gives corrective action only after error has developed but not in advance. The only way to achieve better performance is to use genetic algorithm based PID controllers instead of conventional controllers[2].In this paper the response of three tank level process with genetic algorithm based PID controller for various performance indices are compared.

II. CASE STUDY

A three tank level process used in industrial applications is shown in Fig.1.Here

q : Liquid inflow rate initially

q_0 : Liquid outflow rate finally

h1,h2,h3 :Liquid height in three tanks respectively
A1,A2,A3: c.s.area of three tanks respectively

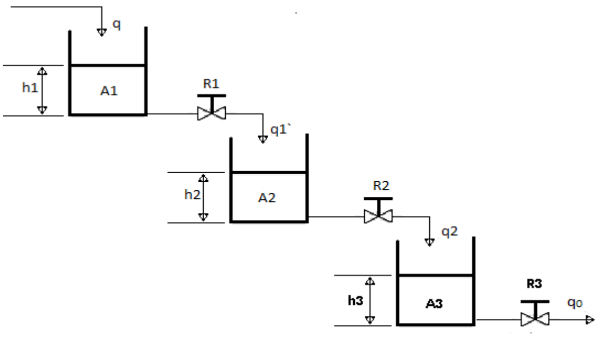


Fig.1 Three tank level control system

The overall transfer function of three tank level control system is

$$G(s) = 1 / (1+ T_1s) (1+ T_2s) (1+ T_3s) \quad (1)$$

Where A1R1 = T_1; A2R2 = T_2; A3R3 = T_3
For simulation purpose the above transfer function is considered as

$$G(s) = 6/(s+1) (s+2) (s+3) \quad (2)$$

III. GENETIC ALGORITHM

Genetic Algorithm (GA) is the stochastic global search optimization technique based on the process of natural evolution. GA is an effective and efficient algorithm used to solve difficult search and optimization problems quickly and reliably. A genetic algorithm is typically initialized with a random population consisting of individuals. This population is usually represented by a real-valued number or a binary string called a chromosome. How well an individual performs a task is assessed by the objective function. The objective function assigns each individual a corresponding number called its fitness. The fitness of each chromosome is assessed and a survival of the fittest strategy is applied.

GAs include three major operators: selection, crossover, and mutation [3], in addition to four control parameters: population size, selection, crossover and mutation rate.

A. Selection

During the reproduction phase the fitness value of each chromosome is

Manuscript Received on October 2013.

P.Srinivas, Associate Professor, Department of EIE, V.R.Siddhartha Engineering College, Vijayawada, India.
K.Vijaya Lakshmi, Assistant Professor, Department of EIE, V.R.Siddhartha Engineering college, Vijayawada, India.
V.Naveen Kumar, Assistant Professor, School of Electronics Engineering, VIT University, Vellore, India.

assessed. This value is used in the selection process to provide bias towards fitter individuals. Just like in natural evolution, a fit chromosome has a higher probability of being selected for reproduction. Four common methods for selection are:

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

B. Crossover

Once the selection process is complete, the crossover algorithm is initiated. The crossover operation swaps certain parts of the two selected strings in a bid to capture the good parts of old chromosomes and create better new ones. Genetic operators manipulate the characters of a chromosome directly, using the assumption that certain individual's gene codes, on average, produce fitter individuals. The crossover probability indicates how often crossover is performed. Crossover can be performed with binary encoding, permutation encoding, value encoding and tree encoding.

C. Mutation

Mutation is a process by which the chance for the GA to reach the optimal point is reinforced through just an occasional alteration of a .value at a randomly selected bit position. The mutation process may quickly generate those strings which might not be conveniently produced by the previous reproduction and crossover processes. Although mutation is necessary, sometimes it may suddenly spoil the opportunity of the current appropriate generation. The probability of mutation is normally low because a high mutation rate would destroy fit strings and degenerate the genetic algorithm into a random search. Like crossover, mutation can also be performed for all types of encoding techniques.

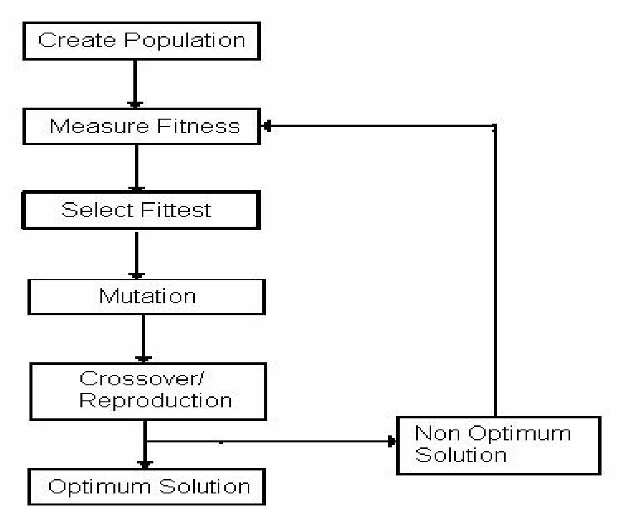


Fig. 2 Flowchart of GA

The flowchart of Genetic algorithm is shown in Fig. 2.Genetic algorithms (GAs) begin with a set of solutions represented by chromosomes, called population. Solutions from one population are taken and used to form a new population, which is motivated by the possibility that the new population will be better than the old one. Further, solutions are selected according to their fitness to form new solutions, that is, off springs. The above process is repeated until some condition is satisfied.

The convergence criterion of a genetic algorithm is a user-specified condition e.g. the maximum number of generations or when the string fitness value exceeds a certain threshold. The essential steps in implementing a GA are

1. Initialization of Population.
2. Setting the GA Parameters.
3. Performing the GA

D. Initializing the Population of the GA

The GA starts with the initialization of the population size, variable bounds and the evaluation function. The codes that are used to initialize the GA are as follows:

Population Size - The first stage of writing a Genetic Algorithm is to create a population. This command defines the population size of the GA. Generally the bigger the population size the better is the final approximation [7].

VariableBounds- In this application, GA is used to optimize the gains of a PID controller. So three strings are assigned to each member of the population, these members will be comprised of a P, I and a D string that will be evaluated throughout the course of the GA processes. The three terms are entered into the GA by declaration of a three-row variable bounds matrix. The number of rows in the variable bounds matrix represents the number of terms in each member of the population

EvalFN - The evaluation function is the Matlab function used to declare the objective function. It will fetch the file name of the objective function and execute the codes and return the values back to the main codes.

Initialisega – This command is from the GAOT toolbox. It will combine all the previously described terms and creates an initial population of 80 real valued members between –100 and 100 with 6 decimal place precision [8].

E. Setting the GA Parameters

The codes that are used to set up the GA are as follows:

Bounds- The variable bounds for the GA are to search within a specified area. These bounds define the entire search space for the GA.

StartPop- It starts the population of the GA

Opts- The options for the GA consist of the precision of the string values i.e. 1e-6, the declaration of real coded values, 1, and a request for the progress of the GA to be displayed, 1, or suppressed, 0.

SelectOps- When using the 'normGeom Select' option, the only parameter that has to be declared is the probability of selecting the fittest chromosome of each generation [9].

XOverOptions - This is where the number of crossover points is specified.

MutFNs- The 'multiNonUnifMutation' was chosen as the mutation operator as it is considered to function well with multiple variables.

F. Objective Function of the Genetic Algorithm

The most challenging part of creating a GA is writing the objective function. In this application, the objective function is required to evaluate the best PID controller for the system. We can create an objective function to find a PID controller that gives the smallest overshoot, fastest

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

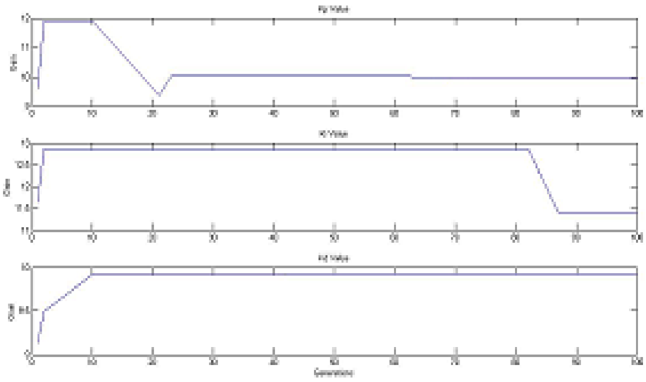
rise time or quickest settling time. However in order to combine all of these objectives, we designed an objective function that minimizes the error of the controlled system instead. Each chromosome in the population is passed into the objective function one at a time. The chromosome is then evaluated and assigned a number to represent its fitness, the bigger its number the better its fitness [10]. The GA uses the fitness value of chromosome's to create a new population consisting of the fittest members. Each chromosome consists of three separate strings constituting a P, I and D term, as defined by the 3-row 'bounds' declaration when creating the population. 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 results in reduction of the program compilation time. The system transfer function is defined in another file and imported as a global variable. A unit step input is given to the controller to evaluate its performance and the error is assessed using an error performance criterion such as Integral Square Error or in short ISE. The ISE is an accepted measure of control and of quality but its practical use as a measure of quality is somehow limited. The chromosome is assigned an overall fitness value according to the magnitude of the error, the smaller the error the larger the fitness value. Below is the codes used to implement the ISE, ISTE, IAE and MSE performance criteria.

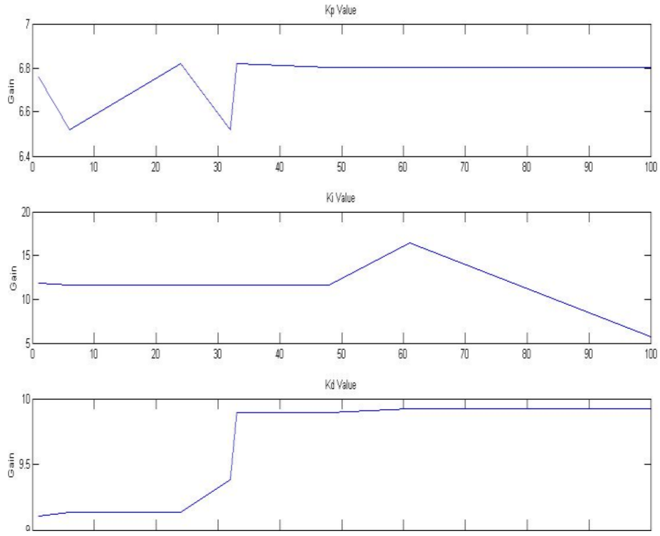
IV. SIMULATION RESULTS

The MATLAB programs for Genetic algorithm based PID controller tuning with ISE,IAE,ISTE and MSE performance indices are implemented and simulated results are plotted. Then the unit step responses of three tank process are plotted and compared for the simulated PID tuning parameters for different performance indices.

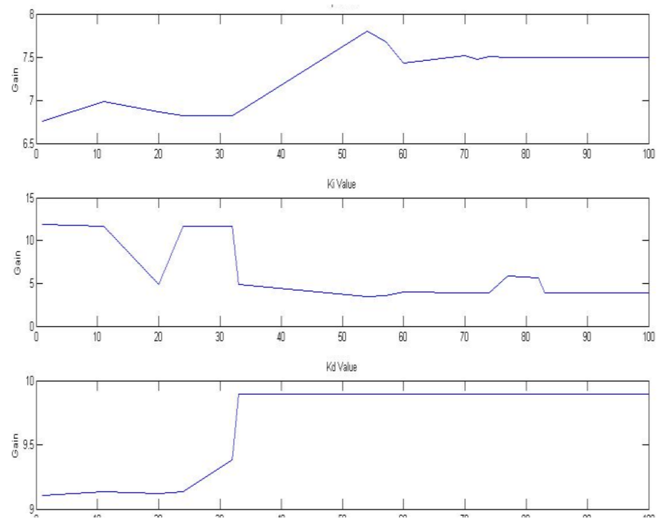
The variation of tuning parameters (K_P, K_I and K_D) of genetic algorithm based PID controller for ISE, ISTE, IAE and IME performance indices are plotted in figures 3(a),3(b),3(c) and 3(d) respectively and the corresponding unit step responses of three tank process are plotted in figures 4(a),4(b),4(c) and 4(d) respectively.



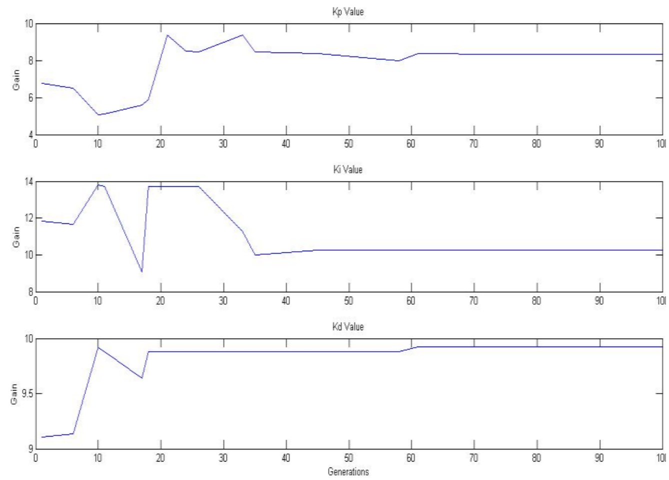
(a) ISE Criteria



(b)ITAE criteria

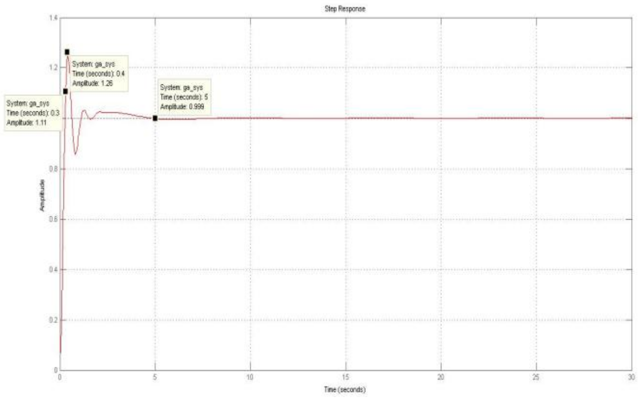


(c) IAE criteria

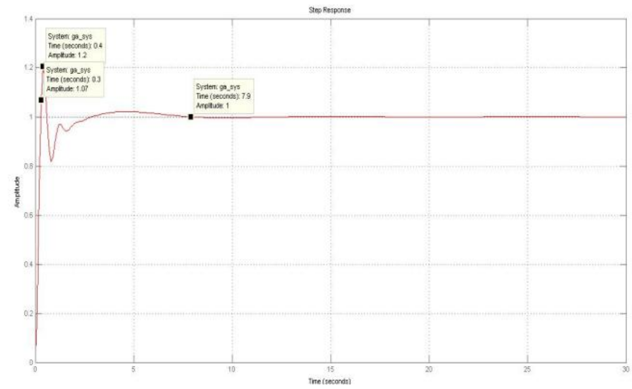


(d) IME criteria

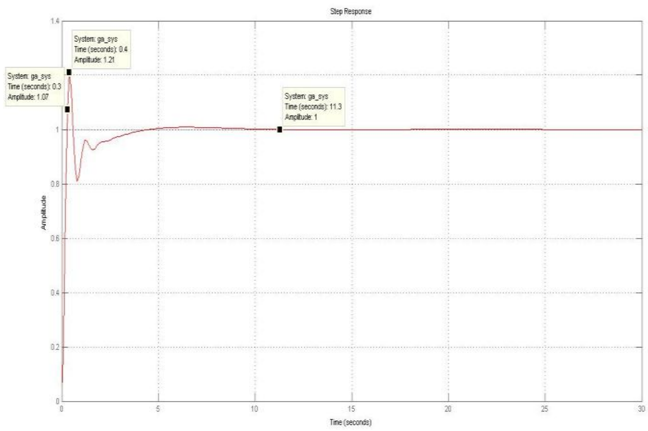
Fig. 3 Variation of tuning parameters of genetic algorithm based PID controller for various performance indices



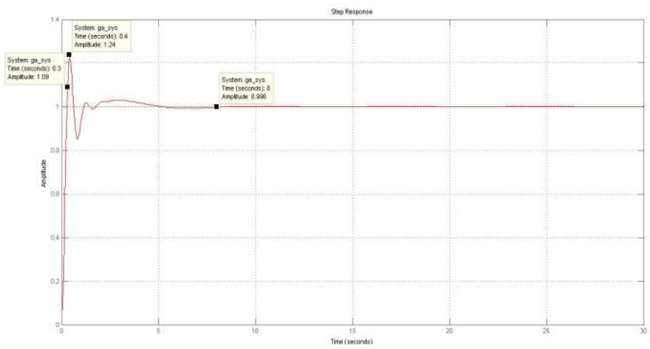
(a) ISE Criteria



(b) ITAE criteria



(c) IAE criteria



(d) IME criteria

Fig. 4 Unit step response of three tank system with genetic algorithm based PID controller for various performance indices

The observed time domain specifications from the response graphs are tabulated in Table. 1.From the tabulated results, we observed that ISE method gave quick response with slightly more overshoot compared with other performance indices. ITAE method gave less peak overshoot compared to other performance indices. Rise and peak times are same in all performance indices.

Table.1: Comparison of time domain specifications for various performance indices

Time domain specifications	Performance indices			
	ISE (1.84)	ITAE (3.11)	IAE (3.75)	MSE (0.05)
Rise time, sec	0.30	0.30	0.30	0.30
Peak time, sec	0.40	0.40	0.40	0.40
Settling time, sec	5.00	7.90	11.3	8.00
Peak overshoot	1.26	1.20	1.21	1.24

V. CONCLUSIONS

Research has been carried out to get an optimal tuning of PID controller by using genetic algorithms. This paper provides the G.A program for various performance indices in mat lab, which can be directly run through Mat lab 7.10. G.A is applied to find optimal solution for the unit step response of three tank system with GA based PID controller & indicates that G.A is powerful global searching method. The G.A designed PID controller with ISE criteria is much better in terms of rise time, settling time, peak time then other performance indices like ITAE,IAE and MSE. Future work includes the use of evolutionary algorithms to optimize the tuning of PID controllers for various industrial processes.

REFERENCES

1. Joseph J. DiStefano, Allen R. Stubberud, Ivan J. Williams,Feedback and control systems, McGraw-Hill Professional,1995- Schaum's Series.
2. David W. Spitzer, ‘Advanced Regulatory Control Applications and Techniques’, Momentum Press, 2009.
3. D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Co., Inc., 1989.
4. S.N.Sivanandam, S.N.Deepa, Introduction to Genetic Algorithms, Springer Publications Co., 2010.
5. Gotshall,S. and Rylander, B., Optimal Population Size And The Genetic Algorithm.”, Proc On Genetic And Evolutionary Computation Conference,2000.
6. K. Krishnakumar and D. E. Goldberg, Control System Optimization Using Genetic Algorithms”, Journal of Guidance, Control and Dynamics, Vol. 15, No. 3, pp. 735-740, 1992.
7. Chris Houck, Jeff Joines, and Mike Kay, "A Genetic Algorithm for FunctionOptimization: A Matlab Iplementation”.
8. Chipperfield, A. J., Fleming, P. J., Pohlheim, H. and Fonseca, C. M., A Genetic Algorithm Toolbox for MATLAB , Proc. International Conference on Systems Engineering, Coventry, UK, 6-8 September, 1994.
9. Ayman A. Aly, PID Parameters Optimization Using Genetic Algorithm Technique for Electrohydraulic Servo Control System, Intelligent Control and Automation, 2011, 2,69-76.
10. Y. J. CAO and Q. H. WU, Bristol, Teaching Genetic Algorithm Using Matlab, Int. J.Elect. Enging. Educ., Vol. 36, pp. 139–153. Manchester U.P., 1999. NCSU-IE TR 95-09, 1995.