Applied Mathematical Modelling 37 (2013) 3829–3847

Contents lists available at SciVerse ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

Real-coded Genetic Algorithm for system identification and tuning of a modified Model Reference Adaptive Controller for a hybrid tank system

K. Asan Mohideen [a,*], G. Saravanakumar [a], K. Valarmathi [a], D. Devaraj [a], T.K. Radhakrishnan [b]

[a] Kalasalingam University, Anandnagar, Krishnankoil, Virudhunagar District, Tamil Nadu 626 126, India
[b] National Institute of Technology, Thiruchirappalli, Tamil Nadu 620 015, India

ARTICLE INFO

Article history:
Received 23 November 2011
Received in revised form 29 June 2012
Accepted 15 August 2012
Available online 7 September 2012

Keywords:
Nonlinear process control
System identification
Model Reference Adaptive Control
Realcoded Genetic Algorithm

ABSTRACT

Modeling and controlling of level process is one of the most common problems in the process industry. As the level process is nonlinear, Model Reference Adaptive Control (MRAC) strategy is employed in this paper. To design an MRAC with equally good transient and steady state performance is a challenging task. The main objective of this paper is to design an MRAC with very good steady-state and transient performance for a nonlinear process such as the hybrid tank process. A modification to the MRAC scheme is proposed in this study. Real-coded Genetic Algorithm (RGA) is used to tune off-line the controller parameters. Three different versions of MRAC and also a Proportional Integral Derivative (PID) controller are employed, and their performances are compared by using MATLAB. Input–output data of a coupled tank setup of the hybrid tank process are obtained by using Lab VIEW and a system identification procedure is carried out. The accuracy of the resultant model is further improved by parameter tuning using RGA. The simulation results shows that the proposed controller gives better transient performance than the well-designed PID controller or the MRAC does; while giving equally good steady-state performance. It is concluded that the proposed controllers can be used to achieve very good transient and steady state performance during the control of any nonlinear process.

1. Introduction

Regulating liquid level in a tank has been a basic control problem in process industries for a long time. The PID controllers have been used for this purpose traditionally. Cohen–Coon and Ziegler–Nichols methods [1] have been widely used to tune a PID controller [2]. In these methods, the values of proportional gain, integral time constant and derivative time constant are determined for an operating point around which the process can be considered linear. But, when the operating conditions change, PID controller parameters also need to be tuned. Hence, using these methods would result in sub-optimal tuning of the controllers for non-linear and dynamic systems and it needs the intervention of an operator. The urge to design better controllers brings out the need to understand the characteristics of the process as well as possible. This can be achieved by the application of system identification procedure. System identification involves building mathematical models of a dynamic system based on a set of measured input and output data samples. System identification helps in designing a controller for a process in a more effective manner than would be the case without identification.

In conventional identification methods such as least squares [3] and maximum likelihood method [2], a model structure is selected and the parameters of the model are calculated by optimizing an objective function. These methods hold good

* Corresponding author.
E-mail addresses: asan4uall@gmail.com (K. Asan Mohideen), saravana.control@gmail.com (G. Saravanakumar), krvalarmathi@yahoo.co.in (K. Valarmathi), deva230@yahoo.com (D. Devaraj), radha@nitt.edu (T.K. Radhakrishnan).

0307-904X/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.apm.2012.08.019

only when the process is operated at about the selected operating point since the process can be considered to be linear around the operating point. But, for the systems with considerable amount of non-linearity, the conventional parameter identification methods fail to provide globally-optimum solution. Further, they require huge amount of input–output data from the system to be identified. There are a few approaches for the identification of non-linear systems like Nonlinear Least Squares, Voltera series, Weiner series, Wavelets [4], and soft computing approaches like Neural Network (NN), Fuzzy Logic (FL) and Genetic Algorithm (GA) [5]. The difficulties with classical approaches can be overcome by the use of the afore-mentioned intelligent techniques and their hybrids for the purpose of identification. NN based approach has been applied for system identification of a pH process by Valarmathi et al. [6]. Kristinsson and Dumont [7] used GA to estimate the location of poles and zeroes of a transfer function and then used this estimate to design a discrete time pole placement adaptive controller. Jiang and Wang [8] used binary-coded GA to estimate the system parameters for a class of nonlinear systems. Boroomand and Menhaj [9] proposed Fractional Dynamic Neural Networks (FDNNs) by means of the Lyapunov-like analysis for nonlinear system approximation. This paper proposes GA for the parameter tuning of the identified model of a hybrid tank system in order to improve the accuracy of the identified model as much as possible.

To have a good performance in spite of environment induced changes in the process set up, the controller must possess the ability to adapt to changes in plant dynamics. Adaptive control methods like Gain Scheduling, Dual Control, Model Reference Adaptive Control (MRAC) and Self Tuning Regulator (STR) can be used to improve the performance of a controller [10]. Gain scheduling suffers from the drawback of being an open loop compensation technique. There is no feedback to make necessary changes in the values of the controller parameters in the event of the schedule being incorrect [11]. The STR estimates the process parameters to adapt the controller to its dynamic changes. The problem with this approach is that small model error can lead to large changes in parameters resulting in possible oscillation of process variables. Dual Control is too complicated to be used for practical problems [11]. Goodwin [12] established global convergence and asymptotic properties of a direct adaptive controller for continuous time stochastic linear systems. Tsai et al. [13] used MRAC to control temperature in a variable frequency oil-cooling machine. Liu and Hsu [14] proposed adaptive back stepping control and MRAC to improve the performance of a sensor-less direct torque control synchronous reluctance motor drive system. Kamalasadan et al. [15] introduced a new concept of Intelligent Supervisory Loop in a Neural Network (NN) based intelligent adaptive controller for the control of dynamic systems and the author used an on-line Radial Basis Function NN in parallel with an MRAC. Miller and Mansouri [16] proposed a scheme in which simultaneous probing, estimation and control is carried out to give better noise performance in MRAC. The scheme used linear predictive control in an MRAC set up. In a basic Model Reference Adaptive System (MRAS), the process output takes a finite amount of time to converge with the output of the reference model. Improving the transient performance of the standard MRAC has been the point of research for quite some time. Datta and Ioannou [17] proposed a modified traditional MRAC that showed improved transient and steady state performance. Mean square tracking error criterion and the L_∞ tracking error bound criterion were used to assess performance in the ideal and non-ideal situations. Miller and Davison [18] proposed a controller comprising an LTI compensator together with a switching mechanism to adjust the compensator parameters so that an arbitrarily good transient and steady-state response is provided for a single-input single-output linear time-invariant plant. The hybrid tank process considered in this paper is a classic example of a highly nonlinear system [19]. In this paper, a modification to the MRAC scheme is proposed to obtain very good transient and steady-state performance during the application of MRAC for the purpose of controlling liquid level in the coupled tank system. The proposed controller is referred to as Modified MRAC throughout the paper.

In order to incorporate intelligence in the controller one could go for soft computing methods like NN, FL, GA, their hybrid structures and other evolutionary algorithms like Particle Swarm Optimization (PSO), Ant Colony, Bacterial Foraging etc. Chang [20] applied Real-coded GA (RGA) for system identification and off-line tuning of PID controller for a system whose structure is assumed to be known previously. Chao and Teng [21] proposed a two-stage tuning method for a PD-like self-tuning fuzzy controller which automatically detects the operating ranges of input variables and then adjusts the scaling factors. Hu and Mann [22] proposed a conservative design strategy for realizing a guaranteed-PID-performance fuzzy controller in which GA is also employed for optimization. Valarmathi et al. [23] made use of PSO for system identification and tuning of Proportional-Integral (PI) controller in a pH process.

In this paper, performance of the Modified MRAC is further improved by using RGA to fine tune some controller parameters off-line. The Modified MRAC which uses RGA is referred to as GA based Modified MRAC in the paper. As the search space is large, application of RGA would more likely give optimal or near-optimal solution.

2. Hybrid tank process

Fig. 1 shows the schematic representation of a hybrid tank system. The hybrid tank system consists of two cylindrical tanks, inter connected by two coupling channels at different heights, both with drain valves to common reservoir situated below. A variable-area valve in these channels is used to vary the interaction between the tanks. Water from a reservoir is pumped by a variable speed pump to tank1 through a control valve. F1 is the flow rate of the influent stream to tank1 and is measured by a turbine flow meter. Water from tank1 flows to tank2 through the coupling channels, and then it finally flows out of tank2. Level of water in tank2 (L2) is measured by a DPT. The flow rate F1 and level L2 are acquired by a Data Acquisition Card NI USB 6008 in Lab VIEW environment. Command signal to open the control valve to vary F1 is given through the

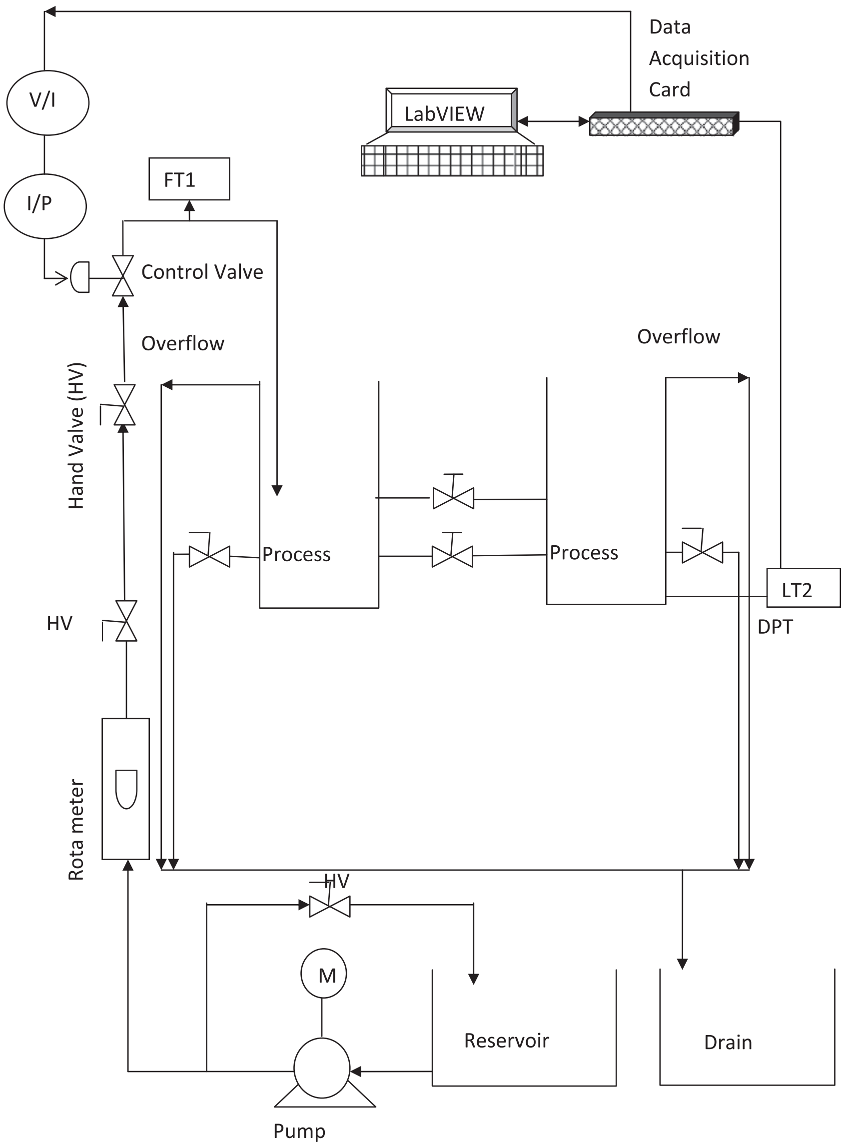

Fig. 1. Coupled tank setup.

Lab VIEW program. The flow rate F1 is selected as the manipulated variable, and level L2 of the liquid in the second tank, tank2, is the output or the controlled variable.

3. System identification

System identification is the process of building mathematical model of dynamic systems from the observed input–output data. The process models describe a system transfer function in terms of zeros, poles, integration and delay terms. In process models, number of non-zero poles is represented by P_n, where P stands for pole and n represents the number of non-zero poles. P_n is followed by Z for zero, I for integration, D for delay and U representing under-damped behavior. A P1D structure models the system as a first order system with delay as shown in (1)

$$G(s) = \frac{Ke^{-T_d s}}{(1 + sT_{p1})}, \tag{1}$$

where the steady-state-gain of the process is K, T_d is the time delay and T_{p1} is the process time constant.

Eq. (2) represents a P2D model that has two real poles placed at $s = -1/T_{p1}$ and $s = -1/T_{p2}$. Eq. (3) represents a P2DU model which is essentially a second order system with either a pair of complex poles or two real poles depending upon the damping.

$$G(s) = \frac{Ke^{-T_d s}}{((1 + sT_{p1})(1 + sT_{p2}))}, \tag{2}$$

$$G(s) = \frac{Ke^{-T_d s}}{\left(1 + 2\zeta T_w s + T_w^2 s^2\right)}, \tag{3}$$

where T_{p1} and T_{p2} are the time constants corresponding to real poles, T_w is the time constant of the second order system shown in (3) and ζ is the damping factor. The parameters of the best model in terms of fit are further fine-tuned using RGA in order to increase the accuracy of the identified model.

4. Model reference adaptive system

Nonlinear and non-stationary nature of a system warrants the use of adaptive controllers such as MRAC and STR [24]. The Model Reference Adaptive System (MRAS) can be regarded as an adaptive servo system in which the desired performance is expressed in terms of a reference model, which gives the desired response to a command signal [11]. Thus, the response of the reference model acts as the set point for the process in a normal feedback loop. The system has an ordinary feedback loop composed of the process and the controller and another feedback loop that changes the controller parameters as shown in Fig. 2. It depicts a general MRAS in terms of blocks. The parameters of the controller are changed on the basis of error (e), which is the difference between the output (y) of the system and the output (y_m) of the reference model. $G(s)$ and $G_M(s)$ are the transfer functions of the process and the reference model respectively. The premise of MRAC design is that the process characteristics are not absolutely known. The goal of the designer is to make the process approach the model. This is achieved by adjusting the process input value u in such a way that the output y follows the model output y_m as closely as possible. The mechanism for adjusting the parameters in a model-reference adaptive system can be obtained in two ways: by using a gradient method or by applying stability theory [11].

The adaptation law uses the error (e), the process output (y), and the input signal (u_c), to vary the parameters of the control system. These parameters are varied so as to minimize the error. In this paper, the parameters are adjusted so that the cost function $J(\theta) = 1/2^*(e^2)$ is minimized. The MIT rule is the original approach to Model Reference Adaptive Control (MRAC).

5. Tuning of controller

A Model Reference Adaptive Controller, which uses the famous MIT rule for parameter tuning is designed and implemented as shown in Fig. 2. Here, the controller has two parameters θ_1 and θ_2. Output ‘u’ of the controller is calculated from the parameters, the command signal u_c and the output y of the process, as shown in (4)

$$u = \theta_1 u_c - \theta_2 y. \tag{4}$$

If the cost function is taken as $J(\theta) = 1/2^*(e \wedge 2)$, change in the value of parameters of the controller with respect to time as per the MIT rule is given in (5)

$$\frac{d\theta}{dt} = -\gamma \frac{\partial J}{\partial \theta} = -\gamma e \frac{\partial e}{\partial \theta}, \tag{5}$$

where $\partial e/\partial \theta$ is the sensitivity derivative of the system, γ is the adaptation gain, e is the tracking error ($y{-}y_m$) and y is the model output.

The speed of convergence depends on the value of the adaptation gain. An MRAS with too small a value of γ gives a stable response, but it takes a long time for the output y to converge with y_m whereas too large a value of γ makes the output y to

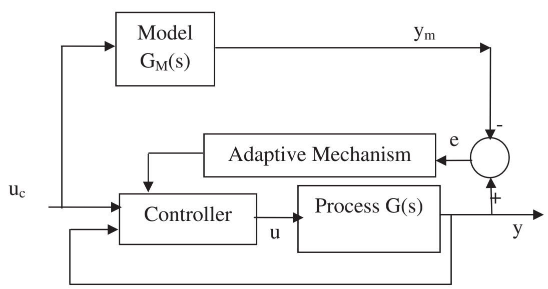

Fig. 2. Model Reference Adaptive Control.

oscillate. Hence, a trade-off is always required between the stability and the speed of convergence while selecting the value of γ.

In order to improve the transient performance a modification to the MRAC scheme is proposed in this paper. In the proposed controller, the controller output u is modified as given by Eq. (6). Here, a Proportional Integral Derivative (PID) controller is also employed along with the MRAC controller and the resultant controller is called 'Modified MRAC' in this paper. Parallel form of PID controller is used here.

$$u = \theta_1 u_c - \theta_2 y - \left(K_p e + K_i \int e dt + K_d \frac{de}{dt}\right), \tag{6}$$

where K_p is the proportional gain, K_i is the integral gain and K_d is the differential gain. Initial tuning of PID controller is carried out by using PID tuner available in the Control System Tool Box of MATLAB. This Modified MRAC considerably improves the performance criteria as established by the results.

In the quest for optimizing the performance in terms of rise time, settling time and mean square error (MSE), the RGA is called upon to fine tune the parameters K_p, K_i and K_d of the PID controller which is present in the Modified MRAC and this controller is referred to as 'GA based Modified MRAC'.

A well-designed PID controller is also implemented for the process model. As MRAC controller and its modifications are used in this paper, the output of the reference model that is used in all the MRAC controllers in this paper is given as the reference signal to the PID controller loop.

6. Review of Genetic Algorithm

Genetic Algorithm [25] is a global search and optimization method that makes use of Darwinian Theory of evolution. The GA searches for the optimal solution from many directions in the search space as opposed to classical search algorithms which use local derivatives and inch closer towards the optimal solution in a single direction. The classical approaches which use gradient method have a relatively high probability of getting trapped in local minima whereas GA escapes local minima by the virtue of mutation which alters genetic information of an offspring randomly with a very small probability.

At any given time GA works with a population of solutions, and each member of the population is assigned a fitness value which is derived from the objective function of the given problem. The members which represent better solutions are assigned higher values of fitness ensuring their survival over successive generations. Initial population is selected randomly and the genetic operators such as reproduction, crossover and mutation are applied to generate successive generations of populations. These successive generations would yield better solutions eventually approaching the optimal solution to the problem.

GA is quite successful in locating the regions of the search space containing the global optimum, but not the global optimum itself [26]. To avoid getting stuck in local minima the following steps can be followed:

- The number of generations and the population size can be increased as long as there is considerable improvement in the best fitness value. When the improvement in the value of best fitness is very small and the average fitness converge to the best fitness value for sufficiently large time, this step may be stopped.
- Mutation rate can also be increased in order to find better solutions.
- The ranges for the individual parameters can be reduced such that the new search space still contains the minimum obtained in the previous step.
- Random restart can be carried out to see if there is any further improvement.

If the above algorithm is followed and the solution remains the same, it can be said that an optimal or near-optimal solution has been found. By this procedure, optimal or near-optimal solutions can be ensured.

The optimization variables are encoded into binary strings in binary-coded GA. This GA has some difficulties such as time-consuming encoding and decoding in each step, large memory requirement and loss of precision due to quantization error [6]. In using RGA the above said difficulties are not faced as the variables can be represented as floating point numbers. This paper uses RGA in system identification as well as controller tuning.

7. Genetic algorithm implementation

7.1. GA in system identification

At first a suitable model of the process is obtained using the standard system identification procedure and then the parameters of the identified model are further fine-tuned by RGA as shown in Fig. 3. The simulation output, $\hat{y}(k)$, of the model is compared with the actual output of the process, $y(k)$, and the error is given to the RGA in the form of mean square error (MSE) as given in (7).

$$\text{MSE} = \left(\frac{1}{N}\right)\sum_{k=0}^{N}(y(k) - \hat{y}(k))^2, \tag{7}$$

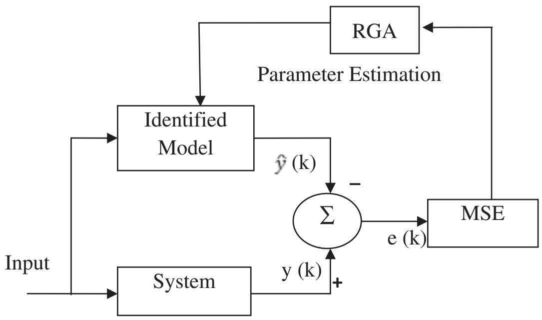

Fig. 3. Parameter estimation using RGA.

where k is the time instant.

The GA finds an optimal solution for the model parameters so that the MSE is minimized. The fitness function F to be maximized by GA is formulated as

$$F = \frac{K}{(1 + \text{MSE})} \tag{8}$$

As 'F' is maximized, the value of MSE is minimized. For this particular case, the value of K is selected to be 10, which is included in the function to amplify the value of (1/(1 + MSE)). This would ensure that the fitness value of the chromosome is in a wide range. The denominator has a '1' added with MSE so that denominator can never be zero.

7.2. GA in controller tuning

The setup in which RGA is used to find the optimal values of PID controller parameters which is part of the Modified MRAC is shown in Fig. 4. Command signal is applied to reference model and the controller. Error, 'e' is obtained by subtracting the output y_m of the reference model from the output y of the plant. A PID controller is also employed to work along with the MRAC. Error and the plant output are given to GA, and it returns the optimal values of controller parameters. Controller parameters are updated in each generation until the stopping criterion is met.

The objective function to be minimized is the MSE given by

$$\text{MSE} = \left(\frac{1}{N}\right)\sum_{k=0}^{N}(y(k) - y_m(k))^2, \tag{9}$$

where $y(k)$ is the output of the model at the instant k and $y_m(k)$ is the output of the reference model at the instant k.

The GA maximizes the fitness function given by (10)

$$F = \frac{1}{(1 + \text{MSE})} \tag{10}$$

Maximizing 'F' in effect minimizes the objective function MSE. In the denominator, a '1' is added to make sure that denominator value never becomes zero. Tournament selection is used in this application.

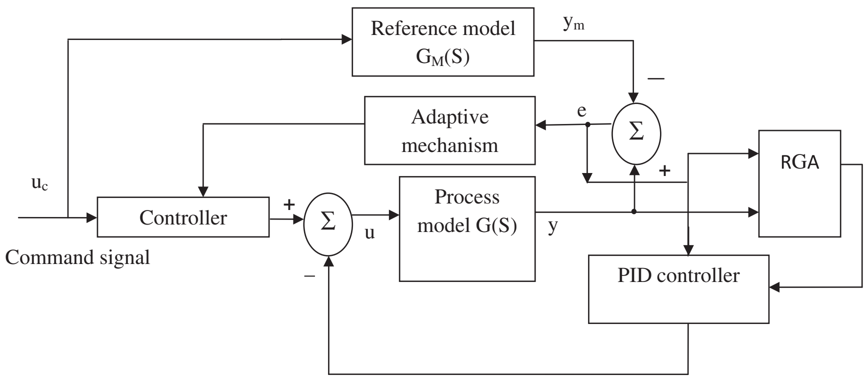

Fig. 4. Controller tuning using RGA.

8. Results and discussion

8.1. Experiment and data acquisition

The coupled tank setup shown in Fig. 1 is implemented and the photograph is shown in Fig. 5. Two variable speed pumps with a capacity to displace 1300 lph pump water through control valves. The pumps are driven by induction motors of power rating of 125 W. The control valves are provided at the inflow sides. The valves have equal percentage characteristics and are normally at fully closed condition (Air-to-open). When the valves are fully open and the pump is run at full speed, Flow F1 can reach a maximum of 880 lph. Each tank has a heater consisting of three heating elements and their power supply is controlled by TRIAC. The maximum current rating of the heaters is 35 A. The system permits one to use the tanks either separately or with some interaction. The water level in each tank is made visible through the transparent glass that stands beside the tank. The level of liquid in each tank is 60 cm at the maximum.

Turbine flow meters and rotameters are provided to measure and indicate the volumetric flow rates respectively. Turbine flow meters are energized by a 24 V DC, 100 mA power supply, and the output range is 4–20 mA. Its measurement range is 0–1000 lph. The range of the rotameter is 40–880 lph. Differential Pressure Transducer (DPT) measures the pressure exerted by the liquid column in the tank thereby indirectly measuring level of the liquid in the tank. The DPT used here can measure pressure up to 50 kPa. Resistance Temperature Detectors (RTDs) are employed to measure temperature of the inlet and the liquid in the tank. PT 100 is the RTD model used here. Each tank has a motor driven stirrer to make sure that the heat supplied by the heaters is distributed uniformly.

8.2. Step response analysis

First, the height of the lower coupling channel is made the reference level by having water up to that height in both tanks. Then, flow F1 is increased to 615 lph and level L2 reaches a steady state of 18.5 cm. Once the steady state is reached, in an open loop environment, a step signal between 615 lph and 740 lph is applied to the flow rate F1 and the level of the liquid in the tank 2 is then measured and plotted as shown in Fig. 6. Level L2 reaches a steady-state value of 41 cm. The process response is like the response of a second order system given below.

$$G(s) = \frac{K_p}{(\tau^2 s^2 + 2\zeta\tau s + 1)} e^{-T_d s}, \tag{11}$$

where K_p is the gain of the process, τ is the time constant, ζ is the damping factor and T_d is the time delay. By using system identification tool box the following estimates of the process parameters are obtained. The estimates are: $K_p = 0.18$ cm/lph; $\tau = 142.17$ s; $\zeta = 2.3227$ and $T_d = 0$ s. Process settling time is approximately 2500 s.

8.3. Experiment with PRBS signal

A binary input signal such as pseudo-random binary signal (PRBS) is persistently exciting as it contains frequencies in a wide range. When the process is excited by such signals maximum amount of information possible about the process [2] can be obtained. As the signal has positive and negative excursions it can be used to test the nonlinearities and different gains. The selection of sampling time is crucial. If it is too short the poles group around $z = 1$ whereas if it is too long fundamental information about the process dynamic behavior will not be captured [27]. An adequate sampling time can be determined by dividing the process settling time by 10 or 20 approximately [2]. In this study, the amplitude of the PRBS signal is varied from 615 to 740 lph. The operating point of the process corresponds to 677 lph and 30 cm.

Fig. 5. Photograph of the experimental setup of hybrid tank process.

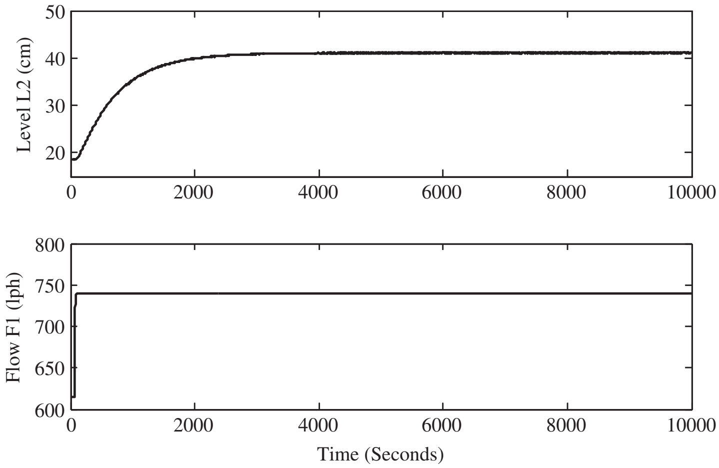

Fig. 6. Step response of the coupled tank level process.

The input–output data of the experiment is shown in Fig. 7. A total of 720 samples were collected with a sample time of 50 s and the samples from 1 to 400 are used for estimation and the rest for validation of the model. The input–output data of the experiment were pre-processed before system identification procedure is applied. Pre-processing removes deficiencies in the data set that can affect estimation and it included the removal of mean and linear trend in both input and output data.

8.4. Process model selection

From this point onwards, throughout this paper zero flow represents 677 lph, zero level represents 30 cm and the time is scaled. The order of the model is selected by using MATLAB system identification tool box using 400 sample data. The prediction error method (PEM) is used to identify the model. The hybrid tank process has two tanks and thus the minimum order of the system is two. This is supported by the results of both step test and the PRBS experiment. The parameters of the process models have obvious physical significance such as gain of the process and time constants. One can easily relate these parameters with the physical structure of the process. Fig. 8 shows the comparison between the simulated outputs of the second order process models with the estimation data set.

Considering the fact that the coupled tank process has two interacting tanks the P2DU model is selected. The transfer function of the identified P2DU model of the coupled tank system considered in this paper is given by

$$G_p(s) = \frac{0.18024}{(15.9417s^2 + 13.9976s + 1)} \tag{12}$$

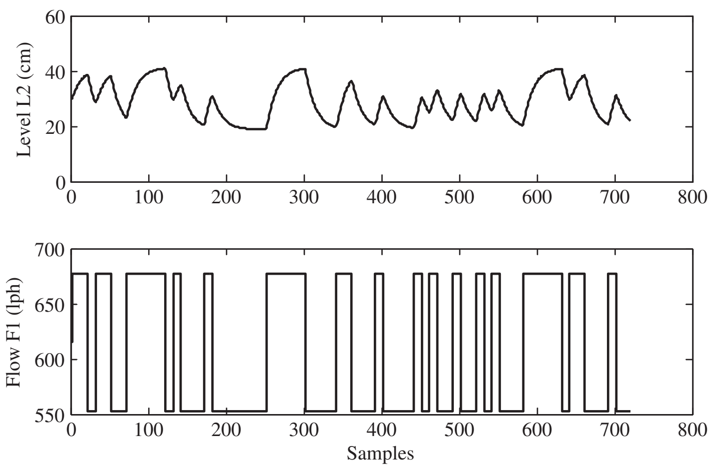

Fig. 7. Input–output data of binary input experiment.

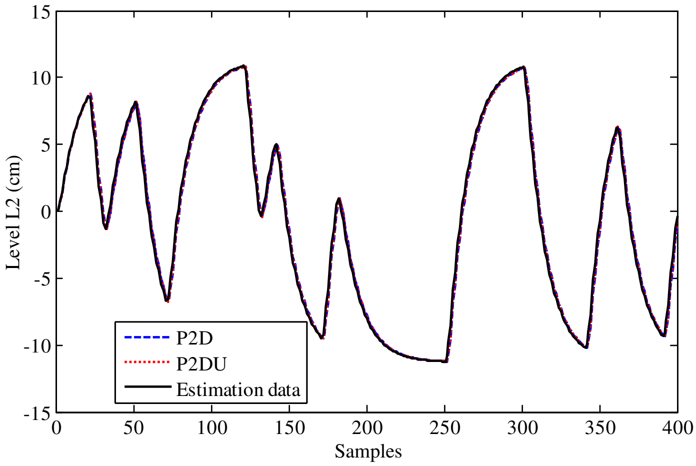

Fig. 8. Comparison of estimation data and model outputs.

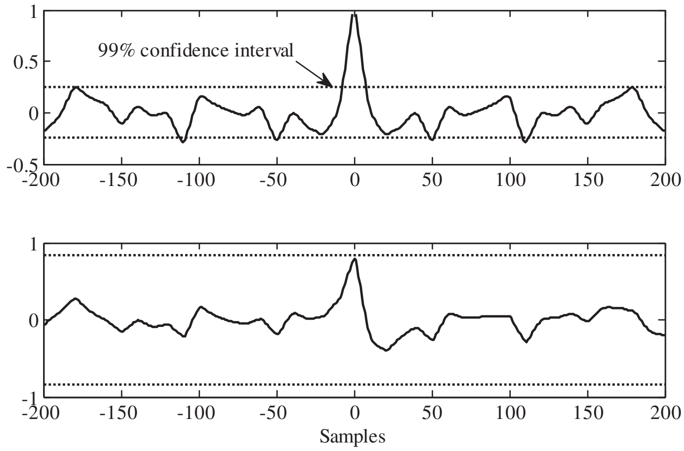

Fig. 9. Residual analysis of the P2DU model.

8.4.1. Model validation

The cross-validation procedure is the best to validate models [2]. In the procedure the estimated P2DU model is applied to a data set different from the one used for estimation. In this experiment, a fit % of 92.14 is obtained. Model validation can also be done by applying residual analysis and correlation analysis on the experimental data.

The residual analysis gives the autocorrelation of the output residuals and the cross correlation between the input to the process and the residuals of the output. If both the correlation functions lie in between the confidence intervals it can be said that the model has captured the dynamics of the process. The residual analysis of the P2DU model is shown in Fig. 9. The upper half of the figure shows the autocorrelation of the output residuals and the lower half shows the cross-correlation between the process input and the output residuals. Both the signals lie in between the 99% confidence intervals. Hence the model can be considered valid.

The impulse response of the coupled tank process is estimated by using correlation analysis of the experimental data and is shown in Fig. 10. The response of the process is that of an over-damped process without delay. The step response analysis also resulted in zero delay. The step response of the coupled tank process is also calculated by using correlation analysis and the same is shown in Fig. 11. The response is slow initially, then becomes fast and then becomes slow again before settling. It is typical of a high order process. The response has no delay and the steady-state gain is 0.18. All these results ratify those obtained through the previous tests.

Fig. 12 shows the comparison between the simulated output of the model (P2DU) and the experimental output for the validation data set. The P2DU model agrees well with the validation data also. Thus, the model P2DU of the coupled tank process is said to be validated.

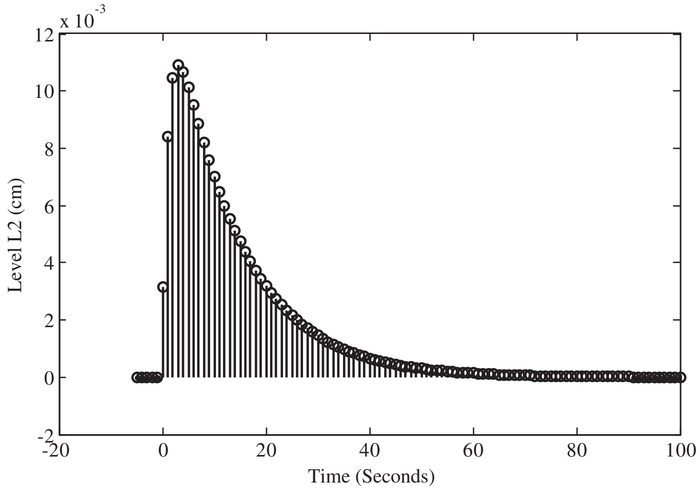

Fig. 10. Impulse response estimated by using correlation analysis.

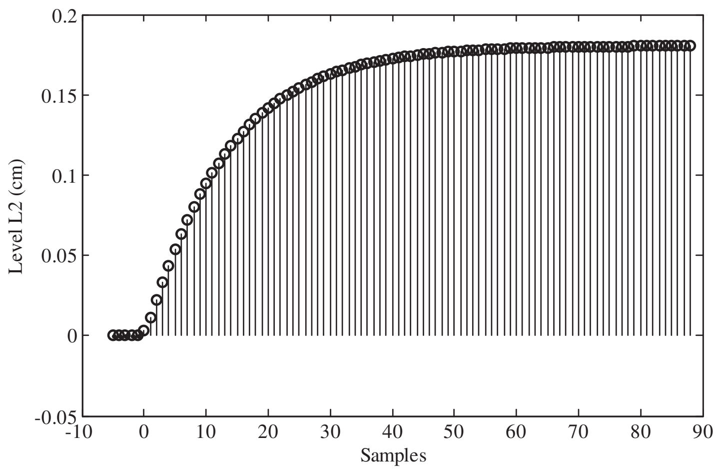

Fig. 11. Step response estimated by using correlation analysis.

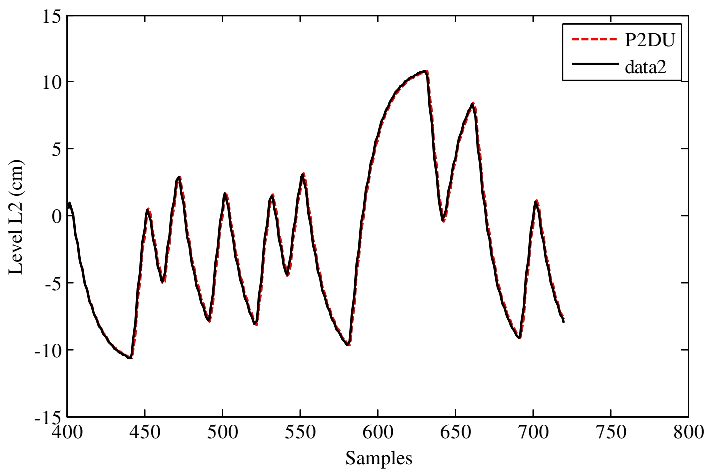

Fig. 12. Comparison of validation data and model output.

Table 1
Parameter estimates by using RGA in different iterations.

N_g	N_p	P_c	P_m	Range			Fitness	Estimate		
				a	*b*	*c*		*A*	*b*	*c*
50	50	0.8	0.01	[0 0.3]	[0 50]	[0 50]	542	0.1768	13	14
60	50	0.8	0.04	[0 0.3]	[0 50]	[0 50]	1129	0.1812	12	14
80	50	0.8	0.02	[0 0.3]	[0 50]	[0 50]	1140	0.1813	12	14
80	50	0.8	0.04	[0.15 0.3]	[0 25]	[0 25]	841	0.184	11	14
80	50	0.75	0.02	[0.15 0.3]	[0 25]	[0 25]	1192	0.1829	11	14
100	100	0.75	0.01	[0.15 0.3]	[0 25]	[0 25]	1218	0.1816	12	14

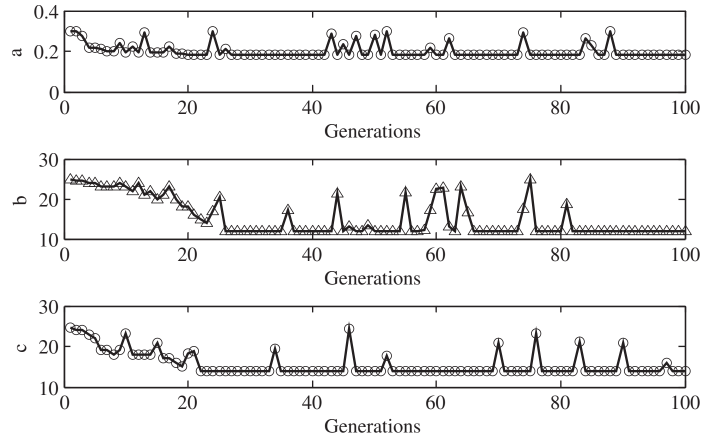

Fig. 13. Convergence of model parameters.

8.4.2. Application of RGA in system identification

In order to fine-tune the parameters of the model further, RGA is used and a new set of values are obtained for the model parameters. The software for the RGA is written in MATLAB. In this work, the parameters of the model are represented by the variables *a*, *b* and *c* where a represents gain '*K*', *b* represents 'τ^2' and *c* represents '$2\tau\zeta$' terms of a second order transfer function. Each variable is represented by a six digit floating point number. Here, the stopping criterion used is the number of generations. The best estimate of the variables is obtained with the following design parameters.

Number of generations (N_g): 100
Population size (N_p): 100
Crossover probability (P_c): 0.75
Mutation probability (P_m): 0.02

The optimality of the estimate is ensured by following the steps explained in section 6. The parameter estimates at various stages are shown in Table 1.

Fig. 13 shows the optimal values for the model parameters *a*, *b* and *c* in each generation using RGA. Fig. 14 shows the convergence of the best fitness and the average fitness. In the thirtieth generation itself both values have come close to each other. Since then, the average fitness stays very close to the best fitness. After 30 generations, the best fitness does not change with more generations; thus indicating optimal or near optimal solution for the parameter values. The resultant model is called P2DU-GA. Simulated output of the P2DU-GA model is obtained and compared with the experimental data. Its fit percentage with the estimation data and validation data are determined. Fig. 15 shows the comparison among the simulated outputs of P2DU and P2DU-GA, and the estimation data. Fig. 16 shows the comparison among the simulated outputs of the two afore-mentioned models and the validation data and Table 2 compares the parameters and fit percentages of the models P2DU and P2DU-GA. These results establish the effectiveness of the application of RGA in the system identification procedure.

The transfer function of the identified P2DU-GA model of the coupled tank system is given by

$$G_{ga}(s) = \frac{0.1816}{12s^2 + 14s + 1}. \qquad (13)$$

The coupled tank setup in the hybrid tank process is identified as an over-damped second order system with zero delay. The damping factor is 2.0207 and the scaled value of the time constant of the process is 3.4641 s. The actual value of the process time constant is obtained by multiplying the scaled value by the time scale used (=50) and the value is 173.21 s.

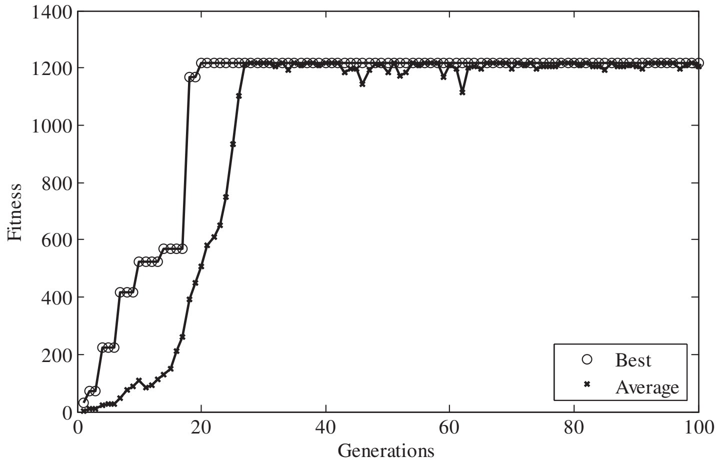

Fig. 14. Convergence of fitness.

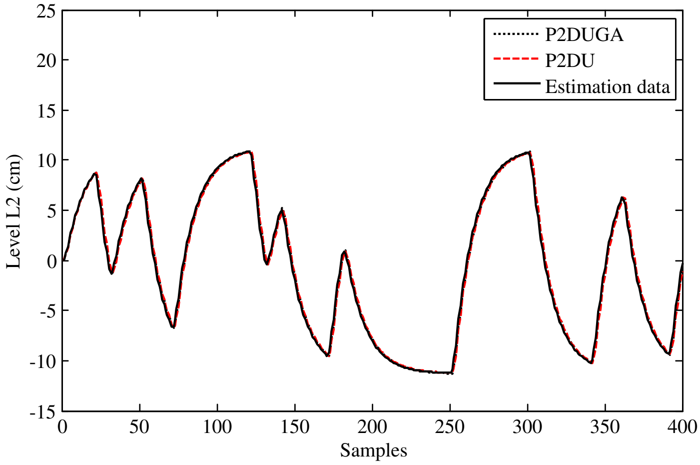

Fig. 15. Comparison of estimation data and model output before and after GA tuning.

8.5. Controller tuning

In this section an MRAC, 'Modified MRAC' and 'GA based modified MRAC' are designed for the coupled tank system. The interaction between the two tanks makes the coupled tank process more sluggish than the coupled tank system without interaction. The scaled time constant of a single tank level process is found to be approximately 4 s during the initial study of the hybrid tank process. Hence, a model representing a coupled tank system without interaction is selected as the reference model initially. Some other reference models are also tried to show that the proposed controllers work well even if the reference model and the process model are very different. The transfer function of the first reference model is

$$G_{M1}(s) = \frac{0.1}{16s^2 + 8s + 1}. \tag{14}$$

The adaptation rules for the MRAC parameters θ_1 and $\theta_2$are

$$\frac{d\theta_1}{dt} = -\gamma_1 e\left(\frac{8s+1}{16s^2+8s+1}\right)u_c, \tag{15}$$

$$\frac{d\theta_2}{dt} = -\gamma_2 e\left(\frac{8s+1}{16s^2+8s+1}\right), \tag{16}$$

where γ_1 and γ_2 are the adaptation gains for θ_1 and θ_2 respectively.

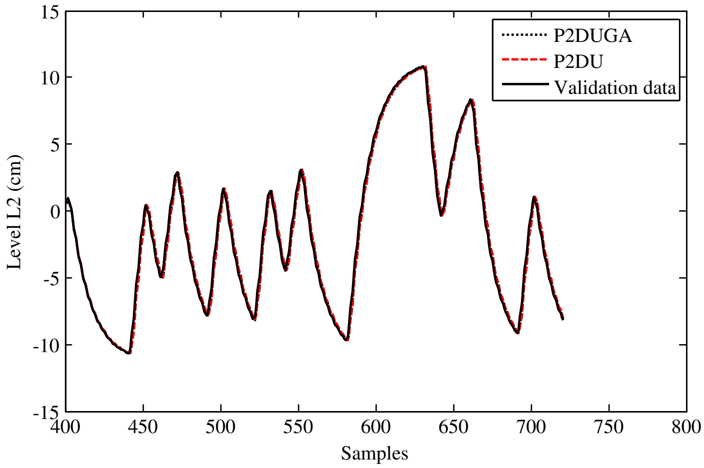

Fig. 16. Comparison of validation data and model output before and after GA tuning.

Table 2
Comparison of models before and after GA tuning.

Model	Parameters			Fit (%)	
	a	b	c	Estimation data	Validation data
P2DU	0.18024	15.9417	13.9976	93.1	92.14
P2DU-GA	0.1816	12.0000	14.0000	94.83	93.73

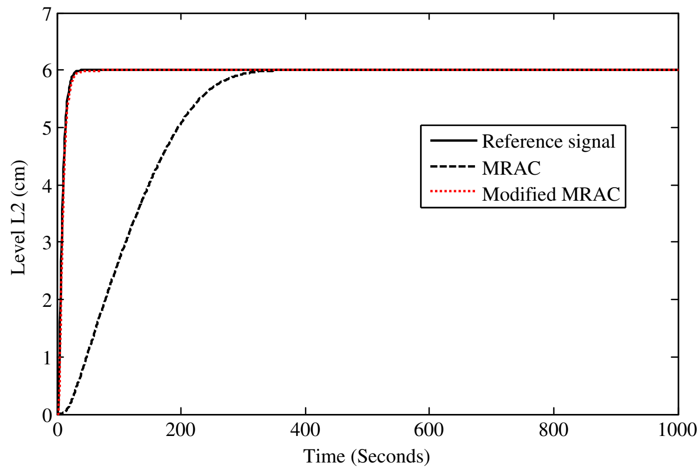

Fig. 17. Step response of MRAC and Modified MRAC.

8.5.1. Modified MRAC

An MRAC as per (4) and (5), and a Modified MRAC as per (6) are designed, and their step responses are obtained. The size of the step input applied to u_c is 60. Fig. 17 shows the responses of the reference model, MRAC and Modified MRAC.

The output of MRAC takes long time to reach the steady state when compared with the time taken by the Modified MRAC. The values of the adaptation gains are 0.0001 and 0.001. Higher values of adaptation gains have resulted in oscillation. The Modified MRAC closely follows the dynamics of the reference model. In this context, rise time, T_R, is taken as the time it takes the plant output y to reach 90% of the final value for the first time. The rise time of the output of the Modified MRAC and MRAC are 13.5973 and 185.9743 s, respectively. Even though it is a huge improvement as far as the Modified MRAC is concerned, the value of $T_R$is still greater than that of the reference model which lies at 13.4361 s. In order to improve the performance of the Modified MRAC further, RGA is applied to find the optimal values of the controller parameters k_p, k_i and $k_{d.}$

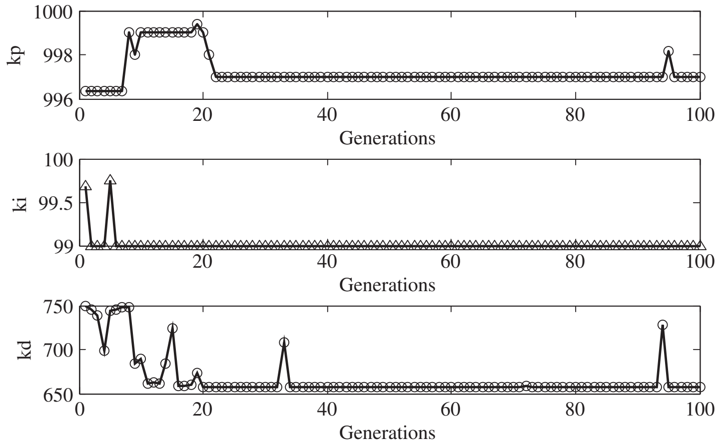

Fig. 18. Convergence of parameters of PID controller.

8.5.2. GA based Modified MRAC

The steps followed in Section 8.4.2 to ensure the optimality of the solution are followed here also. Each controller parameter is represented by a six digit floating point number. Maximum fitness is obtained with the assignment of the values 100, 50, 0.8 and 0.02 to number of generations, population size, crossover rate and mutation rate respectively. The stopping criterion used here is the number of generations.

Fig. 18 shows how the values of PID controller parameters, k_p, k_i and k_d converge when RGA is applied to minimize mean square error (MSE). Fig. 19 shows how the best fitness value and the average fitness value converge. At the twenty-sixth generation, the average fitness approaches the best fitness and from then onwards they stay close to each other. The best fitness does not change for many more generations afterwards. Hence, the estimate can be said to be optimal or near-optimal.

8.5.3. Performance analysis of the proposed controllers

Fig. 20 shows the Simulink model of the GA based Modified MRAC. When the PID parameters are tuned by the PID tuner in the MATLAB control system toolbox the model becomes the proposed Modified MRAC. When the PID parameters are removed from the model it represents the MRAC. A step input is applied to the command signal. Table 3 makes a comparison of how the values of MRAC parameters θ_1 and θ_2 are adapted as the command signal u_c is varied.

In order to carry out performance analysis of different controllers, a step input of size 60 is applied to the command signal u_c and the responses are shown in Fig. 21. It comprises the responses of the Reference model, PID controller, MRAC, Modified MRAC and GA based Modified MRAC. The response of the PID controller has overshoot. The proposed controllers perform better than the MRAC.

The performance criteria are calculated for all the four controllers and shown in Table 4. It makes an obvious comparison of three performance indices, namely, rise time (T_R), settling time (T_S) and mean square error (MSE) obtained from step

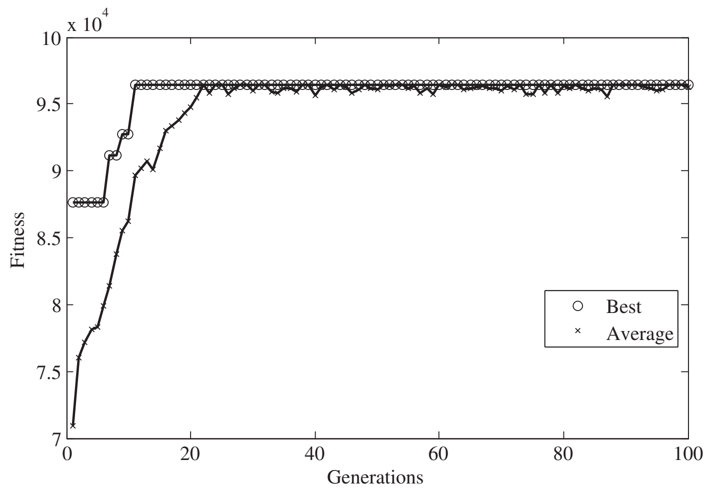

Fig. 19. Convergence of fitness.

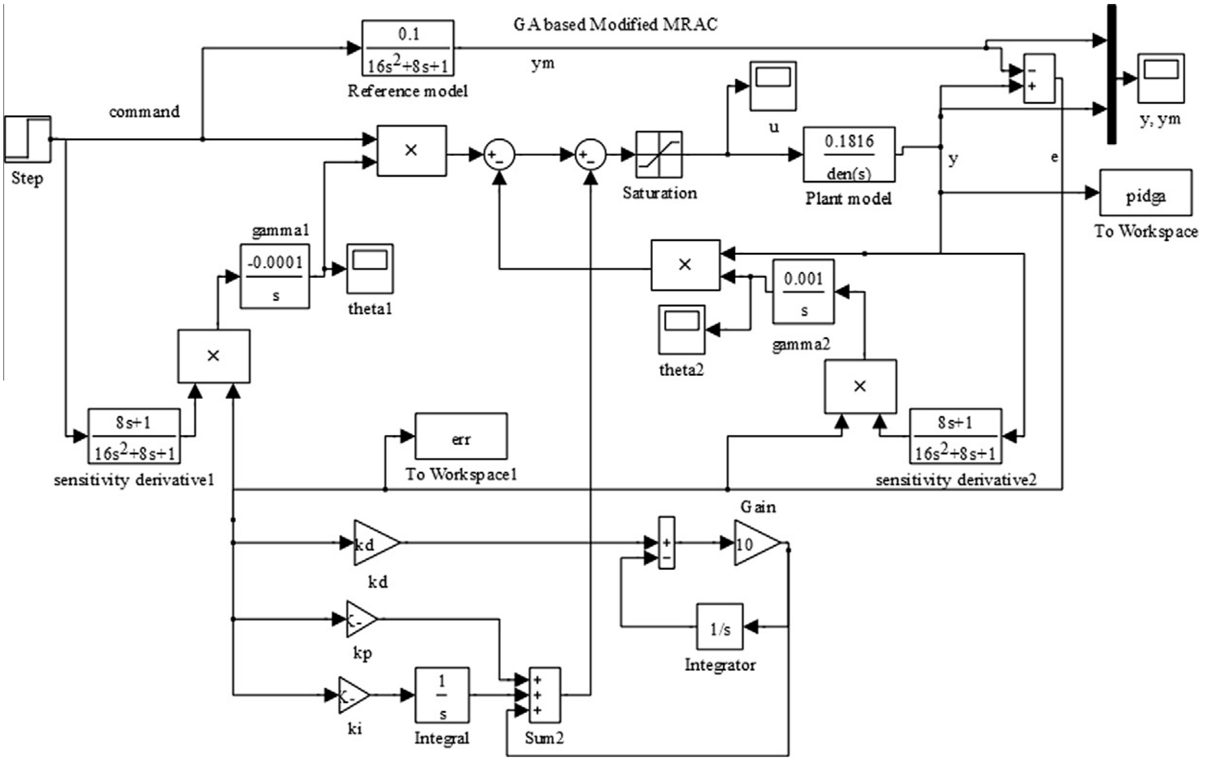

Fig. 20. Simulation model of GA based Modified MRAC.

Table 3
Adaptation of θ_1 and θ_2 for different values of command signal.

Command signal u_c	MRAC		Modified MRAC		GA based modified MRAC	
	θ_1	θ_2	θ_1	θ_2	θ_1	θ_2
20	0.3912	−1.593	0.0026	−0.0061	0.0021	−0.000006
50	0.4016	−1.4901	0.0161	−0.0346	0.0128	−0.000036
60	0.407	−1.4367	0.0229	−0.0471	0.0183	−0.00005

response analysis of all the models. Here, the objective is to make the process follow the reference model as close as possible. In this context, T_R is the time it takes the process output y to reach its final value for the first time and $T_S$is the time it takes y to reach within ±5% of its final value and stay there. In order to appreciate the effect of all the four controllers the relative values of T_R and T_S have to be considered rather than the absolute values. It is because the output of the process has to be compared with the output of the reference model rather than the command signal input u_c. The relative values are obtained by subtracting the values of the reference model from the values of the respective controller models.

Table 5 gives a comparison of T_R and T_S of different controllers with respect to that of the Reference Model. The proposed modification to the MRAC has resulted in reduced values of T_R and T_S than the standard MRAC has. The MSE is also less than that of the MRAC. The Modified MRAC has fared better than the PID controller also in all the three aspects. Thus, it can be said that the Modified MRAC has met the objective of better transient performance.

However, it is clear that the proposed strategy of using RGA to fine tune the parameters of the Modified MRAC performs the best as far as rise time, settling time and MSE are concerned. The shortest rise time enables the proposed controller to quickly adapt to any change in the reference level of the process. Usually the shorter rise time would result in larger value of MSE because of the greater overshoot and the accompanying oscillations. But, the proposed GA based Modified MRAC performs far better than other controllers when it comes to MSE. During the above experiment the process inputs remain well within the bounds and are shown in Fig. 22. It shows that the proposed controllers can be implemented to control the coupled tank process.

In order to check the robustness of the proposed controllers, different types of command signals and different reference models are applied and the results are discussed below. The adaptation gains are assigned the values of 0.001. Responses of the MRAC, Modified MRAC, GA Modified MRAC and reference models are shown in Figs. 23–27. Fig. 23 shows the outputs of

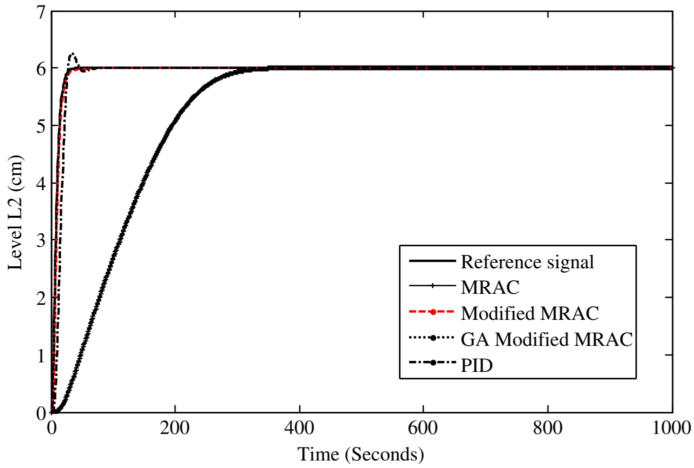

Fig. 21. Step responses of different controllers.

Table 4
Comparison of step responses of different controllers.

Performance index	Reference model	MRAC	Modified MRAC	GA based Modified MRAC	PID controller
T_R (seconds)	13.4361	185.9743	13.5973	13.4476	15.1955
T_S (seconds)	23.3393	282.8218	26.2591	23.3420	39.7807
MSE	–	2.4792	0.0022	1.0518e−5	0.0866

Table 5
Comparison of performance indices with respect to that of reference model.

Performance index	MRAC	Modified MRAC	GA based Modified MRAC	PID controller
T_R (seconds)	172.5382	0.1612	0.0115	1.7594
T_S (seconds)	259.4825	2.9198	0.0027	16.4414
MSE	2.4792	0.0022	1.0518e−5	0.0866

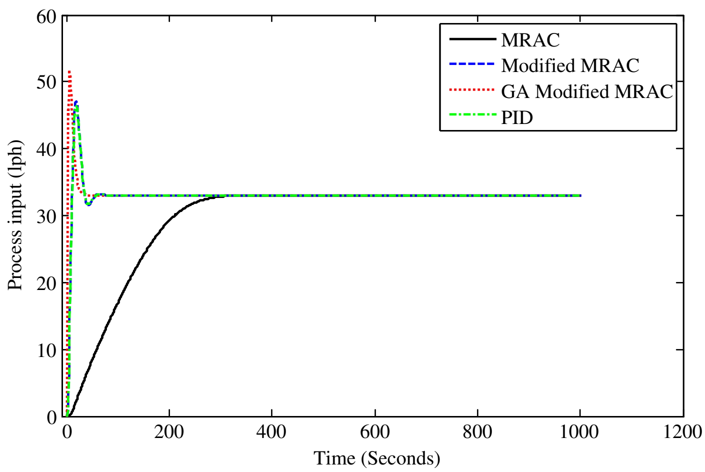

Fig. 22. Process input for a step input of 60 to the command signal.

the reference model and the three controllers to a sinusoidal input of amplitude 20. The output of the MRAC converges with the reference model output after one cycle. But, the outputs of the proposed controllers follow the reference model output from the beginning itself.

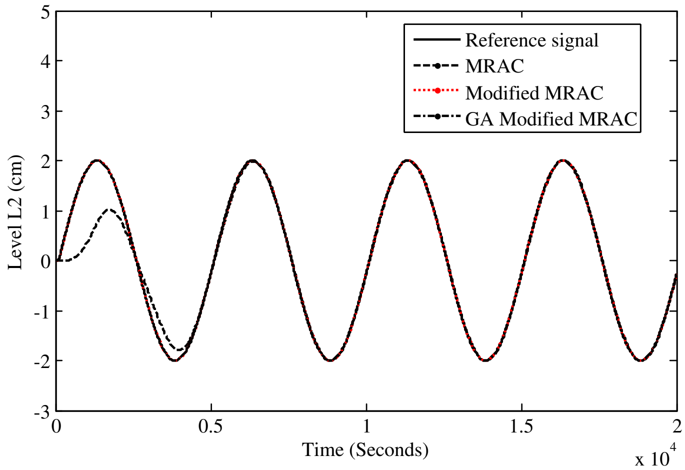

Fig. 23. Responses of the three controllers to a sinusoidal command signal.

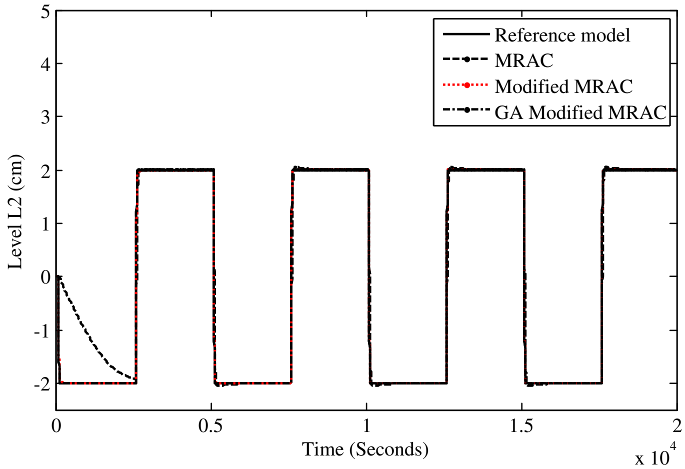

Fig. 24. Responses of the three controllers to a square wave command signal.

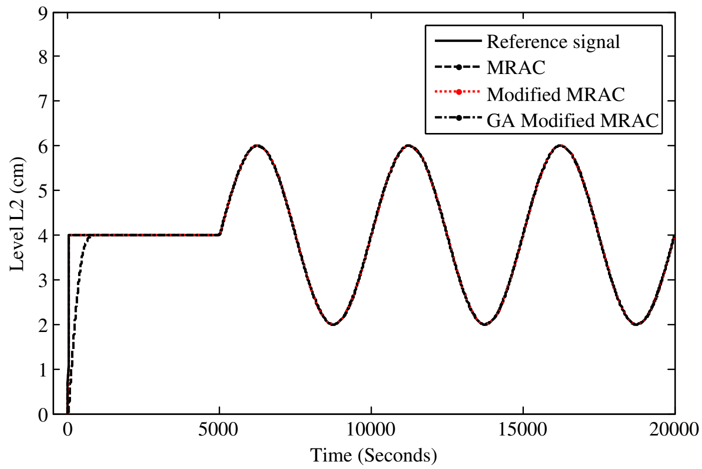

Fig. 25. Responses of the three controllers to a step input followed by a sinusoidal input.

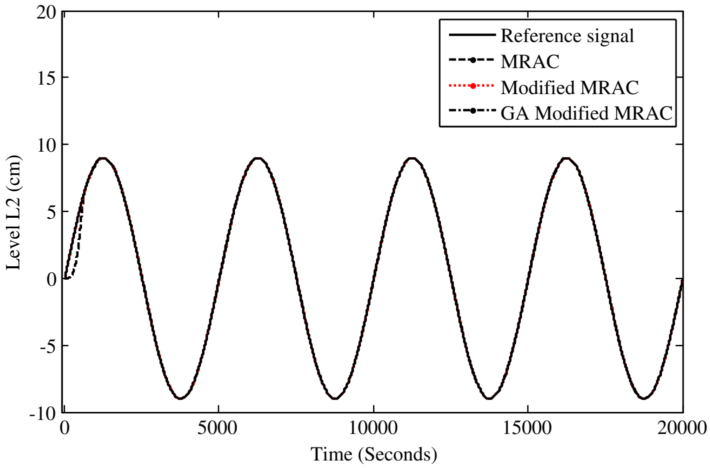

Fig. 26. Responses of the three controllers to a sinusoidal command signal when $G_{M2}(s)$ is used as the reference model.

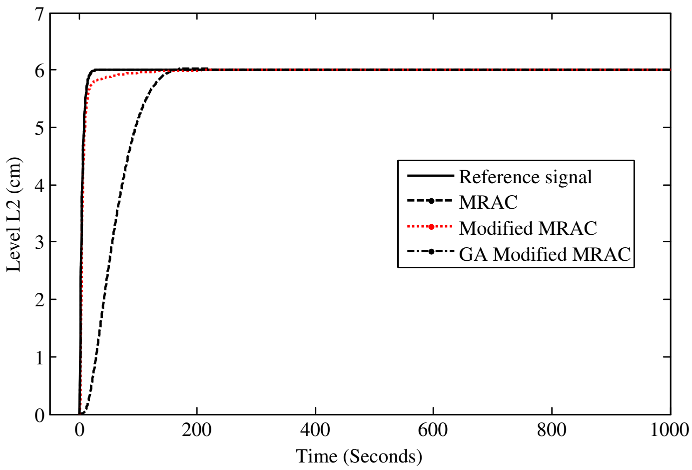

Fig. 27. Step responses of the three controllers when $G_{M3}(s)$ is as the reference model.

Fig. 24 shows the response of the three controllers to a square wave input of amplitude 20. From the results it is understood that even when the input undergoes step change in both directions the proposed controllers give stable response and follow the reference model closely. Fig. 25 shows the response of the three controllers to a step input followed by a sinusoid. The output of the MRAC converges with y_m 1000 s after the step change is applied and remains converged thereafter even when the sinusoidal input is applied to the command signal whereas the outputs of the proposed Modified MRAC and GA based Modified MRAC follow y_m very closely right from the beginning.

In order to test the validity of the proposed controllers even in the case where the parameters of the reference model are very different from that of the process two different reference models are applied. Fig. 26 shows the responses of the three controllers when a reference model whose transfer function is given in (17) is applied. The reference model has smaller time constant when compared to that of the process model.

$$G_{M2}(s) = \frac{0.3}{10s^2 + 12s + 1}. \tag{17}$$

Fig. 27 shows the responses of the three controllers when the reference model has a first order transfer function as given below

$$G_{M3}(s) = \frac{0.2}{4s + 1}. \tag{18}$$

From the analysis of the results it can be said that the proposed controllers give equally good transient and steady-state performance for any type of command signal. It is also shown that even in the case where reference model parameters are very different from that of process model parameters; the proposed controllers give stable response and the process exhibits improved transient and steady-state performance when compared to the performance of the MRAC. The proposed application of RGA to fine tune the PID parameters give better performance than the proposed Modified MRAC.

Even if the plant parameters change due to ageing and other external disturbances, and consequently the difference between the process and the reference model increases the process will faithfully follow the reference model. Thus it is established that the proposed methods can be used to effectively control nonlinear processes.

9. Conclusions

The coupled tank level process is identified as an over-damped second order process using standard system identification procedure. Then, the accuracy of the model is increased by using RGA to fine-tune the parameters of the selected model. The validation results showed that the obtained model is quite capable of capturing the main dynamic characteristics of the coupled tank level process.

The proposed modification to the MRAC scheme has resulted in an improved transient and steady-state performance when compared to MRAC and a well-designed PID controller. The application of RGA to fine tune the parameters of the Modified MRAC has given even better transient and steady-state performance than that of the MRAC, Modified MRAC and PID controller. The simulation results established the supremacy of the proposed GA based Modified MRAC over the other three controllers. The proposed controllers have performed very well even when the reference model order and parameters are very different from the process model parameters indicating the robustness of the design. The proposed Modified MRAC and GA based Modified MRAC give improved transient and steady-state performance in the control of nonlinear processes.

Future work aims to identify the coupled tank level process by using multiple-model approach and to design the proposed GA based Modified MRAC for each local model. The optimal values of the controller parameters for each of the local models will be obtained by using RGA off-line. Then a fuzzy logic controller in which each fuzzy set corresponds to a local model can be designed and implemented.

References

[1] K.J. Åström, T. Hagglund, PID controllers: theory, design, and tuning, Instrument Society of America Research Triangle Park, NC, 1994.
[2] L. Ljung, System Identification: Theory for the User, Prentice-Hall PTR, 1999.
[3] A. Besharati Rad, W.L. Lo, K.M. Tsang, Self-tuning PID controller using Newton–Raphson search method, IEEE Trans. Ind. Electron. 44 (1997) 717–725.
[4] S.A. Billings, Identification of Nonlinear systems, a survey, Proc. IEEE Part D 127 (1980) 272–285.
[5] M. Brown, C.J. Harris, Adaptive neurofuzzy systems for difficult modeling and control problems, in: IEE Colloquium on Advances in Neural Networks for Control and Systems, 1994, pp. 8–10.
[6] K. Valarmathi, D. Devaraj, T.K. Radhakrishnan, Real-coded genetic algorithm for system identification and controller tuning, Appl. Math. Model. 33 (2009) 3392–3401.
[7] K. Kristinsson, G.A. Dumont, System identification and control using genetic algorithms, IEEE Trans. Syst. Man Cybernet. 22 (5) (1992) 1033–1046.
[8] B. Jiang, B.W. Wang, Parameter estimation of nonlinear system based on genetic algorithm, Control Theory Appl. 17 (1) (2000) 150–152.
[9] A. Boroomand, On-line nonlinear systems identification of coupled tanks via fractional differential neural networks, Control Decision Conf. (2009) 2185–2189.
[10] F. Lin, R.D. Brandt, G. Saikalis, Self-tuning of PID controllers by adaptive interaction, in: Proceedings of the 2000 American Control Conference ACC IEEE Cat No00CH36334, American Autom. Control Council. vol. 5, 2000, pp. 3676–3681.
[11] K. Aström, B. Wittenmark, Adaptive Control, Addison-Wesley, 1989.
[12] C. Graham, Goodwin, Continuous time stochastic model reference adaptive control, IEEE Trans. Automat. Control 36 (11) (1991) 1254–1263.
[13] Ching-Chih Tsai, Model reference adaptive predictive control for a variable frequency oil cooling machine, IEEE Trans. Ind. Electron. 51 (2) (2004) 330–339.
[14] T.H. Liu, H.H. Hsu, Adaptive Controller design for a synchronous reluctance motor drive system with direct torque control, IET Electric Power Appl. 1 (5) (2007) 815–824.
[15] Sukumar Kamalasadan et al, A neural network parallel adaptive controller for dynamic system control, IEEE Trans. Instrum. Measure. 56 (5) (2007) 1786–1796.
[16] E. Daniel, Miller, Mansouri Naghmeh, Model reference adaptive control using simultaneous probing, estimation, and control, IEEE Trans. Automat. Control 55 (9) (2010) 2014–2029.
[17] A. Datta, P. Ioannou, Performance analysis and improvement in model reference adaptive control, IEEE Trans. AC-39 (12) (1994) 2370–2387.
[18] D.E. Miller, E.J. Davison, An adaptive controller which provides an arbitrarily good transient and steady-state response, IEEE Trans. AC-36 (1) (1991) 66–81.
[19] R. Jain, N. Sivakumaran, T. Radhakrishnan, Design of self-tuning fuzzy controllers for nonlinear systems, Expert Syst. Appl. 38 (2011) 4466–4476.
[20] W. Chang, Nonlinear system identification and control using a real-coded genetic algorithm, Appl. Math. Model. 31 (2007) 541–550.
[21] C.T. Chao, C.C. Teng, A PD-like self-tuning fuzzy controller without steady-state error, Fuzzy sets and Systems 87 (1997) 141–154.
[22] B. Hu, G. Mann, New methodology for analytical and optimal design of fuzzy PID controllers, Fuzzy Syst. IEEE 7 (5) (1999) 521–539.
[23] K. Valarmathi, J. Kanmani, D. Devaraj, T. Radhakrishnan, Swarm intelligence based system identification and controller tuning, Int. J. Comput. Aid. Eng. Technol. 3 (2011) 443–457.
[24] G. Stephanopoulos, Chemical Process Control: An Introduction to Theory and Practice, Prentice-Hall, Englewood Cliffs, NJ, 1984.
[25] D.E. Goldberg, Genetic Algorithms in Search Optimization and Machine Learning, Addison Wesley, 1989.
[26] S. Rajasekaran, G.A. Vijayalakshmi Pai, Neural networks, Fuzzy logic, and Genetic Algorithms Synthesis and Applications, Prentice Hall of India, 2003.
[27] J.R. Rodriguez Vasquez, R. Rivas Perez, J. Sotomayor Moriano, J.R. Peran Gonzalez, System identification of steam pressure in a fire-tube boiler, Comput. Chem. Eng. 32 (2008) 2839–2848.