CEAI, Vol.22, No.3, pp. 13-24, 2020 Printed in Romania

Model based Controller Design using Real Time Neural Network Model and PSO for Conical Tank System

C. Febina*, D. Angeline Vijula**

Department of Instrumentation and Control Systems Engineering, PSG College of Technology, Coimbatore, India- 641 004
Email: cfb.ice@psgtech.ac.in ; dav.ice@psgtech.ac.in***

Abstract: Most of the industrial processes are nonlinear in nature and demanding an optimal control structure. Conventional controllers do not handle the nonlinear system behaviour effectively and they also have tuning associated problems. In this paper, an optimized new generation RTDA (Robustness, Set point tracking, Disturbance Rejection, Aggressiveness) controller is designed for a nonlinear conical tank system. The enhanced features of RTDA controller enables us to tune the parameters separately without affecting each other to obtain optimum performance where the other contemporary controllers fails to do so. The proposed scheme incorporates NARX (Nonlinear Autoregressive with Exogenous input) neural model in the RTDA controller design as it offers prior multi-step ahead predictions to predict the future plant behaviour. It requires multiple trials to determine the optimal or near optimal values for the tuning parameters for the NN based RTDA controller design and hence a highly skilled meta-heuristic algorithm called Particle Swarm Optimization (PSO) is used to serve that purpose. The performance of NN (Neural Network) based RTDA controller was analysed using MATLAB/Simulink and was compared with NN-MPC and PI-IMC controllers. The results show that the proposed NN RTDA-PSO control frame work performed optimistically well compared to other control schemes.

Keywords: conical tank, modelling, MPC, nonlinear, neural network, PSO, RTDA

1. INTRODUCTION

Nonlinear systems are of massive interest to researchers as the systems appears to be erratic in contrast to linear systems. The challenges emanated by them are to be greatly considered in order to design an optimal controller for an efficient closed loop response. The conical tank process is one such system used in industries like water treatment plants, paper mills, sludge tanks, biofuels, wine making, fertilizer industries etc. The level control of such process with nonlinear dynamics due to varying cross sectional area is a challenging task. Conventional controllers play a vital role in process industries and are exceedingly popular for their simple structure and robust performance. The linear parameter varying model based PI controller was designed for conical tank system for efficient performance (Vijalakshmi et al., 2014). The robustness of the system was upgraded by employing internal model control (IMC) technique for tuning the PI controller (Diwakar et al., 2015). Adaptive passivity based controller (APBC) and fractional order PI controller was proposed for controlling the conical tank system for robust performance (Travieso et al., 2017). The P-IMC algorithm was found to be more efficient than the conventional PID algorithm with respect to tuning procedure and control performance (Vasile, 2020). Conventional controllers was well adopted for inherently nonlinear processes (Prashant et al., 2004) but their shortcomings for the highly complex dynamic processes with uncertainties and tuning associated problems led to the development of supervisory model based controllers. The model predictive controller (MPC) was evaluated on the nonlinear process and proved to be more robust than the PID controller (Jean, 2014). The predictive capability of model based controllers was further enhanced by using a neural network model to give trust worthy predictions. The recurrent neural network (RNN) model based MPC controller was designed for flight of unmanned multi-quadrotor system (Boyang et al., 2019). RNN-MPC was demonstrated over industrial case studies and its performance was positively analysed for the system dealing with nonlinearities, time delays, and noise (Nicolas et al., 2019). But the main drawback lies in the applicability of MPC control structure as it requires high computational online calculations for solving the optimization problem.

In control literature, one such controller named RTDA controller was proposed in order to overcome the inadequacies with its enhanced features. The RTDA controller cartels the simplicity of PID controller with the flexibility of model predictive controller (MPC). The advantage of RTDA controller is that the tuning parameters has a direct relationship with the performance attributes. But the conventional controllers do not succeed in this aspect as the key features of the overall controller performance doesn't sync with its controller parameters (Kapil et al., 2006). RTDA controller is proven to be better than conventional PID controller as it could be designed and implemented transparently and directly (Kapil et al., 2009). Improved tuning rules were proposed for SISO delay dominant and inverse response processes (Antonius et al., 2011). The efficiency of RTDA controller was compared with PID controller and MPC for FOPDT, SOPDT processes (Srinivasan et al., 2012). RTDA controller for a nonlinear CSTR process was designed and analysed for both minimum and non-minimum operating condition (K. Anbarasan et al., 2015). RTDA controller was designed for a general multivariable system and the simulation results was found to improve the dynamic control capability and robustness (Yu Sen et al., 2018). Therefore RTDA controller can be well employed for various nonlinear dynamical applications as it uses a simple control structure with futuristic multi-step ahead prediction and avoids tuning associated problems that exist in PID controller and model predictive controller.

In any control system application, better performance is realized by designing a control law that indirectly relies on the process model which can be identified using experimental data of the process. Therefore system identification (SI) is an important task to be performed before the controller design. There is a need for proper adaptive plant model so as to minimise the modelling errors, increase the robustness and reduce the plant/model mismatch. With emerging new intelligent techniques such as artificial neural networks (ANN), it is possible to offer effective solution to the plant control problems (Antsaklis., 1990). ANN is used for SI and prediction for nonlinear dynamical plants, as neural networks has the ability to approximate nonlinear functions accurately for their use in dynamic models to represent a nonlinear plant (Narendra et al., 1990). ANN model using back propagation algorithm is implemented in MPC design to determine the future inputs to minimize the errors (Wills et al., 1992; Abubakar., 2015). ANN is considered to be a powerful tool for analyzing the data sets and developing a trained model which could make accurate predictions (Gomez et al., 2004). RBF neural network was used for modelling and controller design for a conical tank system (Venkatesh et al., 2018). Further, researchers employed recurrent networks that use a global feedback to a static multilayer perceptron, for various applications. Therefore SI using neural networks plays a major role in many of the industrial applications in the existing control literature.

Based on the literature survey on the nonlinear control, a RTDA controller is proposed for controlling the conical tank process which exhibits highly nonlinear dynamics. To enhance the predictive capability of the RTDA controller, a NARX NN model is developed using the experimental data obtained from real time conical tank system. The best fit NARX model is developed and is implemented in the RTDA controller design for multi-step ahead prediction. The tuning parameters values are to be varied between 0 and 1, to achieve desired performance. If the values of the controller parameters are optimized rather than making a random choice, then it will yield an improved closed loop performance. The well-known optimization technique called particle swarm optimization (PSO) which is based on nature inspiration concept, has gained a unique importance in the field of control system design. Therefore PSO was adopted to optimize the tuning parameter values of the NN model based RTDA controller for the conical tank system. The following sections of the paper are organised as follows: The system description is briefed in section 2. The TF (transfer function) model and NARX neural model are built for both operating regions of the conical tank system in section 3. A brief introduction about MPC is outlined in section 4. The proposed RTDA control structure with NARX NN model for conical tank system is presented in section 5. The optimization of RTDA controller design parameters using PSO is presented in section 6. The results are discussed in section 7 and conclusions are drawn from the results in section 8.

2. SYSTEM DESCRIPTION

Conical Tank System (CTS) is a single input-output process. The level of the tank is the controlled variable and the voltage applied to the motor is the manipulated variable which in turn changes the inflow rate of the tank when the voltage is varied. Thus in the conical tank system, gain and time constant are functions of the process variable. The laboratory conical tank system is shown in figure 1. Conical tank system is a highly nonlinear process with varying cross-sectional area where its area gets steeper towards the end for the guaranteed drainage of fluids in chemical industries. The total height and top radius of the laboratory conical tank setup is 52 cm and 24 cm respectively.

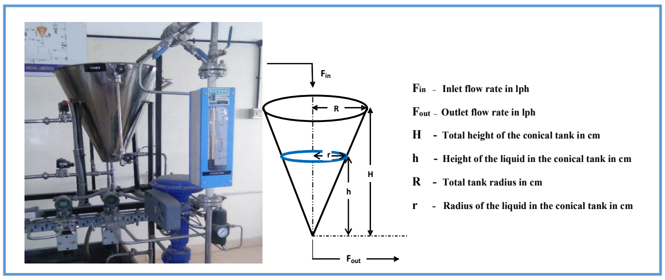

Fig. 1. Laboratory conical tank system.

$$\frac{dh}{dt} = \frac{F_{in-} \text{k}\sqrt{h}}{\left\{\pi h^2 \left(\frac{R}{H}\right)^2\right\}} \tag{1}$$

From equation (1), it is inferred that the level of tank depends on the inflow rate. As the level changes, the gain and the time constant of the conical tank process also changes.

3. EMPIRICAL MODELLING

As conical tank system is a highly nonlinear process, the linearized first principle model result in omission of actual dynamics of the system. Therefore empirical models are obtained using system identification (SI) technique. SI is an experimental technique that establishes a dynamic relationship between the input-output variables with proper data set which is obtained through perfect experimentation and execution. In this study, linear transfer function (TF) model and NARX NN based model are developed using empirical data obtained from the real time conical tank system through proper data acquisition. The operating range of the real time conical tank system is (0-50) cm. It is split into two operating regions as zone 1(4.5-34) cm and zone 2 (34.1-50) cm.

3.1. The Transfer Function (TF) Model

The step change of known value is imposed on the process at steady state level of 4.5 cm and the response is observed from zone 1. The final steady state value is noted as 34 cm. Similarly another known step change is given and the response is noted from zone 2. In the second observation, the initial level at steady state is 34 cm and final steady value is

50 cm. The input-output data obtained by this experimentation is used for system identification using MATLAB. The TF model obtained for both the operating regions is given in the Table 1. The figure 2 depicts the two zones of the conical tank, where voltage (V) is plotted against time (sec).The output voltage (1-5V) from the DAQ relates to the corresponding output level (0cm - 50cm) of the liquid in the conical tank. The empirical (TF) model developed, follows the response of experimental data recorded for both the zones.

Table 1. Transfer function for both the operating zones.

Operating region	Transfer function
Zone 1 Range(4.5 to 34 cm)	$\frac{1}{294.377s+1}$
Zone 2 Range(34.1 to 50 cm)	$\frac{1.495}{593.1198s+1}$

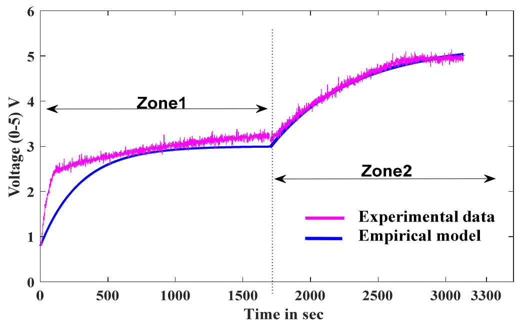

Fig. 2. Open loop response of the conical tank system.

3.2. NARX Neural Network Model

System identification when preceded by artificial neural networks results in effective controller design and enhances the prediction capability. Neural network model based controllers had already demonstrated improved closed loop performance as seen in control literature (Martin T H et al., 2002). As the proposed RTDA controller is a model based controller, artificial neural network (ANN) methodology is used for identifying the model in this work. The static nonlinear mapping between the given inputs and the outputs is given in equation (2) where the output at discrete time step k depends only on the input $u(k)$ at the same time instant.

$$y(k) = f(u(k)) \tag{2}$$

In the case of models built for dynamical systems, output of the system at a given time instant depends on the current inputs as well as the previous output of the system. In feed forward architecture, tapped delay lines are used as inputs of all the neurons. The output of the *n-th* neuron in the layer *d* is given in the equation (3).

$$y_n^{(d)}(k) = g\left(\sum_j {w_{nj}^{(d)}}^T \; x_j^{(d)}(k)\right) \tag{3}$$

where $w_{nj}^{(d)} = \left[w_{nj,0}^{(d)}, w_{nj,1}^{(d)}, \dots\dots, w_{nj,M_n^{(d)}}^{(d)}\right]^T$ is the *j-th* filter coefficient vector of node *n* in layer *d.* The filter's input vector is framed from the delayed outputs of the *j-th* neuron of the previous layer.

$$y_j^{(d-1)}(k) = x_j^{(d)}(k) =$$
$$\left[x_j^{(d)}(k), x_j^{(d)}(k-1), \dots\dots, x_j^{(d)}(k-M_n^{(d)})\right]^T \tag{4}$$

Similar to linear dynamic black box modelling, nonlinear system identification can also be framed. In nonlinear model architecture a regressor vector is used (Sjoberg et al., 1995). The output of the model y_m is defined as a parametrized function of the regressor vector φ as given in the equation (5), where Θ is termed as parameter vector.

$$y_m(k) = f(\Theta, \varphi(k)) \tag{5}$$

The regressor can be designed from the past inputs, past model outputs, past system outputs etc. Specific network architecture designed using past inputs and past system outputs is called as nonlinear autoregressive with exogenous inputs (NARX) model, which is a type of RNN network. NARX model uses the nonlinear mapping capability of multilayer perceptron (MLP) and also uses feedback from the output layer to input layer providing better predictions for SI as it uses additional input stored in the previous values (Zina et al., 2018; Ashok et al., 2018). NARX neural model, has a single input $u(k)$ that is applied to the tapped delay line memory of N units and a single output $y(k)$ which is fed to the input through another tapped delay line memory of M units as shown in figure 3. The signal vector applied to the input layer of MLP consists of exogenous inputs (i.e. the past values of the input) and delayed values of the output are used as regressor as shown in equation (6).

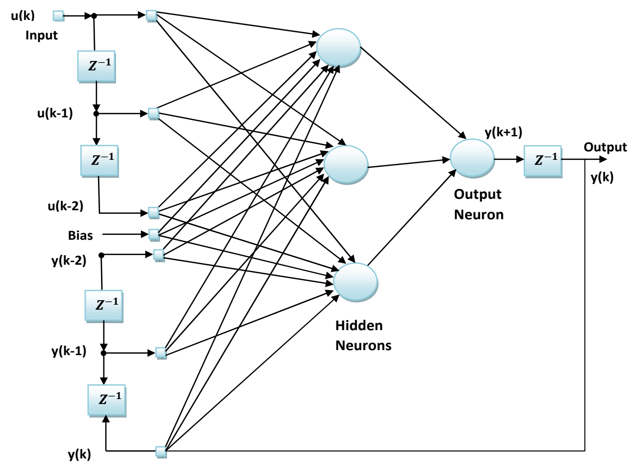

Fig. 3. NARX neural network.

$$\varphi(k) = [y(k\text{-}1),\ y(k\text{-}2), \dots\dots,\ y(k\text{-}M), u(k\text{-}1),\ u(k\text{-}2),\ \dots\dots,\ u(k\text{-}N)\ ] \tag{6}$$

The procedure of training, validation and testing is being done using real time experimental data of the conical tank process. The proposed work uses LM (Levenberg-Marquardt) algorithm because of its good learning capability and convergence behaviour. Training is carried out by gradually increasing the number of hidden neurons as it increases the flexibility of neural network. The goal of the training is to reduce squared error emanating from the elements as a result of the error sequences. The output error *e(k)*, for the dynamic network at discrete time step k and the total error e_{tot} is expressed as given in the equation (7).

$$e_{tot} = \sum_{k=1}^{N} (e(k))^2 \quad (7)$$

A set of test data which is not used in training, is required to check the generalization capability of the model.

The performance of the model is the sum of the squared errors at all test points as well as at all training points. The mean squared error M_e is defined as the sum of the squared errors.

$$M_e(\Theta) = \frac{1}{P} \sum_{k=1}^{P} (e(k))^2 \quad (8)$$

$$M_e(\Theta) = \frac{1}{P} \sum_{k=1}^{P} (y(k) - y_m(k))^2 \quad (9)$$

where $e(k) = y(k) - y_m(\Theta, \varphi(k))$. The length of the tapped delay lines and the number of hidden neurons which implement the nonlinear mapping has to be properly selected. Mean square error (MSE) and regression values are monitored to develop a best fit model suitable for the controller design. The procedure is repeated for developing a best fit model for both the zones of the conical tank. NARX neural model is developed using the input and output experimental data of the conical tank and then implemented in RTDA controller design for better predictions.

3.3. System Identification using Neural Networks for Conical Tank System

The input-output real time data array obtained is of (1X 1700) matrix for zone 1. Apart from training process, validation and testing are the other two steps carried out. Target timesteps for training, validation and testing are 1700 which are randomly divided as 70% target timesteps (1190 samples) for training, 15% (255 samples) for validation and remaining 15% (255 samples) for testing. Similarly the real time data array for zone 2 is of (1X 1431) matrix form. Target timesteps for training, validation and testing are 1431 which are randomly divided as 70% target timesteps (1001 samples) for training, 15% (215 samples) for validation and remaining 15% (215 samples) for testing. The number of hidden neurons and delays are repeatedly changed such that the regression value and MSE for all the 3 steps are one and zero respectively. The number of epochs used for ANN training is 1000. Significant correlation in prediction errors is monitored. The prediction is possibly improved by increasing the number of delays in the tapped delay lines. The number of hidden neurons are increased until the network performance is satisfactory. The best fit model is obtained using 20 hidden neurons and 2 delays. Time response plots for the best NARX model for zone 1 and zone 2 are shown in figures 4 and 5 respectively.

TF model is compared with NARX NN model based on MSE as shown in Table 2. MSE is found to be less for NARX neural model for both the zones as shown in figure 6 and hence could be used in RTDA control design for better prediction of the plant behaviour.

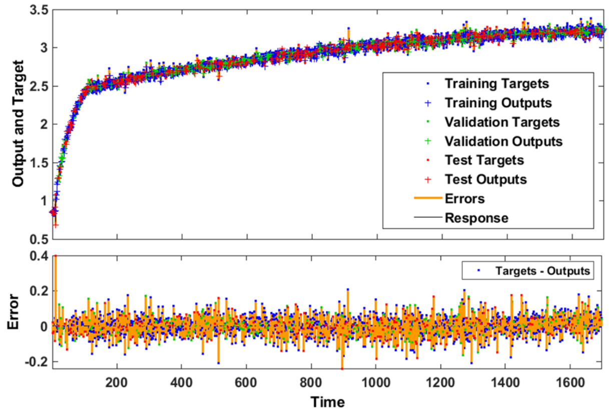

Fig. 4. Time-series response (NARX model) for zone 1.

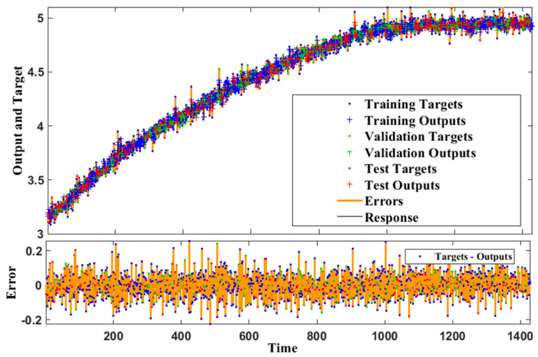

Fig. 5. Time-series response (NARX model) for zone 2.

Thus the data driven NARX model is implemented in control design frame work of RTDA controller and the same model is implemented in MPC control frame work for reasonable comparison.

Table 2. Mean Square Error (MSE).

TF Model		NARX Neural Model	
Zone1	Zone 2	Zone 1	Zone 2
0.1177	0.0064	0.00312	0.00049

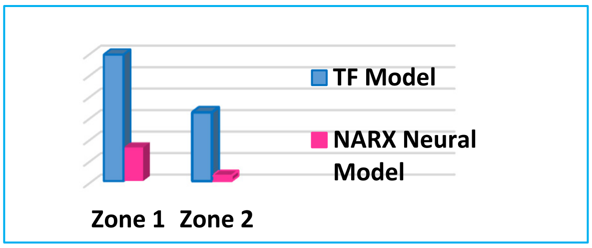

Fig. 6. Performance comparison based on MSE.

4. MPC FORMULATION

MPC contributes to a fair extent as it is a part of autonomous industries with a highly advanced control framework based on receding horizon control concept. MPC comprises of process model, constraints and cost function. The major requisite of MPC is the process model that describes the behaviour of the system. The chief idea of MPC is solving the optimization problem at each time step such that cost function is minimized as shown in equation (10).

$$J=\sum_{k=0}^{P} (y_m - r_f)^T Q (y_m - r_f) + \sum_{k=0}^{P} \Delta u^T R \Delta u \quad (10)$$

where r_f – Setpoint, P – Prediction Horizon, y_m – Predicted process output, R- Control weight matrix, Q- Output error weight matrix, Δu- Change predicted in the control value. The trained developed NARX models for open loop predictions which includes past inputs and past outputs is incorporated into model predictive control structure. This neural architecture shows trade-off between computational tractability, RNN modelling capabilities and solving RNN optimization problem.

5. RTDA CONTROLLER DESIGN

The primary objective of any controller design is to accomplish exemplary closed loop performance in terms of robustness, setpoint tracking, disturbance rejection, and aggressiveness. The RTDA controller has transparent tuning parameters and simplified model prediction which helps us to tune each parameter independently to achieve better performance. It has four tuning parameters (θ_R, θ_T, θ_D and θ_A) which are directly associated to the corresponding properties of the closed loop control system and the values of tuning parameters can be varied between 0 and 1.

5.1. The Block Diagram

The block diagram of RTDA controller in the closed loop system is shown in the figure 7. The controller performance is improved by the robustness parameter θ_R, in the presence of plant/model mismatch and model uncertainties. The parameter θ_D is the disturbance rejecting parameter that deals with future error predictions. It helps to predict the effect of disturbance that is acting on the process output. Changing the value of θ_D has a limited effect on robustness of the controller and does not have any effect on the servo performance.

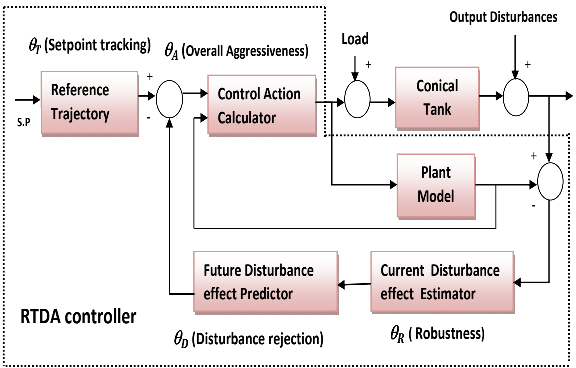

Fig. 7. Block diagram of RTDA controller design.

The parameter θ_T is the setpoint tracking tuning parameter. The characteristic polynomial does not rely on the value of θ_T therefore stability of RTDA controller is independent of the choice of value of θ_T, as the setpoint filter lies outside the feedback loop. The overall aggressive parameter θ_A is the unique one which is used to adjust the speed of response in both servo and regulatory responses. It determines the distance of future output prediction. (Bagyaveereswaran et al., 2016; Geetha et al., 2016).

5.2. RTDA Algorithm

The actual dynamics of the conical tank process is approximated as a first order model to describe the process behaviour. The generalized transfer function model is given by equation (11),

$$y(s) = \frac{K}{\tau s+1} u(s) \quad (11)$$

where K is the steady state gain and τ is the time constant. The predicted output, $\hat{y}(k+i)$ is given in the equation (12),

$$\hat{y}(k+i) = a^i \hat{y}(k) + b\eta_i u(k) \qquad \text{for } 1 \le i \le P \quad (12)$$

where, $\eta_i = \frac{1-a^i}{1-a}$; a =exp $\left(\frac{\Delta T}{\tau}\right)$; b = K(1-a). The control action $u(k)$ remains the same for the whole prediction horizon (P), i.e.,

$$u(k+i) = u(k) \qquad \text{for } 1 \le i \le P \quad (13)$$

This prediction must be updated to include the unmeasured disturbances effect and other sources of modelling errors. The model output $\hat{y}(k)$ has deviation from actual process output $y(k)$ due to plant/model mismatch. Prediction at each instant has to be updated. The model mismatch is given by equation (14), where $e(k)$ is the current error.

$$e(k) = y(k) - \hat{y}(k) \quad (14)$$

The non-biasing prediction error is denoted by $\hat{e}_d(k)$, which is determined by the parameter θ_R as given in the equation (15), $\hat{e}_d(k-1)$ is the weighted sum of prior error information, and e(k) is the current error information.

$$\hat{e}_d(k) = \theta_R \hat{e}_d(k-1) + (1-\theta_R)\, e(k) \quad (15)$$

The future estimates of the error is determined by the disturbance rejecting parameter θ_D as given in the equation (16),

$$\hat{e}_d(k+i) = \hat{e}_d(k) + \frac{1-\theta_D}{\theta_D}[1-(1-\theta_D)^i]\Delta \hat{e}_d(k) \quad \text{for } 1 \le i \le P \quad (16)$$

where, $\Delta \hat{e}_d(k) = \hat{e}_d(k) - \hat{e}_d(k-1)$ i.e., the difference between errors at two consecutive instants. The updated predicted output for P-step prediction is given in the equation (17),

$$\tilde{y}(k+i) = \hat{y}(k+i) + \hat{e}_d(k+i) \qquad \text{for } 1 \le i \le P \quad (17)$$

The desired setpoint trajectory $y_t(k)$ can be determined as given in the equation (18), where S_p is the desired setpoint.

$$y_t(k) = \theta_T y_t(k-1) + (1-\theta_T)\, S_p(k) \quad (18)$$

Assuming, setpoint remains the same for the whole prediction horizon i.e. $S_p(k+i) = S_p(k)$, i = 1,2,…..,P. The future reference trajectory is given in the equation (19).

$$y_t(k+i) = \theta_T^{\,i} y_t(k) + (1-\theta_T^{\,i})\, S_p(k) \qquad \text{for } 1 \le i \le P \quad (19)$$

The objective function of the RTDA controller is given in the equation (20).

$$\min_{u(k)} \Sigma_{i=1}^{P} [y_t(k+i)-\tilde{y}(k+i)]^2 \tag{20}$$

The control action $u(k)$ is updated to minimize the difference between model predicted output $\tilde{y}(k)$ and reference trajectory $y_t(k)$ for P-step. On solving the optimization problem, the expression for $u(k)$ is given in the equation (21),

$$u(k) = \frac{1}{b}\frac{\Sigma_{i=1}^{P}\eta_i \Psi_i(k)}{\Sigma_{i=1}^{P}\eta_i^{2}} \tag{21}$$

$$\Psi_i = y_t(k+i) - a^i\hat{y}(k) - \hat{e}_d(k+i) \quad \text{for } 1\leq i\leq P \tag{22}$$

where Ψ_i is the stipulated error. The overall aggressiveness tuning parameter θ_A depends on prediction horizon (P) as given in the equation (24),

$$P = 1 - \frac{\tau}{t_s}\ln(1-\theta_A) \tag{23}$$

$$\theta_A = 1-e^{-(P-1)\frac{\Delta t}{\tau}} \tag{24}$$

where t_s is the sampling time and θ_A is the aggressiveness tuning parameter.

5.3 Control Structure Design

The integration of NARX neural network model into the RTDA control structure is shown in figure 8. NARX model performs futuristic multi step ahead predictions, as it uses additional information that is previously stored in the input and output variables as shown in equation (6). The estimated output from the neural model is almost same as that of the actual process. The difference between the same is the error $e(k)$ and it fed into current error estimator block, where $\hat{e}_d(k)$ is the estimated current error.

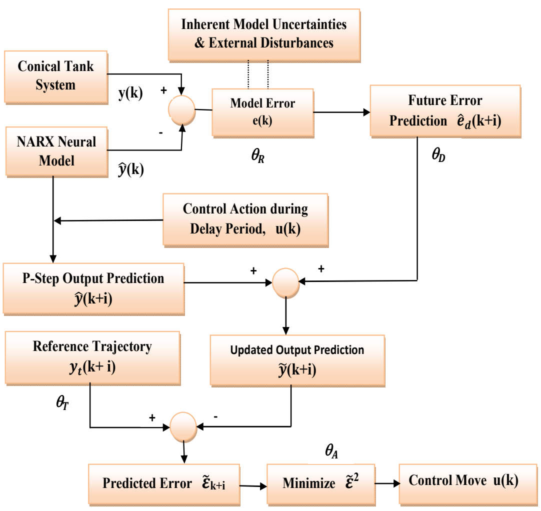

Fig. 8. Integration of neural network into the RTDA controller design.

The future error $\hat{e}_d(k+i)$ is predicted in the future error predictor block. The updated output prediction $\tilde{y}(k+i)$ is compared with the reference trajectory which is calculated for P-step. The required control action is taken by the RTDA controller by minimizing the objective function. The values of the tuning parameters (θ_R, θ_T, θ_D and θ_A) are varied based on trial and error method. The distinguishing feature of neural network is its ability to model nonlinear systems as it is intrinsically adaptive. Hence NARX neural network model is implemented in RTDA controller design and the closed loop responses is evaluated based on ISE (Integral Squared Error) performance criterion.

A good set of controller parameters (θ_T, θ_R, θ_D) can yield improved closed loop performance. Therefore the performance of the proposed RTDA controller is enhanced by optimizing the values of the tuning parameters since it is tedious to select them based on trial and error method (exhaustive search). Optimization is the key root to find the best solution for any problem. The goal of optimization always remains to fetch the value of variables that can maximize or minimize the objective function, satisfying the constraints. To perform this, nondeterministic algorithm called PSO is employed in this work.

6. SCHEDULING PSO FOR RTDA CONTROLLER PARAMETERS

Particle swarm optimization is a nature inspired meta-heuristic, population based algorithm. PSO technique was widely used for many applications where the optimized results were accurate and efficient compared with other popular techniques (Mahmud et al., 2011; Eswaran et al., 2017: Latha et al., 2013). PID controller parameters were efficiently optimized for a nonlinear CSTR process which resulted in minimum ISE (Geetha et al., 2012). This technique is easy to implement and has a significantly different information sharing system and is even employed for distillation column where NARX lag space selection for MLP neural model was optimized using PSO (Mohd N N et al., 2012).

PSO makes use of fixed cluster of particles called the swarm. All the particles are randomly initialized and enter into iteration process and endorsed to move around to explore the whole search space dimension. Over a number of iterations, each particle j exhibits different performance in each iteration considering their present and past values. These particles are guided by previous velocity of each particle, distance from the individual particle's personal best position, and distance from the swarm's global best position. Velocity and position update is given in the equation (25) and (26) respectively.

$$v_{j,m}^{(t+1)} = w\, v_{j,m}^{(t)} + a_1 r_1 [pbest_{j,m}^{(t)} - p_{j,m}^{(t)}] + a_2 r_2 [gbest_{j,m}^{(t)} - p_{j,m}^{(t)}] \tag{25}$$

where j=1,2…N(number of particles in group), $m=1,2,\ldots,d$ (dimension), $v_{j,m}^{(t)}$ is the velocity of the particle at time t ; $p_{j,m}^{(t)}$ is the current position of the particle at time t ; $pbest_{j,m}^{(t)}$ is the individual best value of the particle as of time t ; $gbest_{j,m}^{(t)}$ is the global best position as of time t. Here the parameters to be optimized are (θ_T, θ_R, θ_D), therefore the dimension of the problem is 3. The size of the swarm (number of birds) is chosen as 20 and maximum number of bird steps is 10. PSO parameter a_1 & a_2 are the acceleration coefficients whose value is chosen to be one for both the zones of the conical tank. Inertia coefficient (w) usually selected is in the range of

(0.8 to 1.2). Low value of w hastens the convergence and a high value encourages exploration in space dimension. The value of w is chosen to be 0.9. The random values r_1 and r_2 are of (3x10) matrix. The particles position is updated by adding velocity.

$$p_{j,m}^{(t+1)} = p_{j,m}^{(t)} + v_{j,m}^{(t+1)} \quad (26)$$

All the particles collectively communicate with each other having its personal best and global best to integrate at a particular point to obtain an optimal solution. Flow chart of PSO algorithm for NN based RTDA controller design is shown in figure 9.

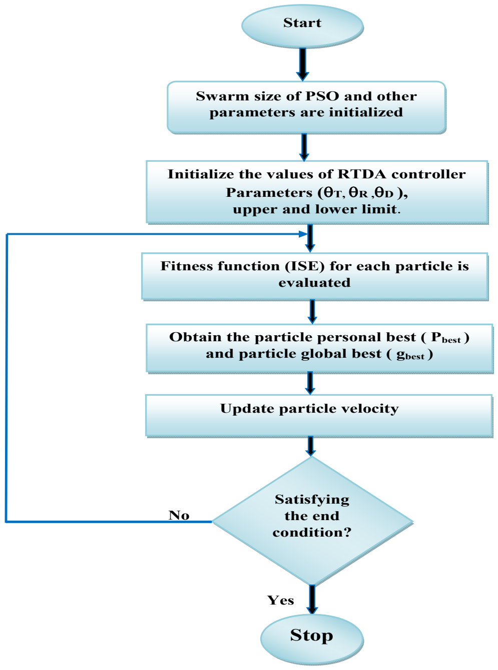

Fig. 9. The flowchart of PSO- NN based RTDA controller

The performance of optimization algorithm in terms of efficiency, speed of convergence, and accuracy depends on objective function. In this paper, two parameter based objective function is considered, peak overshoot(M_P), which is a time domain constraint and integral squared error (ISE) as given in the equation (27).

$$J(\theta) = \alpha \cdot ISE + \beta \cdot M_P \quad (27)$$

$$\theta = [\theta_T, \theta_R, \theta_D] \quad (28)$$

where θ are the parameters to be optimized, and the weighting function $\alpha = 10$ and $\beta = 10$. The search boundary for $(\theta_T, \theta_R, \theta_D)$ is consigned as $\theta_T, \theta_R, \theta_D$: min 0 to max 1. The block diagram of NN based RTDA controller with PSO is shown in figure 10. The Meta-heuristic algorithm (PSO) is used to tune the parameters of RTDA controller in order to ensure optimum control performance of the conical tank system. The controller parameters $(\theta_T, \theta_R, \theta_D)$ of RTDA controller are tuned offline for 100 iterations using the process model for set point tracking and regulatory performance. The tuning procedure is repeated 15 times independently, then the best values among the trials are implemented in the controller design in order to achieve efficient performance for both the operating points. The value of the controller parameter θ_A depends on the prediction horizon (P) chosen. Initial swarm of particles in search space dimension are first developed by PSO and represented by the matrix. Each particle in search space represents an entrant solution for RTDA control parameters and their values are normalised to stay between 0 and 1 value. Here the position and velocity of the particle are represented by matrices of 3x swarm size where, swarm size is the number of particles. Optimized values for NN-RTDA controller parameters using PSO yielded a very good response for both the operating zones of the conical tank.

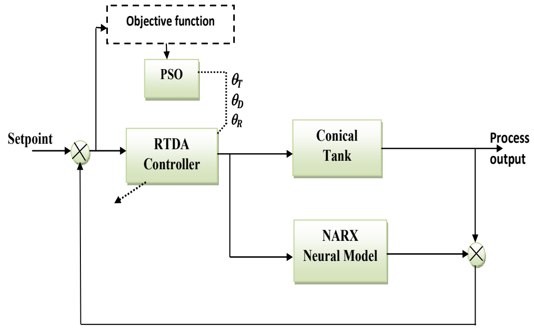

Fig. 10. The block diagram of proposed NN-RTDA (PSO) control design

7. RESULTS AND DISCUSSION

The developed NARX model is integrated into the RTDA control structure. Programming is done using if-else block in MATLAB\Simulink in order to select specific zone of conical tank in accordance with the given setpoint conditions. The effects of each tuning parameter (θ_R, θ_T, θ_D and θ_A) of RTDA controller are analysed by assigning the setpoints in two different zones where each tuning parameter is varied, by keeping other three tuning parameter values constant. To serve this purpose some parameters are chosen on trial and error basis and are listed in the Table 3 and Table 4. Figures 11 and 15 shows the overall aggressiveness response for both the operating zones. The prediction horizon P is selected as 2, 10, 15, 20 for both the zones and the value of θ_A is calculated using the equation (24) as 0.00338, 0.03, 0.0464, 0.0625 respectively for zone 1 and 0.0016, 0.015, 0.02, 0.03 respectively for zone 2. By analysing the responses it is found that good setpoint tracking and disturbance rejection is observed for smaller value of θ_A. Thus it is proved that the overall aggressiveness of the system can be easily altered with the help of θ_A.

Table 3. Tuning Parameters (Zone 1).

	Trial 1	Trial 2	Trial 3	Trial 4
θ_A [$\theta_T = 0.1, \theta_D = 0.1, \theta_R = 0.6$]	0.0033	0.03	0.04	0.06
θ_T [θ_A=0.0033, $\theta_D = 0.1, \theta_R = 0.6$]	0.1	0.8	0.96	0.99
θ_D [θ_A= 0.0033, $\theta_T = 0.9, \theta_R$= 0.4]	0.2	0.4	0.6	0.8
θ_R [θ_A= 0.0033, $\theta_T = 0.2, \theta_D$= 0.3]	0.2	0.4	0.6	0.8

Table 4. Tuning Parameters (Zone 2).

	Trial 1	Trial 2	Trial 3	Trial 4
θ_A [$\theta_T = 0.1, \theta_D = 0.1, \theta_R = 0.6$]	0.0016	0.015	0.02	0.03
θ_T [θ_A=0.0016, $\theta_D = 0.1, \theta_R = 0.6$]	0.1	0.8	0.96	0.99
θ_D [θ_A= 0.0016, $\theta_T = 0.9, \theta_R$= 0.4]	0.2	0.4	0.6	0.8
θ_R [θ_A= 0.0016, $\theta_T = 0.2, \theta_D$= 0.3]	0.2	0.4	0.6	0.8

The servo response of the system can be easily enhanced by varying θ_T as shown in figures 12 and 16. Sluggish response is observed for higher value of θ_T and good servo response is observed for lower value of θ_T. This observation is analysed by keeping other tuning parameters constant while θ_T is varied. Figures clearly depict that the variations in θ_T don't have any effects on other performance attributes like disturbance rejection and robustness.

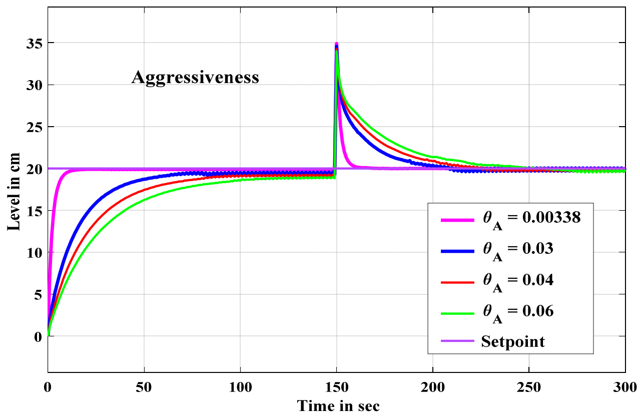

Fig. 11. Effect of θ_A on Aggressiveness (zone 1).

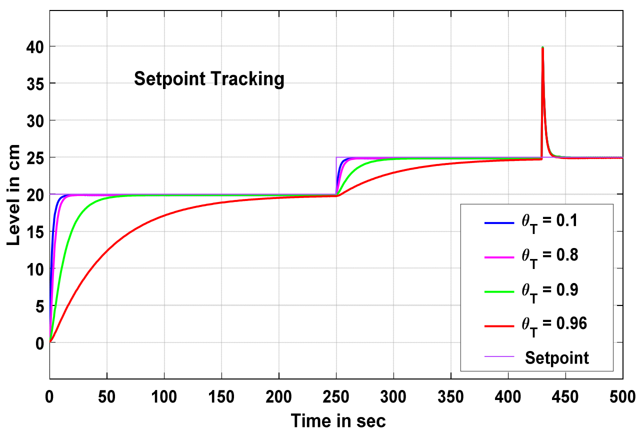

Fig. 12. Effect of θ_T on Setpoint Tracking (zone 1).

The figures 13 and 17 illustrate the effects of tuning parameter θ_D used for disturbance rejection, when the other parameters are kept constant. The time taken for rejecting the disturbance is reduced for smaller values of θ_D whereas for an increase in θ_D, the response becomes more sluggish. The effect of changing the robustness tuning parameter θ_R is shown in figures 14 and 18. The transfer function model was implemented for the plant. These responses are obtained by introducing 20% parametric error (plant/model mismatch) intentionally and the robustness of the system was analysed for different values of θ_R by keeping other tuning parameters at constant values.

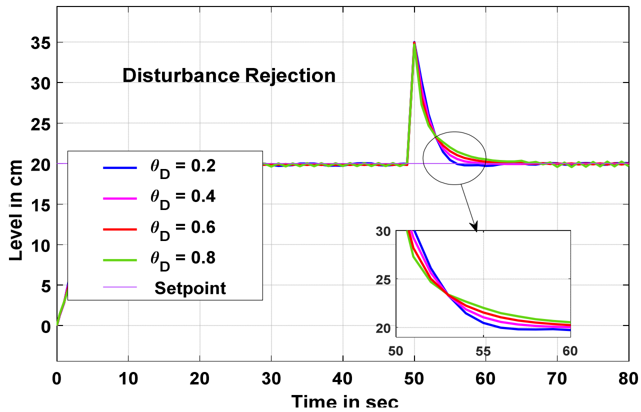

Fig. 13. Effect of θ_D on Disturbance Rejection (zone 1).

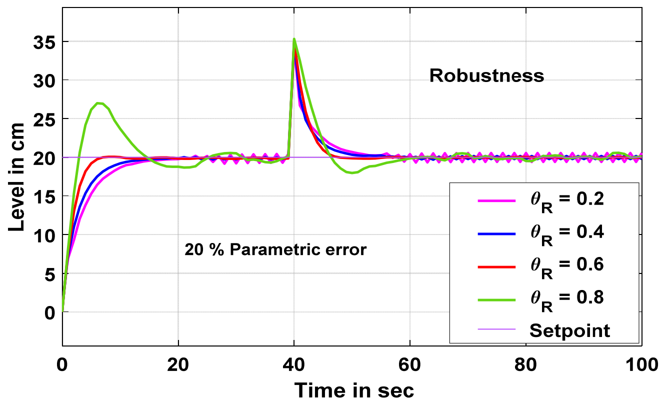

Fig. 14. Effect of θ_R on Robustness (zone 1).

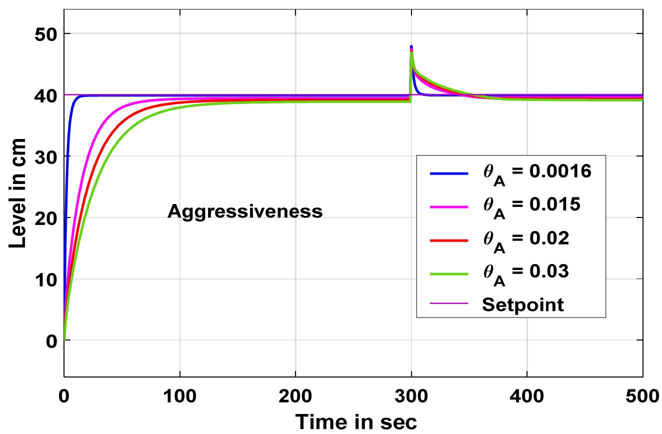

Fig. 15. Effect of θ_A on Aggressiveness (zone 2).

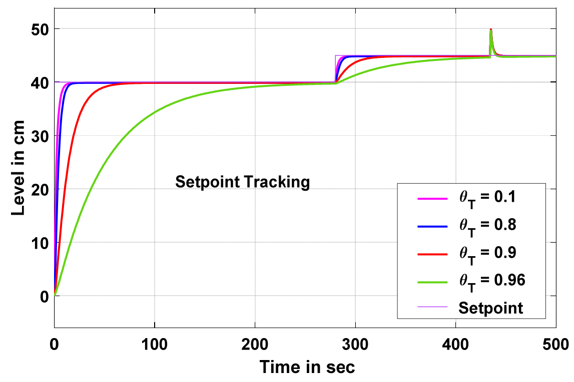

Fig. 16. Effect of θ_T on Setpoint Tracking (zone 2).

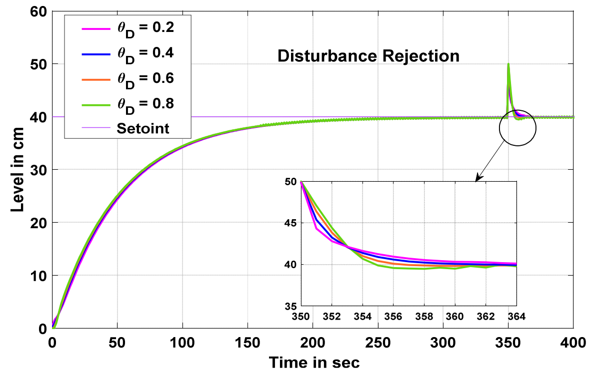

Fig. 17. Effect of θ_D on Disturbance Rejection (zone 2).

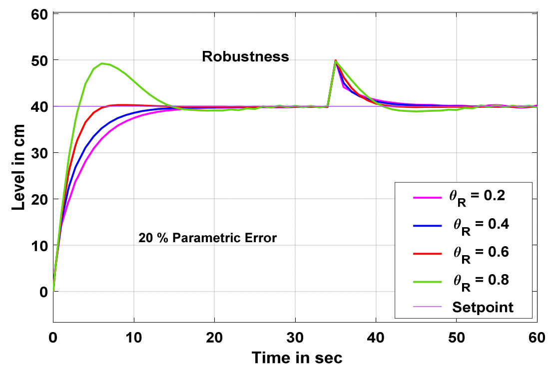

Fig. 18. Effect of θ_R on Robustness (zone 2).

For higher value of θ_R, the response becomes more sluggish, the time taken to reject the disturbance increases and settling time also increases exponentially. Good robust performance is obtained for lower values of θ_R. Optimum response is obtained when the value of θ_R is 0.6. Thus the robustness of the system can be adjusted by the parameter θ_R.

7.1 PSO for the RTDA Controller Parameters

The RTDA tuning parameter values ranging between 0 and 1 are selected using trial and error method which results in an extensive search. The influence of wrong parameter selection for RTDA controller may result in mediocre closed loop performances. To optimize this problem, a computational technique called PSO is employed. The response is analysed for randomly chosen values of tuning parameters for two trials and third trial is based on optimized values of tuning parameters using PSO as shown in Table 5 for both the operating zones.

Table 5. Tuning parameters of NN RTDA controller.

Zone 1				
	θ_A	θ_T	θ_D	θ_R
Trial 1:NN RTDA	0.0033	0.8	0.8	0.9
Trial 2:NN RTDA	0.0033	0.6	0.6	0.8
NN RTDA-PSO	0.0033	0.2452	0.0725	0.673
Zone 2				
	θ_A	θ_T	θ_D	θ_R
Trial 1:NN RTDA	0.0016	0.8	0.8	0.99
Trial 2:NN RTDA	0.0016	0.65	0.5	0.65
NN RTDA-PSO	0.0016	0.1822	0.0887	0.407

Figures 19 and 20 depict the performance of NN-RTDA controller based on PSO optimization for zone 1 and zone 2 respectively. Setpoint tracking characteristics and disturbance rejection capability are analysed and is found to be efficient when the tuning parameters are optimized using PSO. The values of θ_T, θ_D and θ_R are chosen to be high in trial 1. It is observed that the settling time is more and highly oscillating when θ_T is tuned for high value.

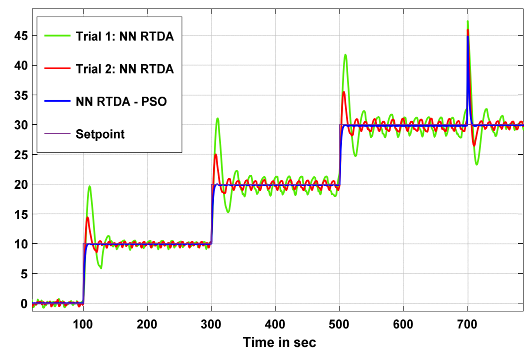

Fig. 19. Response of the system (zone 1).

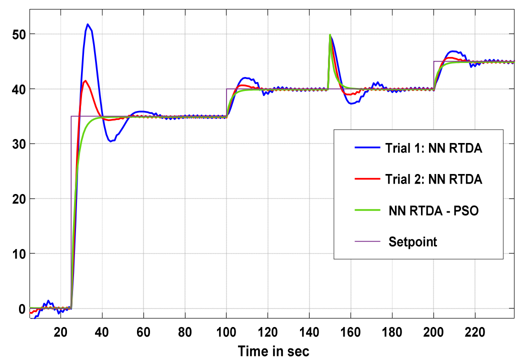

Fig. 20. Response of the system (zone 2).

The value of θ_R has a very high impact on the response of the system. There are noticeable high oscillations for high value of θ_R and hence robustness of the system is also reduced. For the high value of θ_D, the disturbances induced into the system are not rejected quickly but rather takes more time. The values for θ_T, θ_D and θ_R for both the operating zones in trial 2 are chosen. Overshoot is observed and disturbances induced are not rejected instantly. Though it doesn't touch the zenith when compared to the previous selection of values, the response seem to be quite satisfactory. However, the servo and regulatory responses eludes for optimum performance. The optimized values of tuning parameters using PSO when implemented for NN based RTDA controller tend to give very smooth responses and disturbances induced are also rejected at a faster rate. The system is also found to be more robust.

7.2 Performance Comparison of RTDA Controller Design based on Process Models

Performance metrics for both the operating zones is given in Table 6 and Table 7. Based on the observations made, it is concluded that the NARX neural model based RTDA controller for the conical tank is better when compared to TF model based RTDA controller in terms of ISE and gain margin and efficient performance is evident, when the tuning parameters of NN-RTDA controller is optimised using PSO.

The robustness analysis is very important as the process model used in controller design is often a replica of the industrial process. Usually, gain margin is a measure of stability and is also related to the robust performance of the closed loop system. Robustness analysis using Bode's plot for conical tank system with RTDA controller using two different models are analysed in both the operating zones. The TF models were developed using Levenberg-Marquardt algorithm and back propagation algorithm in MATLAB using the same experimental data sets.

Table 6. Performance metrics for zone 1 (h = 20 cm)

Performance Metrics / Tuning Parameters	TF Model based RTDA controller	NN Model based RTDA controller	PSO-NN Model RTDA controller
	$\theta_T = 0.1$ $\theta_R = 0.2$ $\theta_D = 0.7$	$\theta_T = 0.1$ $\theta_R = 0.2$ $\theta_D = 0.7$	$\theta_T = 0.2452$ $\theta_R = 0.673$ $\theta_D = 0.0725$
Settling Time (sec)	18	14	11
Peak Overshoot (%)	0.846	0.386	0.627
Rise Time (sec)	7.730	5.106	4.553
ISE	428.688	404.123	383.900
Gain Margin	12.4	22.6	26.5

Table 7. Performance metrics for zone 2 (h = 40 cm)

Metrics / Tuning Parameters	TF Model based RTDA controller	NN Model based RTDA controller	PSO-NN Model RTDA controller
	$\theta_T = 0.2$ $\theta_R = 0.1$ $\theta_D = 0.7$	$\theta_T = 0.2$ $\theta_R = 0.1$ $\theta_D = 0.7$	$\theta_T = 0.1822$ $\theta_R = 0.4063$ $\theta_D = 0.0887$
Settling Time (sec)	20	16	13
Peak Overshoot (%)	0.407	0.407	0.438
Rise Time (sec)	8.046	5.457	4.383
ISE	1.5769e+03	1.4173e+03	1.2505e+03
Gain Margin	12.8	13.3	14.6

The value of tuning parameters θ_A, θ_T , θ_D , θ_R were kept same for both the controller design. The gain margin for NN-RTDA controller design seems to be comparatively higher than that of TF-RTDA controller. It is also observed that proper selection of RTDA tuning parameters using PSO algorithm further increases the gain margin and ensures better robustness for the control system.

7.3 Performance Comparison based on different Control Schemes

The NN-RTDA controller optimized using PSO is compared with neural network model based MPC and PI-IMC tuned controller as shown in figures 21 and 22. The prediction horizon P for NN-MPC is kept as two similar to that of NN-RTDA controller design for fair comparison whereas control horizon is kept as one and weighting on the control action is not considered. In PI-IMC controller, the value of filter time constant λ plays a vital role as it has control over the robustness and closed loop performance of the system. The value of λ is tuned in such a way that there is a trade-off between the performance and robustness of the system. Based on the value of λ, K_p and K_i are selected as (K_p =30, K_i =0.1) for zone 1 and (K_p =15, K_i =0.0257) for zone 2. The value of λ for zone 1 and zone 2 is chosen as 9.8 and 26 respectively.

Encountering real challenging scenario, step changes are given in reference so as to analyse whether the controller is able to sustain setpoint tracking characteristics. It is observed that despite of the changes in reference, all the control schemes are able to track the changes and maintain a desired setpoint. The setpoint is varied from 0 to 10 cm then 10cm to 20cm for zone 1 and 35cm to 40cm, 40cm to 45cm and 45cm to 35cm for zone 2. NN RTDA controller shows improved performance compared to NN-MPC and PI-IMC controller in terms of speed of output response. The NN-MPC controller exhibits fine oscillations about its steady state but performed well when compared to PI-IMC controller, where the response overshooted before it settled down about its settling point. NN RTDA controller optimized using PSO was able to maintain the level at a given setpoint even in highly nonlinear zone (zone 2).

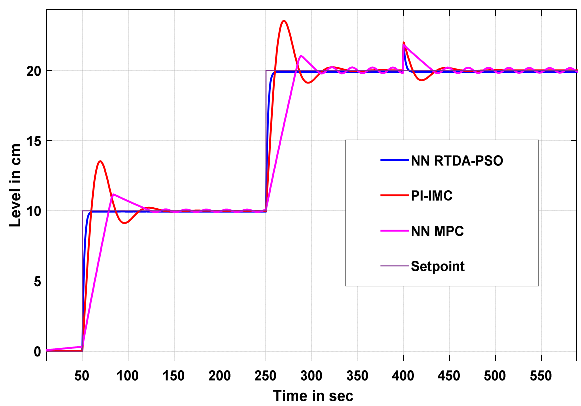

Fig. 21. Setpoint tracking and disturbance rejection capability of controllers for zone 1.

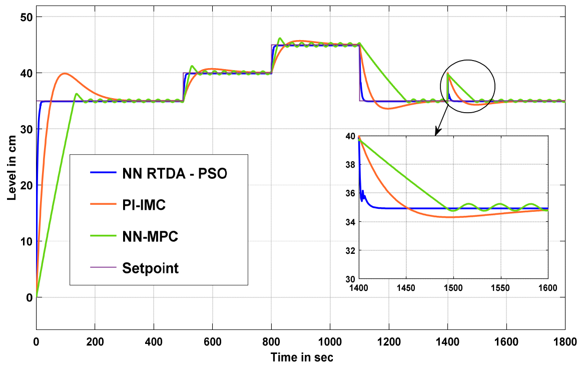

Fig. 22. Setpoint tracking and disturbance rejection capability of controllers for zone 2

All the three control schemes are assessed for disturbance rejection capability in the presence of disturbance of flow rate. Disturbance rejection capability of NN-MPC is meagre compared to NN-RTDA controller whereas PI-IMC controller showed a sluggish response for both the zones. Based on the ISE comparison for all the three control schemes, NN RTDA-PSO controller was pre-eminent for controlling the nonlinear system as shown in Table 8.

Table 8. Performance metrics based on ISE

	Zone 1	Zone 2
NN RTDA – PSO	383.9002	1.2305e+03
NN MPC	866.2214	1.5672e+03
PI-IMC	2.0193e+03	2.0688e+03

8. CONCLUSIONS

The level control for the nonlinear conical tank process is done using NN model based RTDA controller, with its four exclusive tuning parameters. The closed loop performances are analysed in detail which elucidates that each tuning parameter is unique and can be adjusted independently which is not possible with other controllers. Performance attributes namely, setpoint tracking, disturbance rejection, robustness and aggressiveness for the closed loop system is easily altered by using appropriate tuning parameters without affecting each other. To efficiently predict the future plant behaviour, NARX neural model obtained from the empirical data set is used in RTDA controller design. The simulation results prove that NN model based RTDA controller performance is more efficient in terms of ISE, than the TF model based controller. In spite of large plant-model mismatch, NN model based RTDA controller showed significant improvement in terms of increased robustness. Instead of the usual random pick and selecting the values of the tuning parameters, a highly efficient Particle Swarm Optimization (PSO) is employed to identify the optimal values to give a precise, distinct solution for accentuating and enhancing aggressiveness, robustness, set point tracking and disturbance rejection in the closed loop response. NN based RTDA controller which is optimized using PSO is further compared with PI- IMC controller and NN MPC. NN RTDA controller gives better performance like NN MPC. RTDA controller solves all the tuning associated problems that exist in PID and MPC. With its expedient tuning parameters, NN RTDA controller improves the aggressiveness, robustness, tracking, and regulation properties of the system independently to get preferred response which is not possible using the other two control schemes. It has one limitation it cannot handle input output constraints effectively like MPC. In future, deep learning algorithm such as LSTM neural networks can be the next leveraging technology for finer prediction of the output vector in RTDA controller design.

REFERENCES

Abubakar S., Muntaqa D., Kabiru Aminu., Salahud deen G Ahmad. (2015). Modelling and Control of Dynamical Systems using Neural network-A Review. *International Journal of Soft Computing and Engineering* 5 (4).

Anbarasan K., Srinivasan K. (2015). Design of RTDA Controller for Industrial Process using SOPDT Model with Minimum or Non-Minimum Zero. *ISA Transactions 57*, 231–244.

Antonius Yudi Sendjaja., Zhen., Si Si How., Vinay Kariwala. (2011). Analysis and tuning of RTD-A controllers. *Industrial and Engineering Chemistry Research* 50(6), 3415-3425.

Antsaklis P J. (1990). Neural Networks in Control Systems. *The IEEE Control Systems Magazine* 10 (3), 3-87.

Ashok Kumar D., Murugan S. (2018).Performance Analysis of NARX neural Network Back Propagation Algorithm by various Training Functions with Tracking Signal Approach for Time series Data. *International Journal of Engineering Sciences and Research Technology* 3(4), 312-326.

Bagyaveereswaran V., Tushar D Mathur., Sukrit Gupta., P. Arulmozhivarman. (2016). Performance comparison of next generation controller and MPC in real time for a SISO process with low cost DAQ unit. *Alexandria engineering journal* 55, 2515-2524.

Boyang Zhang., Xiuxia Sun., Shuguang Liu., Xiongfeng Deng. (2019). Recurrent neural network based model predictive control for multiple unmanned quadrotor formation flight. *International journal of aerospace engineering*.

Diwakar B., M Gokul Anandh., R Brindaa., S R Jayashree Devi., P Aravind. (2015). IMC based design of PI controller for real time pressure process station. *International journal of innovative research in electrical, electronics, instrumentation and control engineering*. 3(10).

Eswaran., V.Suresh Kumar. (2017). Particle Swarm Optimization (PSO)-based tuning technique for PI controller for management of a distributed static synchronous compensator for improved dynamic response and power quality. *Journal of Applied Research and Technology*, 173-189.

Geetha M., Balajee K A., Jovitha Jerome.(2012).Optimal tuning of virtual feedback PID controller for a continuous stirred tank reactor (CSTR) using particle swarm optimization (PSO) algorithm. *IEEE International conference on Advances in Engineering, Science and Management*.

Geetha Mani., Arun Kumar Pinagapani. (2016). Design and implementation of a preemptive disturbance rejection controller for PEM fuel cell air-feed system subject to load changes. *Journal of electrical engineering and technology* 11, 1921- 1928.

Gomez-Ruiz., Jerez-Aragones., Munoz-Perez., E.Alba Conejo .(2014). A Neural Network Based Model for Prognosis of early breast cancer. *Applied Intelligence*, 20, 231-238.

Jean Thomas. (2014). Model-based power plant master control, *Automatika* 55(3), 228-238.

Kapil Mukati., Michael Rasch., Babatunde A Ogunnaike. (2006).An alternative *structure* for next generation regulator controllers. Part I: Basic theory for design, development and implementation. *Journal of process control* 16,499-509.

Kapil Mukati., Michael Rasch., Babatunde Ogunnaike.(2009). An alternative structure for next generation regulator controllers. Part II Stability analysis, tuning rules and experimental validation. *Journal of process control* 19, 272-287.

Latha k., Rajinikanth V., P. M. Surekha.(2013). PSO-based PID controller design for a class of stable and unstable systems. *ISRN Artificial intelligence,* 1-12.

Mahmud Iwan Solihin., Lee Fook Tack and Moey Leap Kean. (2011). Tuning of PID controller using Particle swarm Optimization (PSO). *Proceedings of the international conference on advanced science*, Malaysia.

Martin T Hagan., Howard B Demuth., Orlando De Jesus. (2002).An introduction to the use of neural networks in control systems. *International journal of robust and nonlinear control*, 1-23.

Mohd Noor Nasriq Nordin., Mohd Hezri Rahiman., Ramli Adnan., Zakiah Mohd Yusoff., Ihsam Mohd. (2012). Optimization of NARX lag space selection for a multilayer perceptron model of a down flowing steam distillation system using particle swarm optimization. *IEEE international conference on signal processing and its applications,* Malaysia.

Narendra K.S and K Parthasarathy. (1990). Identification and Control of Dynamical Systems using Neural Networks. *IEEE Transactions on Neural Networks* 1, 4-27.

Nicolas Lanzetti., Ying Zhoa Lian., Andrea Cortinovis., Luis Domingeuz., Mehmet Mercangoz., Colin Jones.(2019) Recurrent neural network based MPC for process industries. *18th European control conference*.

Prashant Mhaskar., Nael H. El-Farra and Panagiotis (2004).A method for PID controller tuning using nonlinear control techniques," *Proceedings of the 2004 American Control Conference*, Boston, Massachusetts.

Srinivasan K., Singh J., Anbarasan K., Paik R., Medhi R. And Choudhury K.D (2012).MPC And RTDA Controller for Fopdt & Sopdt Process. *Journal of Information Systems and Communication* 3, 109-113.

Sjoberg J., Q Zhang, L Ljung., A. Benveniste., B. Delyon., P-Y Glorennec., H Hjalmarsson, A juditsky (1995). Non-Linear black-box modelling in system identification: a unified overview, *Automatica*, 1694-1724.

Travieso Torres J.C., Duarte Mermoud M.A.,Beytia (2017). Combining fractional order operators and adaptive passivity-based controllers: An application to the level regulation of a conical tank, *Journal of Control Engineering and Applied Informatics* 1(2), 3-10.

Vijayalakshmi, S., Manamalli, D., PalaniKumar, G. (2014), Closed loop experimental validation of linear parameter varying model with adaptive PI controller for conical tank System, *Journal of Control Engineering and Applied Informatics* 16(4), 12-19.

Vasile Cirtoaje. (2020). A practical unified algorithm of P IMC type, Processes 8, 165.

Venkatesh Sivanandam., Ramkumar Kannan. (2018). Modelling and controller design for a conical tank process using radial basis function neural network. *Acta Polytechnica Hungarica*. 1-5.

Wills M.J., G.A Montague., Massimo., M.T. Tham., Morris (1992). Artificial neural networks in process estimation and control. *Automatica* 28, 1181-1187.

Yu Sun., Jizheng Chu., Minghui Chu. (2018)RTD-A Control for General Multivariable Systems. *Arabian Journal for Science and Engineering 43*, 5891-5903.