Real Time Application of Ants Colony Optimization

Dr.S.M.GiriRajkumar
Senior Assistant Professor
School of Electrical & Electronics Engineering
SASTRA University, Thanjavur
Tamilnadu-613402

Dr.K.Ramkumar
Senior Assistant Professor
School of Electrical & Electronics Engineering
SASTRA University, Thanjavur
Tamilnadu-613402

Sanjay Sarma O.V
Department of Mechatronics
School of Mechanical Engineering
SASTRA University, Thanjavur
Tamilnadu-613402

ABSTRACT

Automatic control has played a vital role in the advancement of engineering and science. It is also essential in such industrial operations as controlling pressure, temperature, humidity, viscosity and flow in the process industries. Proportional Integral Differential (PID) controllers marked its place in many of the industrial processes. Tuning a controller is the adjustment of its control parameters. Computational Intelligence (CI) an off shoot of Artificial Intelligence relies on heuristic algorithms mainly evolutionary computation. Swarm intelligence (SI) a derivative of CI, describes the collective behaviour of decentralized, self-organized systems. Ant behaviour was the inspiration for the Meta heuristic optimization technique. This paper presents an application of an Ant Colony Optimization (ACO) algorithm to optimize the parameters in the design of a (PID) controller for a highly nonlinear conical tank system. The proposed work discusses in detail, the ACO, a CI technique, and its application over the parameter tuning of a PI controller in a real time process. The designed controller's ability in tracking a given set point is compared with an Internal Model Control (IMC) tuned controller.

Keywords: PID controllers, Computational Intelligence, Ants Colony Optimization, Internal Modal Control, Meta heuristic optimization.

1. INTRODUCTION

The widely used PID industrial controller uses a combination of proportional, integral and derivative action on the control error to regulate its output. Owing to its simple structure, easy tuning and effectiveness, this technology has been a mainstay for long among practicing engineers [1]. PID control is a generic feedback control technology and it makes up 90% of automatic controllers in industrial control systems. The PID control was first placed in the market in 1939 and has remained the most widely used controller in process control until today. The basic function of the controller is to execute an algorithm based on the control engineer's input and hence to maintain the output at or around the set point [2]. The popularity of PID controllers is due to their functional simplicity, reliability and cost effectiveness. They provide robust and reliable performance for most systems and the PID parameters i.e. the proportional, integral and differential constants are tuned to ensure a satisfactory closed loop performance [3]. A PID controller improves the transient response of a system by reducing the overshoot, and by shortening the settling time of a system [4]. The PID control algorithm is used to control almost all loops in process industries and is also the cornerstone for many advance control algorithms and strategies [2]. For this control loop to function properly, the PID loop must be properly tuned. Standard methods for tuning include Ziegler-Nichols Ultimate-cycle tuning [5], Astrom and Hagglund [6], Cohen-Coon's method [7], and many other traditional techniques. Although new methods are proposed for tuning the PID controller, their usage is limited due to the complexities arising at the time of implementation and their incompetence towards nonlinear systems.

However, despite decades of development work, surveys indicating the state of the art of control industrial practice report sobering results. For example, Ender (1993) states that, in his testing of thousands of control loops in hundreds of plants, it has been found that more than 30% of installed controllers are operating in manual mode and 65% of loops operating in automatic mode poorly tuned. The Handbook of PI and PID controller Tuning Rules by Aidan O.Dwyer has recorded 408 separate sources of tuning rules since the first such rule which was published by Callender et al. in 1935. In a striking statistic, 293 sources of tuning rules have been recorded since 1992 reflecting the upsurge of interest in the use of the PID controller recently.

Although these many tuning rules are available in literature, most of the rules are applicable only for a first order system with a time delay. So clearly they are not meant to be applied for higher order nonlinear systems. In order to apply them we may go for approximating the system to a FOPTD (first order with time delay)**.**This can primarily be done either using Taylor's approximation or Skogestad's approximation. But the word approximation itself suggests that the parameters obtained using the application of these traditional tuning rules on the approximated system will also be a very big compromise. The intensity of compromise depends on the magnitude of degree diminution. This approximation could itself fail if the higher order system has a complex time constant where it will be a tedious process and sometimes impossible.Certain methods are available in applying over specific systems. And hence reduces the acceptance of the method.

Tuning a PID controller means setting the proportional, integral and derivative constant to get the best possible control for a particular process. Adjusting the controller gains, to satisfy the performance specifications like margin of stability, transient response and bandwidth, improves the system robustness. The performance of the tuned controller can be represented as a function of error for quantitative analysis. The commonly employed performance indices are Integral Absolute Error, Integral Squared Error, Integral of time multiplied by absolute value of error and Integral of time multiplied by squared error.

Control design is called “optimal control” when a predefined criterion is optimized .Optimality is just with respect to the criterion at hand and the real performance depends on the suitability of the chosen criterion. The Ant Colony Optimization (ACO) algorithm is a meta-heuristic algorithm for the approximate solution of combinatorial optimization problems that has been inspired by the foraging behavior of real ant colonies [9-11]. In this algorithm, computational resources are allocated to a set of relatively simple agents that exploit a form of indirect communication mediated by the environment to find the shortest path from the ant nest to a set target. Ants can follow through to a food source because, while walking, they deposit pheromone on the ground, and they have a probabilistic preference for paths with larger amount of pheromone [12-13].

The model of the process under study is very important for its tuning as the accuracy of the tuned controller parameters is greatly dependent upon the degree of accuracy of the system model with that of the real system. As per the fundamentals, it is possible to approximate the actual input-output mathematical model of a very-high order, complex, dynamic process with a simple model consisting of a first or second order process combined with a dead-time element. Thus, a common practice followed in industries for the purpose of control design and process analysis is to model the dynamics of the process near the operating point by simpler models such as first order process with time delay (FOPTD).

Analysis of the proposed controller design gives a satisfactory performance over a wide range of process operations. Control design is called “optimal control” when a predefined criterion is optimized .Optimality is just with respect to the criterion at hand and the real performance depends on the suitability of the chosen criterion. Using ACO approach, global and local solutions could be simultaneously found for the better tuning of the controller parameters. The controller designed is independent of the mathematical model of networks, thus getting rid of adverse effects. Hence, in industries, the difficulties to achieve an optimal PID gain without prior expert knowledge can be overcome.

The proposed work’s objective is to use Ants Colony Optimisation in order to obtain optimal values for control parameters, Kp and Ki of a PI controller for a conical tank process, which is highly non-linear. The problem of non-linearity is overcome by linearizing over four suitable ranges. Hence four sets of PI parameters are proposed in this paper. Each of the parameters proportionality constant Kp and integral constant Ki represents a particle which changes in the search space in order to minimize the error function (objective function in this case). The error function used here is Integral Time of Absolute Errors (IAE). The proposed work deals with the development of the mathematical model for the non-linear conical tank process. The tuning results of conventional techniques are discussed in section 3. Sections 4 deal with the explanation of the ACO algorithm and its implementation. The comparative studies and results are given in section 5.

2. DEVELOPMENT OF MATHEMATICAL MODEL FOR THE REAL TIME PROCESS.

Feedback control systems are often referred to as closed-loop control systems. In a closed-loop control system the actuating error signal, which is the difference between the input signal and the feedback signal, is fed to the controller so as to reduce the error and bring the output of the system to a desired value.

The conical tank system, which exhibits the property of non-linearity, is taken for the real time analysis of the designed controllers. The process dynamics are analysed in four segments so as to obtain effective models for the operating ranges. The operating ranges are concluded for 0-15cm as model-1, 15-27cm as model-2, 27-36cm as model-3 and 36-43 cm as model-4. And the corresponding mathematical models are obtained for these sections.

2.1. Experimental Setup

The real time experimental system consisting of a conical tank, reservoir and water pump, current to pressure converter, compressor, Differential Pressure Transmitter(DPT), ADAM module, and a Personal Computer which acts as a controller forms a closed loop system. The inflow rate to the conical tank is regulated by changing the stem position of the pneumatic valve by passing control signal from computer to the I/P converter through digital to analog converter (DAC) of ADAM module. The operation current for regulating the valve position is 4-20 mA, which is converted to 3-15 psi of compressed air pressure. The water level inside the tank is measured with the differential pressure transmitter which is calibrated for 0-43 cm and is converted to an output current range of 4-20 mA. This output current from DPT is passed through 1K ohms resistance converting it to 1-5V range, which is given to the controller through analog to digital converter (ADC) of ADAM module.

The ADAM module is used for interfacing the personal computer with the conical tank system thus forming a closed loop. It has four slots for four converter cards. In the current process, two slots are used, one containing Analog to Digital Converter (ADC) card and the other containing the Digital to Analog Converter (DAC) card. The ADC card has 8 analog input channels with a range of 4-20 mA and DAC has 4 analog output channels with a range of -10V to +10V accommodating both positive and negative terminals. The sampling rate of the module is 10 samples/sec and the baud rate is set to 9,600 bytes per sec with a 16 bit resolution. The ADAM module is connected to the personal computer through RS-232, serial cable. The module can be operated manually through console software provided and also with programming software like LABVIEW, MATLAB etc., Here MATLAB based script files are used in interfacing the controller with the real time system.

MATLAB software gives us the flexibility of interfacing the ADAM module with the personal computer. The system is prepared to access the module through m-files. The controller equation which is discussed further in this paper is accessed through this programmed frame work.

The piping and instrument diagram of the system is shown in Figure 1. The system specifications are shown in the Table 1. The ADAM interfacing module, control valve, differential pressure transmitter are shown in the Figure 2.The real time system developed is shown in the Figure 3.

Table 1: System specifications

PART NAME	DETAILS
Conical tank	Stainless steel body, height – 45 cm,
	Top Diameter – 33.74cm, Bottom Diameter-0.8 cm,
Differential Pressure Transmitter	Capacitance type, Range 2.5-250 mbar,
	Output 4-20 mA
Pump	Centrifugal 0.5 HP
Control Valve	Size ¼" Pneumatic actuated,
	Type: Air to open, Input 3-15 psi
Rotameter	Range 0-18 lpm
Air regulator	Size 1/4" BSP, Range 0-2.2 bar
I/P converter	Input 4-20 mA, Output 0.2-1 bar
Pressure gauge	Range 0-30 psi
Compressor	20 psi

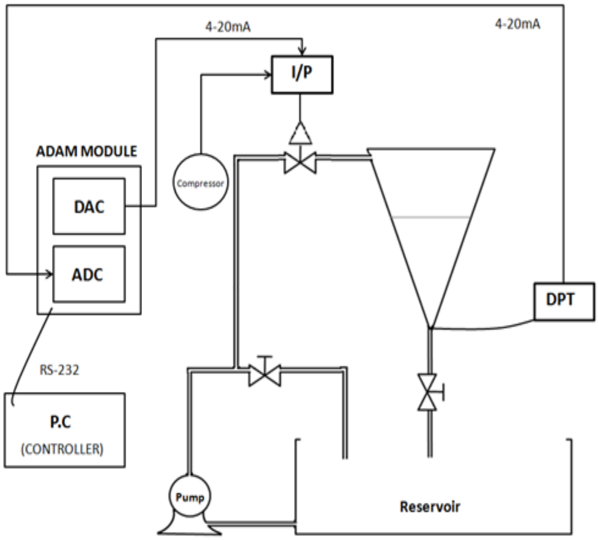
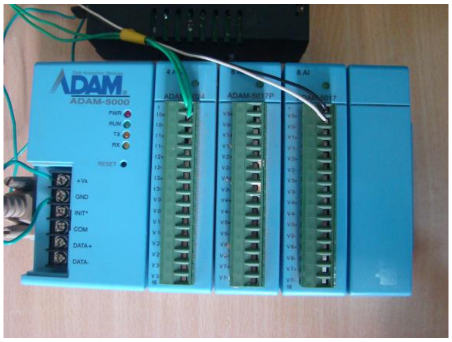


Figure 1: Piping and instrument diagram of the process system



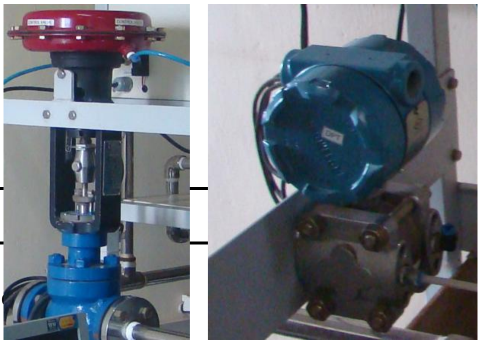


Figure 2: Images of ADAM module, Control valve, Differential Pressure Transmitter.

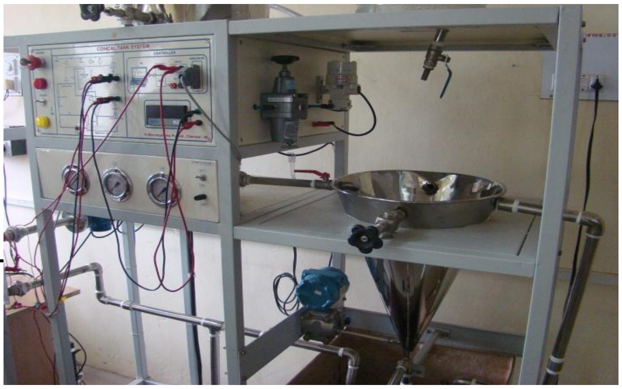


Figure 3: Photograph of experimental setup.

2.2 Process Modeling:

System identification is normally done by applying step response methods. In application of these methods, response of the conical tank for a given flow is required. Through this response, suitable model and model parameters are estimated. model selection is based upon the open loop step response of the system, for which the inflow valve is set to different positions by manual operation through ADAM console software. Owing to the non-linearity in the shape of the conical tank, a single range response cannot cover the entire range.

So, various trials were conducted for different flow ranges and valve openings to obtain a typical response curve. Four responses covering the full height of the conical tank were obtained for 0-15cm as model-1, 15-27cm as model-2, 27-36cm as model-3 and 36-43 cm as model-4.

The models were checked with the two point method and Sunderesan Kumaraswamy [14] method. The models checked with the two point method were not close to the real time response where Sundaresan Kumaraswamy method is found to be more coinciding. So, in this work we follow the Sunderesan Kumarasamy method, with the models obtained as the response curves of the open loop response with the time delay inclusion directly, instead of Pade's approximation techniques.

As per the structure of the curves, the model is predicted to be of the form similar to FOPTD,

$$G(s) = \frac{Ke-\tau ds}{\tau s+1} \qquad \text{Eq(2.1)}$$

Where K= process gain

τ=first order time constant

τ_d = delay time

The maximum inflow rate to the tank is maintained at 5.467 lpm.

The comparative response curves between the real time and simulated model, with reference to the range 0-15 cm .The model was estimated as

$$G(s)_{\text{model 1}} = \frac{2.74e^{-10.64s}}{(4.24s+1)} \qquad \text{Eq(2.2)}$$

For the region 15-27 cm the mathematical model is given below .

$$G(s)_{\text{model 2}} = \frac{2.19e^{-13.98s}}{(8.9s+1)} \qquad \text{Eq(2.3)}$$

For the region 27-36 cm the mathematical model and the validation curves are given below .

$$G(s)_{\text{model 3}} = \frac{1.6e^{-15.51s}}{(12.19s+1)} \qquad \text{Eq(2.4)}$$

For the region 36-43 cm the mathematical model is given below.

$$G(s)_{\text{model 4}} = \frac{1.36e^{-19s}}{(15.56s+1)} \qquad \text{Eq(2.5)}$$

The validation curves for the four models are given in Figures 4-7. The so framed model is simulated for a step input for the set points whose graphs are presented in the figures. The system is then subjected to a step input for all the four set points and the corresponding real time graphs are presented comparing with the simulated graphs.

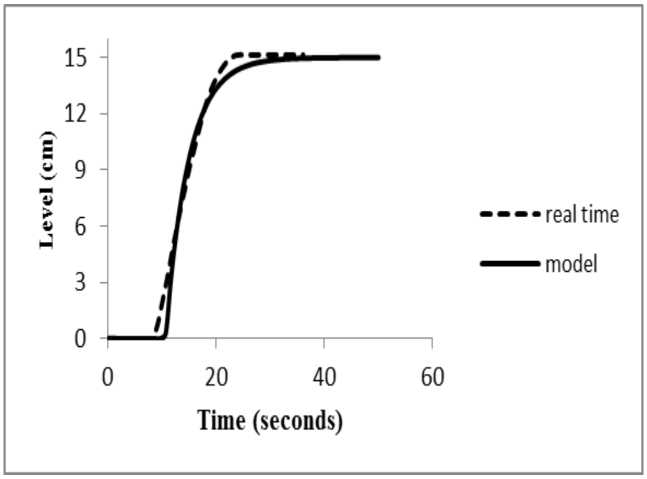


Figure 4: Comparison of real time and simulated responses of model 1.

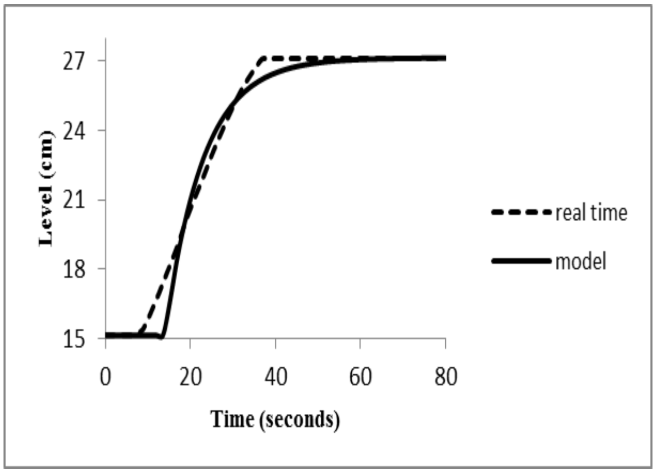


Figure 5: Comparison of real time and simulated responses of model 2

.

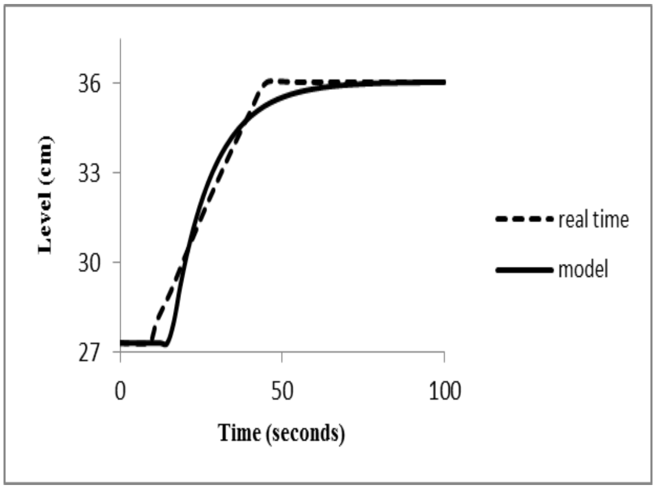


Figure 6: Comparison of real time and simulated responses of model 3.

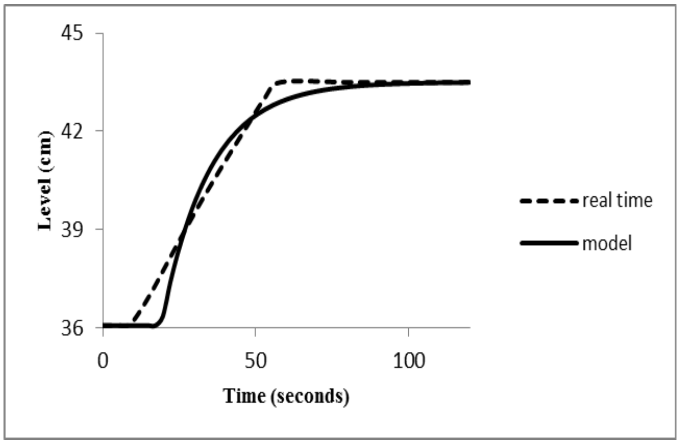


Figure 7: Comparison of real time and simulated responses of model 4.

3. CONVENTIONAL DESIGN TECHNIQUE

The basic PI controller parameters are proportional gain, Kp and integral gain Ki. Numerous methods were developed over last fifty years for setting the parameters of a PID controller. In this paper, it is considered to proceed the tuning with Internal Model Control(IMC), a tuning technique proposed by Skogestad for PI tuning.

The IMC technique is one of the recent traditional tuning techniques that yield better values among the techniques available for conventional methods [2]. For a FOPTD model of the mentioned form in Equation-1, the IMC tuning values based on Skogestad proposal is given as

$$Kp = \frac{1}{k}\left(\tau/(\tau c + \tau d)\right) \qquad \text{Eq(3.1)}$$

Where $\tau_c=\tau_d$ as per Skogestad, and integral time constant T_i is given as, $T_i = \tau$, and hence, we have $K_i=K_p/K_i$. Applying the technique for both the models, we get the IMC tuning parameters as in Table 2.

Table 2: Control parameters for PID controller for the four models by conventional design technique

PARAMETERS	Model1	Model 2	Model 3	Model 4
K_p	0.0727	0.1453	0.2456	0.3011
K_i	0.0171	0.0163	0.0201	0.0193

4. ANTS COLONY ALGORITHM

The Ant System is a new kind of co-operative search algorithm inspired by the behavior of colonies of real ants. The ants colony algorithm was applied to travelling salesman problem [15-17] The blind ants are able to find astonishing good solutions to shortest path problems between food sources and their home colony. The medium used to communicate information among individuals regarding paths, and decide where to go, was the pheromone trails. A moving ant lays some pheromone on the path they move, thus marking the path by the substance. While an isolated ant moves essentially at random, it can encounter a previously laid trail and decide with high probability to follow it, and also reinforcing the trail with its own pheromone. The collective behavior that emerges in a form of autocatalytic behavior where the more the ants following a trail, the more attractive that trail becomes for being followed.

4.1 The Path of Ants

There is a path along which ants are walking from nest to the food source and vice versa. In the forties and fifties of the twentieth century, the French entomologist Pierre-Paul Grass´e [18] observed that some species of termites react to what he called “significant stimuli”. If a sudden obstacle appears and the path is cut off, the choice is influenced by the intensity of the pheromone trails left by proceeding ants. On the shorter path more pheromone is laid down. The Figure 8 details the behavior of ants when faced with an obstacle in its search path.

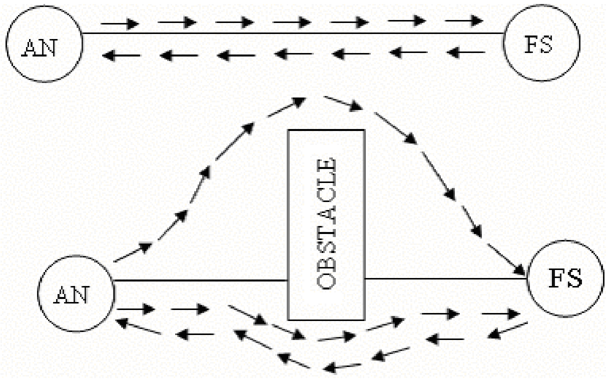


Figure 8: Path traced by ants without and with obstacle

Ants Colony algorithm can be applied for the continuous function optimization problems. Here, the domain has to be divided into a specific number of *R* randomly distributed regions. These regions are indeed the trial solutions and act as local stations for the ants to move and explore. The fitness of these regions are first evaluated and sorted on the basis of fitness. When an algorithm designed for combinatorial optimization is used to tackle a continuous problem, the simplest approach would be to divide the domain of each variable into a set of intervals. However, when the domain of the variables is large and the required accuracy is high, this approach is not viable. For this reason, ACO algorithms have been developed, which are specifically designed for continuous and mixed continuous-discrete variable [19-20] Totally a population of ants explores these regions; the updating of the regions is done locally and globally with the local search and global search mechanism respectively. The distribution of local and global ants is illustrated in Figure 9 and the flowchart of the Ants Colony algorithm is presented in Figure 10.

4.2 Global Search

The global search creates G new regions by replacing the weaker portions of the existing domain. In the ACO random walk and trial diffusion are utilized for global search. By random walk procedure, the ants move in new directions in search of newer and richer stocks of food source. In the ACO simulation such a global search in the entire domain is done by process equivalent to crossover operation and mutation operations in G.A. Adding or subtracting with a probability proportional to the mutation probability carries out the mutation step in ACO. The mutation step is reduced as per the relation.

$$\Delta(T, R) = R\left[1 - r^{(1-T)^{b}}\right]$$

Where ‘r’ is a random number from [0, 1] ‘R’ is the maximum step size. ‘T’ is the ratio of the current iteration number to that of the total number of iterations; ‘b’ is a positive parameter controlling the degree of nonlinearity.

The mutation radius is nonlinearly reduced with increasing iterations. The scaling down enables enhanced probability of locating maximum by concentrated search procedure called trial diffusion. The trial diffusion is quite similar to arithmetic cross over. In this step two parents are selected at random from the parent population space. The elements of the child’s vector can have either (1) The corresponding element from the first parent,

(2) The corresponding element from the second parent and (3) A combination arrived from a weighted average of the above. i.e. If the random number is less than 0.5

$$X_{(child)} = \alpha X_{i(parent1)} + \left(-\alpha\right) X_{i(parent2)}$$

where α is a uniform random number in the range (0 –1).

If the random number is in between 0.5 and 0.75 then:

$$X_{(child)} = X_{i(parent1)}$$

And if the random number is in between 0.75 and 1.00 then

$$X_{(child)} = X_{i(parent2)}$$

4.3 Local Search

In the local search, the ants have the capability of selecting regions proportional to the current pheromone values of superior and inferior regions. Local updating is applied only on superior regions. In an ACO, local ants select a region 'i' given by

$$P_i(t) = \frac{\tau_i(t)}{\sum \tau_{k(t)}}$$

where 'i' is the region index and τi is the pheromone trail on region i at time t. After selecting the destination, the ant moves through a short distance. The direction of the movement will be the same as that of the previous direction if there is an improvement in the fitness. It there is no improvement it searches in a random direction. If improvement in fitness is found in the above procedure, the regions position vector is updated.

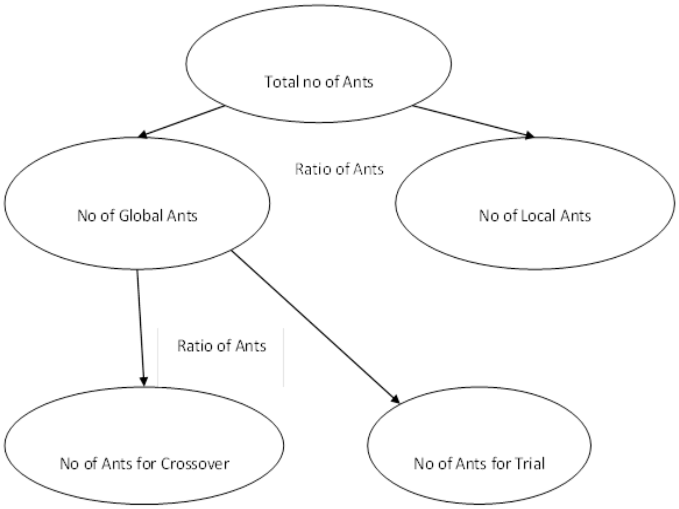


Figure 9: Distribution of Ants for Local and Global search

The Pheromone deposited by the ant is proportional to the increase in fitness. The age of the region is another important parameter in the ACO algorithm. The size of the ant movement in the local search depends on the current age.

•Initially all the regions are assigned with the pheromone value of 1.0

•It better results are obtained the pheromone of region i is modified by

$$Pheromone_{i(new)} = \frac{P_{r(new)} - P_{r(old)}}{P_{r(old)}} + Pheromone_{i(old)}$$

•Now again the average pheromone for each region is calculated and the procedure is repeated as many number of times as they are local ants.

•The termination criterion is the total number of iterations.

4.4 IMPLEMENTATION OF ACO

The parameters Kp and Ki of a PID controller are optimised using ACO.

4.4.1 Selection of ACO parameters

To start up with ACO, certain parameters need to be defined. Selection of these parameters decides, to a great extent, the ability of global minimization. The population size balances the requirement of global optimization and computational cost. Initialising the values of the parameters is as per Table 3.

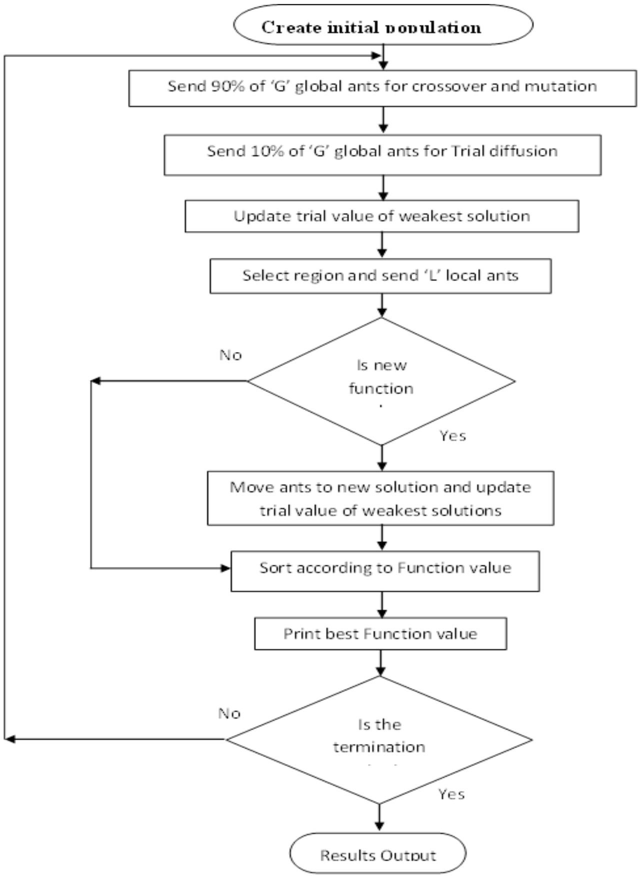

Create initial population
Send 90% of 'G' global ants for crossover and mutation
Send 10% of 'G' global ants for Trial diffusion
Update trial value of weakest solution
Select region and send 'L' local ants
No
Is new function
Yes
Move ants to new solution and update trial value of weakest solutions
Sort according to Function value
Print best Function value
No
Is the termination
Yes
Results Output

Figure 10: Flowchart of Ants Colony Optimization algorithm

Table 3: Initial values of the parameters

Parameter	Specification
Learning constant,c1	2
Learning constant,c2	2

4.4.2 Performance Index of ACO Algorithm

The objective function considered is based on the error criterion. The performance of a controller is best evaluated in terms of error criterion. A number of such criteria are available and, in the proposed work, the controller's performance is evaluated in terms of an Integral of Absolute Errors (IAE) criterion, given by

$$I_{IAE} = \int_0^T |e(t)|dt$$

The IAE weights the error with time and hence emphasizes the error values over a range of 0 to T, where T is the expected settling time.

4.4.3 Termination Criteria

Genetic Algorithm termination can take place either when the maximum numbers of iterations are performed, or when a satisfactory fitness value is attained. The fitness value is the reciprocal of the error, since we consider for a minimization of objective function. In this work, the termination criterion is considered to be the maximum number of iterations. The distribution of the values for the first iteration for Kp and Ki are given below for all the models as shown in Figures 11 - 18. It is clearly seen that the values are well distributed.

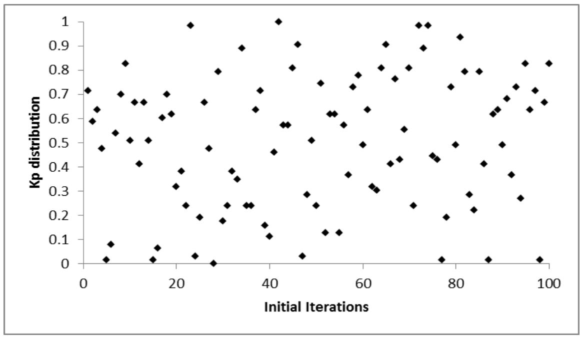


Figure 11: Kp distribution for model1

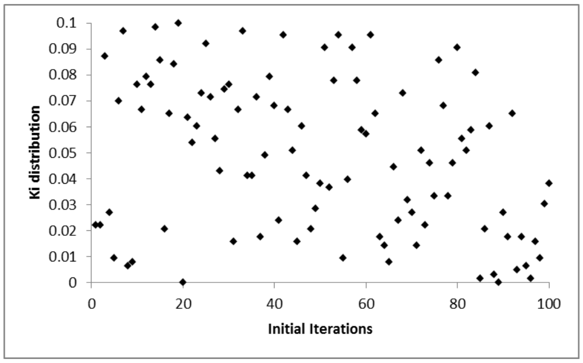


Figure 12: Ki distribution for model 1

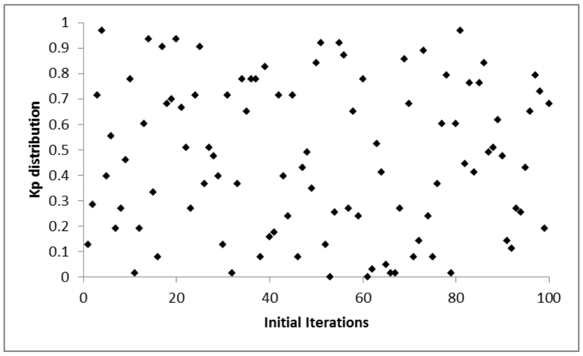


Figure 13: Kp distribution for model2

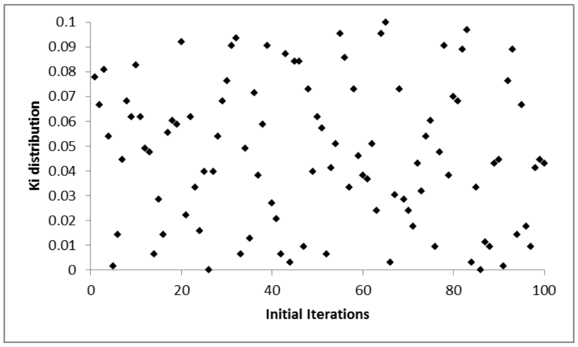


Figure 14: Ki distribution for model2

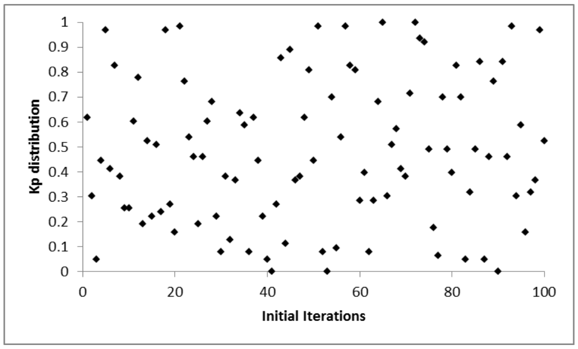


Figure 15: Kp distribution for model3

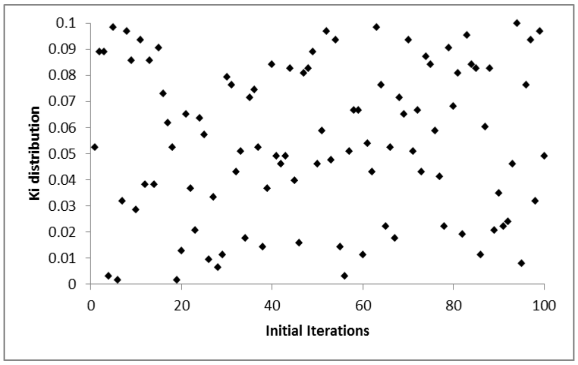


Figure 16: Ki distribution for model3

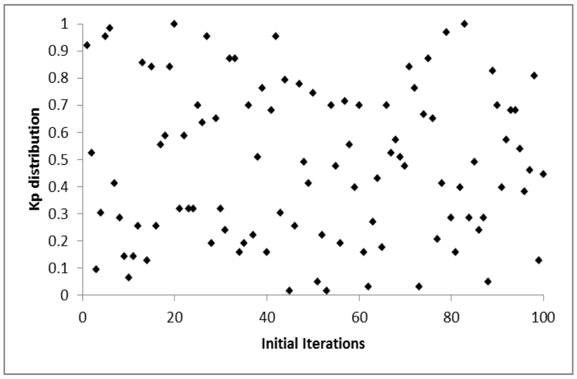


Figure 17: Kp distribution for model4

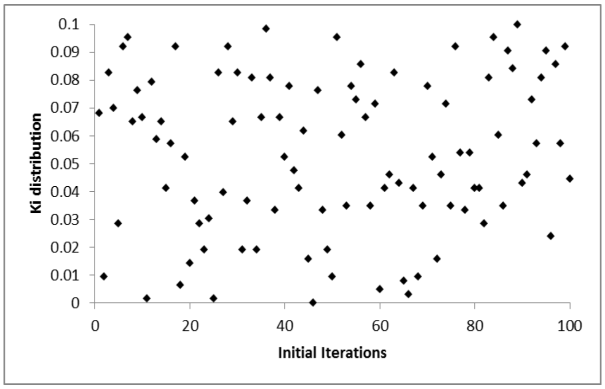


Figure 18: Ki distribution for model4

For each iteration, the best among the 100 particles considered as potential solution are chosen. Therefore the best values for 100 iterations are sketched with respect to iterations for Kp and Ki and are shown in Figures 19-26 for all the four models.

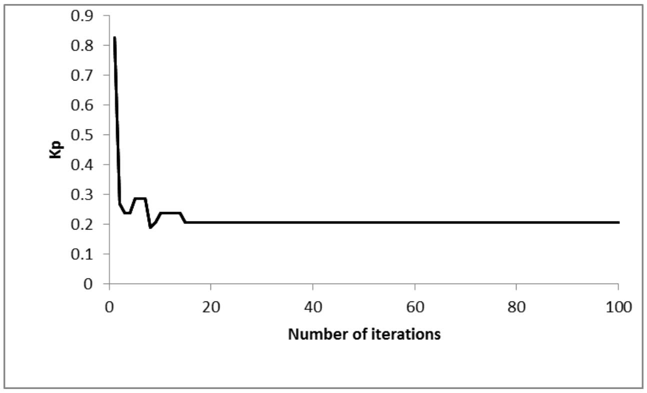


Figure 19: Best solutions of Kp to model 1 for 100 iterations

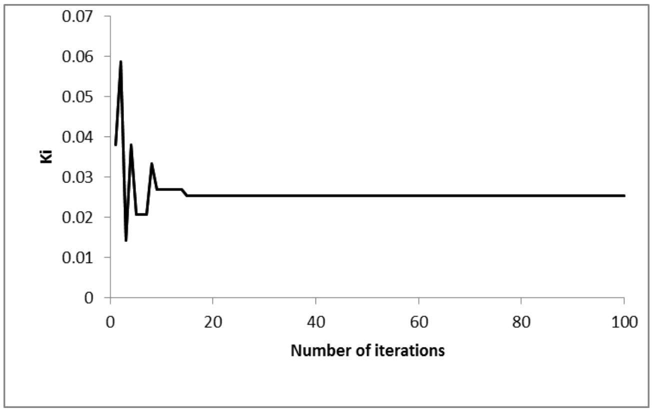


Figure 20: Best solutions of Ki to model 1 for 100 iterations

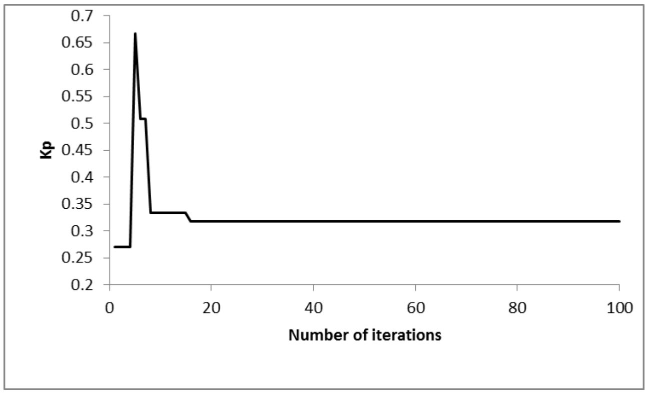


Figure 21: Best solutions of Kp to model 2 for 100 iterations

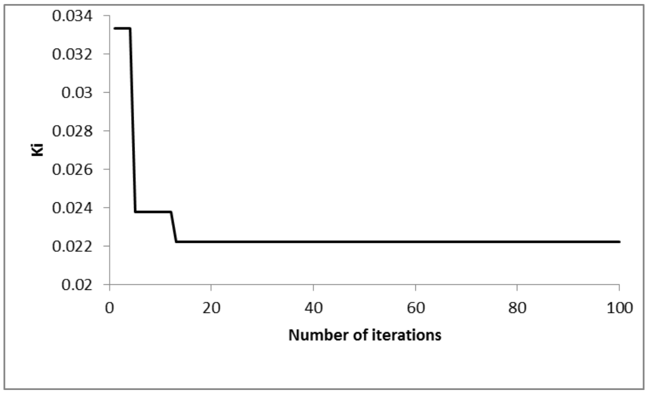


Figure 22: Best solutions of Ki to model 2 for 100 iterations

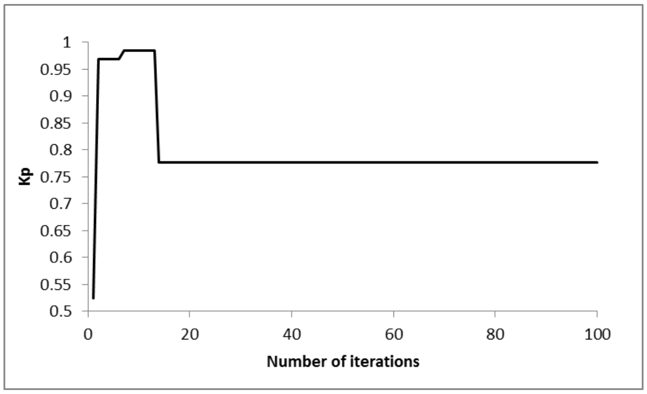


Figure 23: Best solutions of Kp to model 3 for 100 iterations

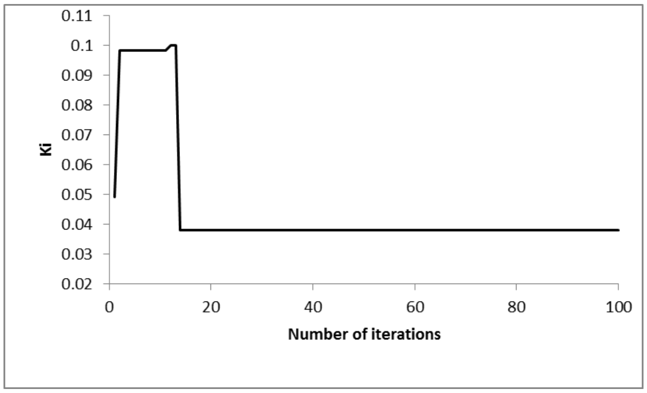


Figure 24: Best solutions of Ki to model 3 for 100 iterations

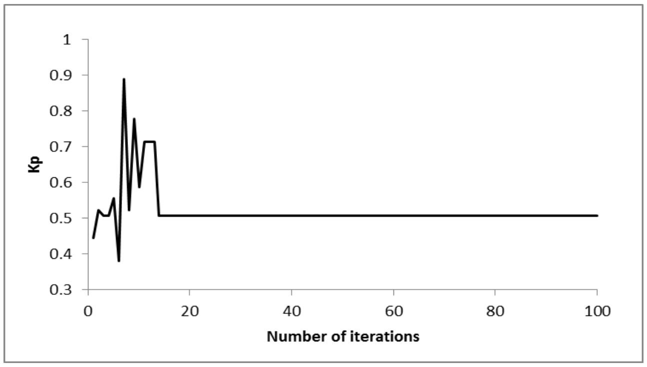


Figure 25: Best solutions of Kp to model 4 for 100 iterations

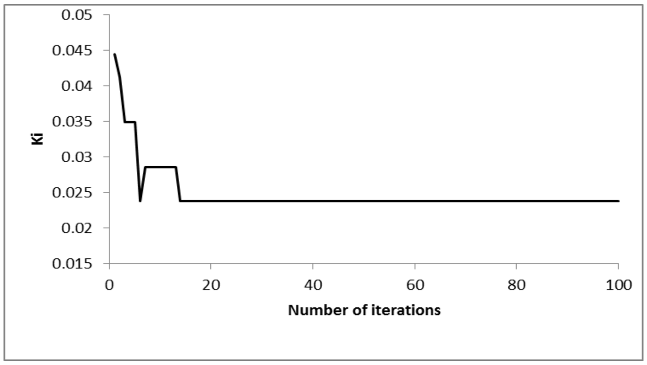


Figure 26: Best solutions of Ki to model 4 for 100 iterations

The PI controller was formed based upon the respective parameters for 100 iterations, and the gbest (global best) solution was selected for the set of parameters which had minimum error. A sketch of the error based on IAE criterion for 100 iterations is given in Figures 27 - 30. It is seen that the error value tends to decrease for a larger number of iterations. As such, the algorithm was restricted to 100 iterations beyond which there was only a negligible improvement. Based on the ACO algorithm for the application of the PI tuning, we get the PI tuning parameters for the model is given in the Table 4.

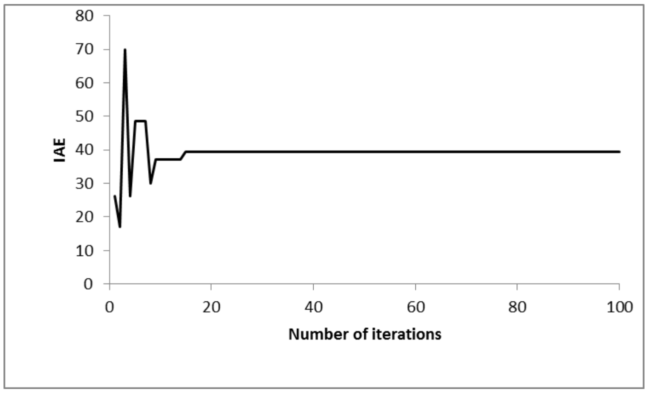


Figure 27: IAE values for 100 iterations of model 1

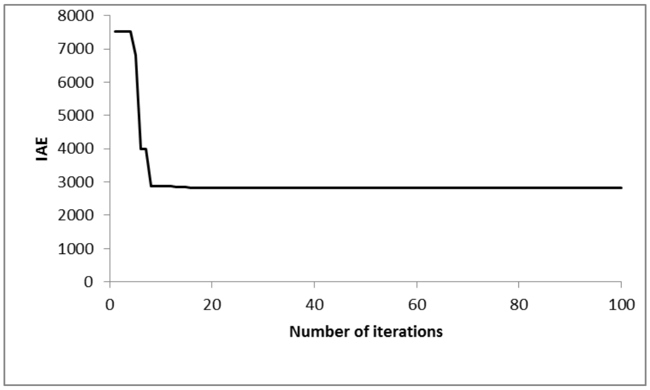


Figure 28: IAE values for 100 iterations of model 2

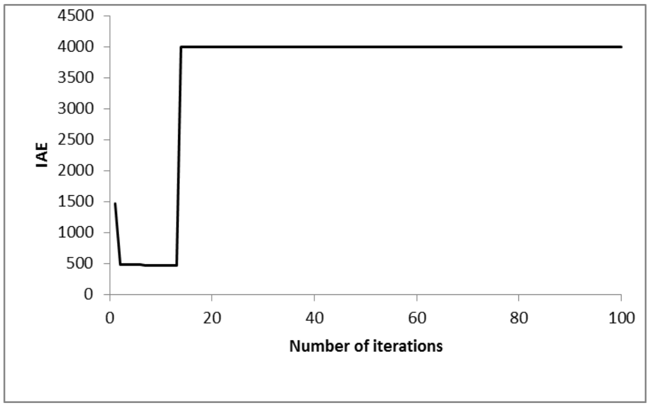


Figure 29: IAE values for 100 iterations of model 3

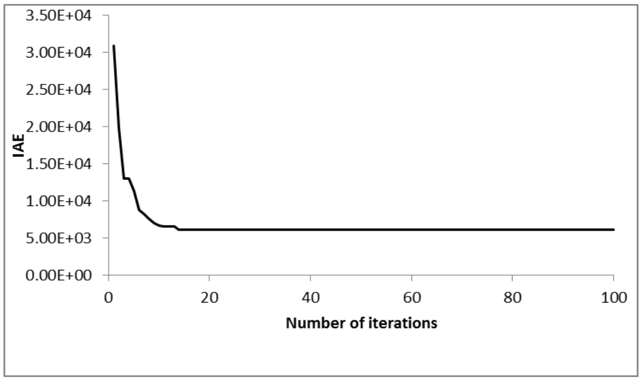


Figure 30: IAE values for 100 iterations of model 4

Table 4: Control parameters for PI controller for the four models got from global best by ACO technique

PARAMETERS	Model1	Model 2	Model 3	Model 4
Kp	0.21805	0.31813	0.4797	0.5502
Ki	0.02606	0.02278	0.0264	0.0252

5. RESULTS AND COMPARISION

After the tuning process is done through traditional methods and proposed techniques, analysis was done for their responses to a unit step input, with the help of real time application for the conical tank. The performance index comparison for the obtained models with the designed controllers is presented in Tables 5-8 for all the four models.

Table 5:Comparison of performance index for Model 1

METHOD	IMC	ACO
IAE	192.3461	145.47
ISE	156.8276	134.07
MSE	0.0786	0.0671
ITAE	2638.9	1258.4

Table 6: Comparison of performance index for Model 2

METHOD	IMC	ACO
IAE	235.8107	201.8154
ISE	208.3621	184.86
MSE	0.0694	0.0616
ITAE	4251.3	2885.4

Table 7: Comparison of performance index for Model 3

METHOD	IMC	ACO
IAE	249.8921	216.21
ISE	221.2638	200.1470
MSE	0.1158	0.1008
ITAE	6237.9	3905.3

Table 8: Comparison of performance index for Model 4

METHOD	IMC	ACO
IAE	305.2351	270.119
ISE	290.908	268.5775
MSE	0.0969	0.0895
ITAE	9469.1	6135.9

5.1 REAL TIME RESPONSE OF THE EXPERIMENTAL SETUP FOR SET POINT CONDITIONS

The parameters designed for the experimental setup were implemented for 4 set points. The real time response of the system was observed by giving set points of 12cm, 22cm and a servo process including the points 32cm and 39 cm. The corresponding variation of level from a reference value of zero was recorded. The outflow valve of the tank was kept partially open and the position was maintained same for the various trails of controller settings. The responses of the conical tank for all the set points with various controller settings are presented in the figures 31-33.

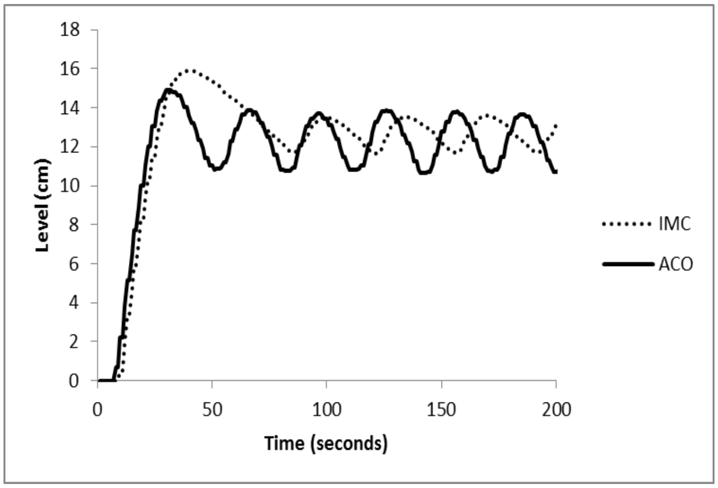


Figure 31: Real time response for a set point of 12 cm.

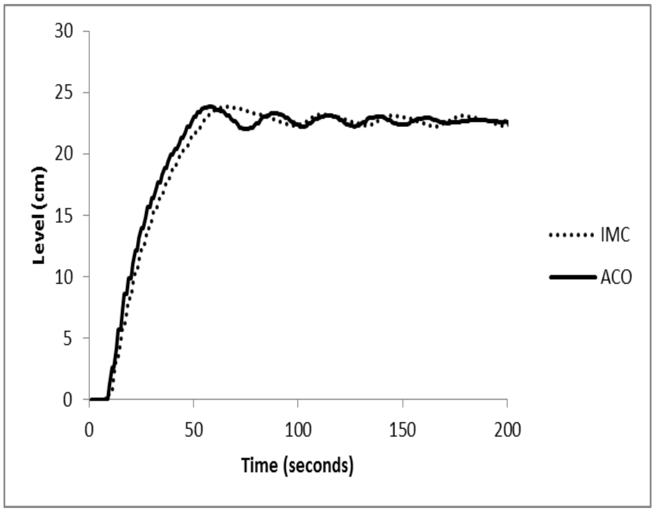


Figure 32: Real time response for a set point of 22 cm.

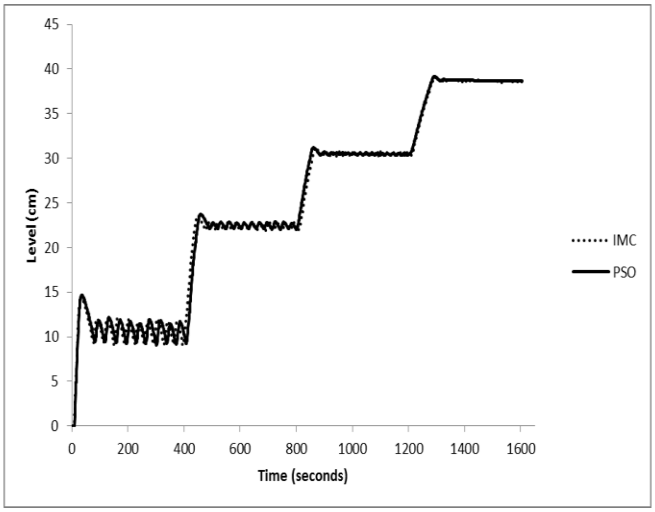


Figure 33: Real time servo response for ACO and IMC

5.2 Robustness Investigation

The PI controllers tuned by the GA based method is compared with the performance index from the four major error criterion techniques of Integral Time of Absolute Error (ITAE), Integral of Absolute Error (IAE), Integral Square of Error (ISE), and Mean Square Error (MSE). Robustness of the controller is defined as its ability to tolerate a certain amount of change in the process parameters without causing the feedback system to go unstable. In order to investigate the robustness of the proposed method in the face of model uncertainties, the model parameters were altered.

Here the values of gain constant K, time constant, τ and delay time τd are deviated by as much as ±15% of nominal values, In the proposed models for the experimental setup, the value of K is incremented by 15% , the value of τ is incremented by 15% and that of τd is reduced by 15%. Thus, we have the models with the proposed uncertainties as, modified model 1 for segment from 0-15 cm as

$$G1(s)_{\text{model 1}} = \frac{3.151e^{-9.044s}}{(4.876s+1)} \qquad \text{Eq(5.1)}$$

Modified model 2 for segment 15-27 cm as

$$G1(s)_{\text{model 2}} = \frac{2.5185e^{-11.883s}}{(10.235s+1)} \qquad \text{Eq(5.2)}$$

Modified model 3 for segment 27-36 cm as

$$G1(s)_{\text{model 3}} = \frac{1.84e^{-13.1835s}}{(14.0185s+1)} \qquad \text{Eq(5.3)}$$

Modified model 4 for segment 36-43 cm as

$$G1(s)_{\text{model 4}} = \frac{1.5640e^{-16.15s}}{(17.8940s+1)} \qquad \text{Eq(5.4)}$$

Table 9: Comparison of performance index for 15% change in Model 1

METHOD	IMC	ACO
IAE	192.3461	130.984
ISE	156.8276	260.64
MSE	0.0786	0.0587
ITAE	2638.9	1095.9

Table 10: Comparison of performance index for 15% change in Model 2

METHOD	IMC	ACO
IAE	235.8107	180.77
ISE	208.3621	162.07
MSE	0.0694	2423.4
ITAE	4251.3	1095.9

Table 11: Comparison of performance index for 15% change in Model 3

METHOD	IMC	ACO
IAE	249.8921	197.24
ISE	221.2638	179.14
MSE	0.1158	0.0908
ITAE	6237.9	3467.2

Table 12: Comparison of performance index for 15% change in Model 4

METHOD	IMC	ACO
IAE	305.2351	252.0237
ISE	290.908	237.0833
MSE	0.0969	0.0790
ITAE	9469.1	5153.1

Table 13: Comparison of time domain specifications for 12cm

Height	IMC	ACO
Over shoot	1.81	1.81
Raise Time	41.3339	37.2207
Delay Time	9	7

Table 14: Comparison of time domain specifications for 22cm

Height	IMC	ACO
Over shoot	3.91	2.91
Raise Time	18.4013	15.9150
Delay Time	9	7

6. CONCLUSION

The developed controller tuning for various set points can be suitably tracked by providing a program which can allow the system to choose that value based on the set point selected. The ACO tuning for model 1 will be used for set points between 0-15 cm, model 2 for set points between 15-27 cm, model 3 for set points between 27-36 cm and model 4 for set points between 36 and 43 cm. The servo response of the system shows its stability over continuous change in set points at regular intervals.

The various results presented prove the betterness of the ACO tuned PI settings than the IMC tuned ones. The simulation responses for the models validated reflect the effectiveness of the GA based controller in terms of performance index. The performance index under the various error criteria for the proposed controller is always less than the IMC tuned controller. Above all, the real time responses confirm the validity of the proposed ACO based tuning for the conical tank.

ACO presents multiple advantages to a designer by operating with a reduced number of design methods to establish the type of the controller, giving a possibility of configuring the dynamic behaviour of the control system with ease, starting the design with a reduced amount of information about the controller (type and allowable range of the parameters), but keeping sight of the behaviour of the control system. These features are illustrated in this work by considering the problem of designing a control system for a plant of a first-order system with a time delay and deriving the possible results.

REFERENCES

[1] Novel Approach to Nonlinear PID Parameter Optimization Using Ant Colony Optimization Algorithm , Duan Hai-bin', Wang Dao-bo2, Yu Xiu-fen3, Journal of Bionic Engineering 3 (2006) 073-078

[2] M Araki: Control systems, Robotics and Automation-Vol II – PID Control –,Kyoto University ,Japan

[3] Kim Dong Hwa and Park Jin Ill: Intelligent PID Controller Tuning of AVR system using GA and PSO: Springer-Verlag Berlin Heidelberg: ICIC 2005, Part II, LNCS 3645, pp 366-375.(2005).

[4] Ziegler, G. and Nichols, N. B, 1942.Optimum settings for automatic controllers, Trans. ASME, 64,759-768.

[5] K.J.Astrom, T.Hagglund, The future of PIDcontrol, Control Eng.Pract.9(11)(2001)1163–1175.

[6] Astrom, K J.;. Hagglund .T,1984, Automatic tuning of simple regulators with specifications on phase and amplitude margins, Automatica, 20,645-651.

[7] G.H Cohen and G.A Coon: Theoretical Consideration of Retarded Control , Trans ASME 75,pp.827/834,(1953)

[8] Haber , Rodolf., Toro1, Raúl M. and. Alique, José R, ' Using Simulated Annealing for Optimal Tuning of a PID Controller for Time-Delay Systems. An Application to a High-Performance Drilling Process', Springer.

[9] Colorni A. Dorigo M, Maniezzo V. Distributed optimization by ant colonics. *Proceedings of the First European Conference on Artjficial Life.*Elsevier Publishing, paris. 1992, 134-142.

[10] Bonabeau E. Dorig M, Theraulaz *G.* Inspiration for optimization from social insect behavior. *Nature*, 2000, 406.

[11] Katja V, Ann N. Colonies of learning automata. IEEE Transactions on Systems, Man, and Cybernetics-Part B, 2002,32,772-780.

[12] Dorigo M, Gambardella L M. Ant colony system: A cooperative learning approach to the traveling salesman problem. IEEE Transactions on Evolutionary Computation,1997,1,53-66.

[13] James M, Marcus R. Anti-pheromone as a tool for better exploration of search space. Proceedings of the 3rd International Workshop ANTS, Brussels, 2002, 100-1 10.

[14] Sundaresan, K. R., Krishnaswamy, R. R., Estimation of time delay, time constant parameters in Time, Frequency and Laplace Domains, *Journal of Chemical Engineering*., 56, 1978, p. 257.

[15] Dorigo M, Maniezzo V, Colomi A. Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions*on *Systems, Man, and Cybernetics-Part B*, 1996,26,29-41

[16] Duan H B, Wang D B. A novel improved ant colony algorithm with fast global optimization and its simulation.*Information and Control,* 2004,33,241-244, (in Chinese).

[17] Duan H B, Wang D B, Zhu J Q, Huang X H. Development on ant colony algorithm theory and its application. *Control and Decision,* 2004, 19, 1321-1326, 1340, (in Chinese).

[18] P. P. Grass´e, *Les Insectes Dans Leur Univers*. Paris, France: Ed. Du Palais de la d´ecouverte, 1946.

[19] K. Socha, “ACO for continuous and mixed-variable optimization,” in *Ant Colony Optimization and Swarm Intelligence, 4th International Workshop, ANTS 2004*, ser. LNCS, M. Dorigo *et al.*, Eds., vol. 3172. Springer Verlag, 2004, pp. 25–36.

[20] K. Socha and M. Dorigo, “Ant colony optimization for continuous domains,” *European Journal of Operational Research*, 2006,