Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 2 | Pages 327-334 | e-ISSN: 2347-470X

Performance Analysis of Heat Exchanger System Using Deep Learning Controller

Bhawesh Prasad[1], Raj Kumar[2] and Manmohan Singh[3]

[1]*Research Scholar, Department of EIE, Sant Longowal Institute of Engineering and Technology, Longowal-148106, India*
[2,3]*Associate Prof., Department of EIE, Sant Longowal Institute of Engineering and Technology, Longowal-148106, India*

*__Correspondence:__ Bhawesh Prasad; Email: bpsliet2010@gmail.com

ABSTRACT- Conventional PID controllers have utilised in most of the process industries. Despite being the most used controller, the traditional PID controller suffers from several disadvantages. Due to rapid development in the field of the process control system, various controllers have been developed that try to overcome the limitations of the PID controller. In this paper, a heat exchanger system has been simulated, and the generated data has been used to train a deep learning-based controller using Backpropagation. The obtained results are compared with the conventional controller on several metrics, including time response, performance indices, frequency response etc. The proposed model outperforms the conventional controller on all the evaluation metrics.

Keywords: Artificial neural networks, Deep Learning controller, PID controller, Heat Exchanger

ARTICLE INFORMATION

Author(s): Bhawesh Prasad, Raj Kumar and Manmohan Singh

Special Issue Editor: Dr. S. Muthubalaji
Received: 17/05/2022; **Accepted**: 16/06/2022; **Published**: 30/06/2022;
e-ISSN: 2347-470X;
Paper Id: 0322SI-IJEER-2022-25;
Citation: 10.37391/IJEER.100244
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-10/ijeer-100244.html

This article belongs to the Special Issue on **Intervention of Electrical, Electronics & Communication Engineering in Sustainable Development**

1. INTRODUCTION

The Control system is the backbone of modern-day industries. It is necessary to move toward automation [1] [2]. The methods for designing a feedback control system are well-established. These include classical linear control system design, adaptive control, H-∞ control, robust control, non-linear control techniques etc. Controls like ANN and Fuzzy logic controls are becoming popular due to their model-free control architecture. [3][4] [5]. The closed-loop neural network control applications differ from open-loop classification and image processing applications. Werbos [6] was the first to use NN in a closed-loop control system.

In recent years, the advances in the field of artificial intelligence have not left the field of process control untouched and the development of several deep learning-based controllers are the evidence. Deep learning [7][8] has recently gotten a lot of interest from a variety of sources. Deep learning has more hidden layers and neurons than traditional neural networks, which allows it to increase learning performance. Deep learning algorithms can address vast and complicated issues that traditional neural networks couldn't. As a result, deep learning has been used to solve a variety of pattern identification and classification issues. Two layer NN is sufficient for a feedback control system [4] but the addition of hidden layers improves the stability of the feedback control system and produces a damping effect [3]. However, to the authors' best knowledge, only few results have been published in the automatic control field. In a recent study [3], the deep learning control technique is used for the second order linear plant model and needs to extend for real non-linear systems. In [9] PID controller is replaced with a deep learning controller via a DBN for speed control of the DC motor. A similar approach is used in [10], but hardware requirement is the main disadvantage of DBNs. Robotic manipulation controllers have recently used Deep RL and, based on RL, have also been implemented in [11]. Also, RL requires large data and is computationally expensive. Some other applications includes including non-linear system identification and model reduction [12] [13], model predictive controller [14][15][16], architecture and size of the neural network, and comparative study [17] [18].

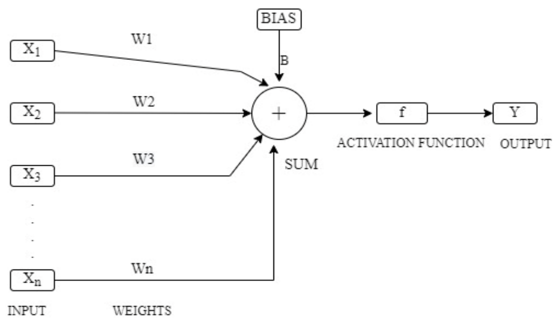

Figure 1: ANN Functioning

In this paper, a deep learning controller has been designed for temperature control of the heat exchanger system. The heat exchanger system, which is a non-linear system and needs adequate controller that can overcome the limitstions of PID controller and can also be used for non-linear systems. The authors of a study [3] concluded that when subjected to a second-order linear plant in a closed-loop control system, their proposed deep learning controller worked best in the training phase as compared to works that use DBNs, RLs, and LSTMSNNs while keeping DL advantages. After an extensive survey of related literature, several lacune were discovered, one of them being the field of non-linear plant control problem, which very few studies have covered. In the present work, a heat exchanger system, which is a non-linear system is considered. To study the performance of deep learning model on a non-linear problem, the heat exchanger is simulated and the results of proposed deep learning controller (DLC) are compared with the conventional PID controller. The rest of this paper is organised as follows. *Section 2* describes the second-order plant and its mathematical modelling. The DLC is described in *section 3*. The results and simulations are presented in *section 4*. Conclusion and discussion are presented in *section 5*.

2. MATHEMATICAL MODELLING OF HEAT EXCHANGER

Heat exchangers (HEs) facilitate heat transfer between two fluids of differing temperatures while preventing them from mingling. HEs are key devices used in various thermal applications in the chemical process industries [19][20]. The heat exchange process is influenced by a number of factors like flow pattern, flow rates of fluids, temperature difference, the heat transfer area etc. Different types of HEs are spiral tube, double pipe and shell and tube. Shell and tube type heat exchanger is commonly used. It contains many tubes packed in a shell with their axes parallel to the shell [21]. The temperature of outlet fluids can be controlled by the conventional controller or any other advanced controller [22]. Consider a HEs as shown in *figure 2* [23][24].

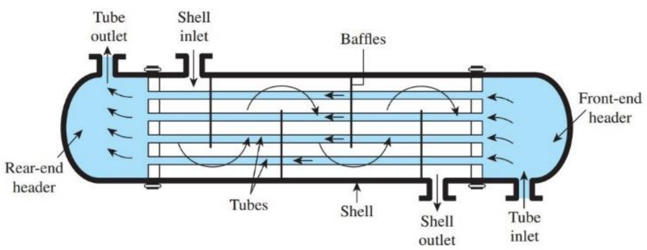

Figure 2 Shell and tube type heat exchanger

The following assumptions were used to develop the model [25]:

(i) The volume, density and heat capacity of both fluids remain constant.

(ii) Constant heat transfer area and heat transfer coefficient of both fluids

(iii) Both cold and hot fluid streams are well-mixed, and the outlet temperature Tco and Tho approximate the temperature of the cold and hot streams inside the tube.

The relationship between the U nd Q is

$$Q = UA\Delta T \tag{1}$$

The energy balance equations in the shell side and tube side are

$$\frac{dT_{co}(t)}{dt} = \frac{F_c(T_{ci}(t)-T_{co}(t))}{V_c} + \frac{U_cA_c(T_{ho}(t)-T_{co}(t))}{\rho_cV_cC_{pc}} \tag{2}$$

$$\frac{dT_{ho}(t)}{dt} = \frac{F_h(T_{hi}(t)-T_{ho}(t))}{V_h} + \frac{U_hA_h(T_{co}(t)-T_{ho}(t))}{\rho_hV_hC_{ph}} \tag{3}$$

The Taylor series expansion of the above equations only for linear terms and substituting steady-state values are

$$\frac{dT_{co}(t)}{dt} = -11.5T_{co} + 4.65T_{ho} - 7.18F_c + 6.85T_{ci} + 0F_h + 0T_{hi} \tag{4}$$

$$\frac{dT_{ho}(t)}{dt} = 1.15T_{co} - 6.248\,T_{ho} + 0\,F_c + 0\,T_{ci} + 2.451\,F_h + 5.098\,T_{hi} \tag{5}$$

The state-space representation is

$$\frac{dx}{dt} = Ax + Bu \tag{6}$$

$$y = Cx + Du \tag{7}$$

Where x, u, y, A, B, C, and D is the state variable, input variable, output variable, state matrix, input matrix, output matrix, and translational matrix respectively.

$$A = \begin{bmatrix} -11.5 & 4.65 \\ 1.15 & -6.248 \end{bmatrix} \tag{8}$$

$$B = \begin{bmatrix} 0 & -7.18 & 6.85 & 0 \\ 2.451 & 0 & 0 & 5.098 \end{bmatrix} \tag{9}$$

$$C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{10}$$

$$D = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{11}$$

The overall transfer function of the process becomes

$$\frac{T_{co}(s)}{F_h(s)} = \frac{11.39715}{s^2+17.748s+66.4125} \tag{12}$$

This equation represents the transfer function of a shell and tube type heat exchanger system where the control variable is outlet temperature of cold water and the manipulated variable is input flow rate of hot water. FOPDT model [26] of heat exchanger system is represented as

$$\frac{T_{co}(s)}{F_h(s)} = \frac{0.1716}{0.205757s+1}e^{-0.070521s} \tag{13}$$

The pade approximated model [27] of above system is

$$\frac{T_{co}(s)}{F_h(s)} = \frac{-6.2119s+171.6}{7.3196s^2+238.4s+1000} \tag{14}$$

3. DEEP LEARNING CONTROLLER

Input-output data of the HE system given by equations 4 and 5 is used to train a DLC offline. The backpropagation (BP) procedure, which starts from the output layer and propagates backwards until it reaches the hidden layer adjacent to the input layer to update the weights, is the most common training algorithm for neural network weights. So, we started by setting a random value to weight by random weight generation and then propagated forward. Now error is calculated at output and

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 2 | Pages 327-334 | e-ISSN: 2347-470X

then is backpropagated to adjust the weight. We try to optimise the performance [28] given in *equation (15).*

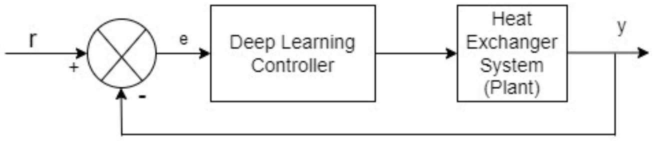

Figure 3: Feedback system with a deep learning controller [29]

The instantaneous value of the error as ${e_k}^2(n)$ where k represents kth neuron.

$${e_k}^2(n) = \sum_{j=N_1}^{N_2}\left(y_r(t+j) - y_m(t+j)\right)^2 + \rho\sum_{j=1}^{N_u}\left(u'(t+j-1) - u'(t+j-2)\right)^2 \quad (15)$$

Where ${e_k}^2(n)$= cost function
N_1= 1= minimum prediction horizon
Y_r=desired response trajectory
N_2= cost horizon
u'= tentative control signal
Y_m=network model response
ρ= controller weighting factor
N_u=control horizon

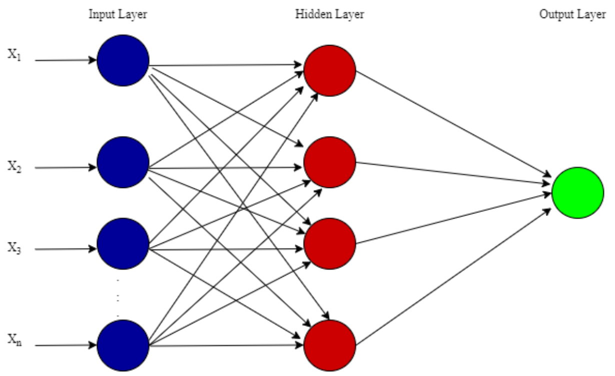

Figure 4: ANN structure

The cost horizon, control horizon, controller weighting factor, training samples, size of hidden layers taken as 5, and training epoch are taken for deep learning parameters. We have generated 8000 training samples, and these training samples have been selected from random number generation process. So, basically, we have generated input and target data for the neural network training in this step. There are various types of BP depending on the update strategy, the most popular being BP based on gradient descent (GD)

The total error energy is represented as

$$E(n) = \sum_{j\in C}{e_j}^2(n) \quad (16)$$

The average squared error energy is [30]

$$E_{av}(n) = \frac{1}{N}\sum_{n=1}^{N}E(n) \quad (17)$$

The goal of the learning process is to update the network's parameters to reduce Eav.

$$\vartheta_j(n) = \sum_{i=0}^{m}w_{ki}(n)y_i(n) \quad (18)$$

Where the vj (n) associated with neuron k is induced local field. So, we can write

$$y_k(n) = \varphi_k\vartheta_k(n) \quad (19)$$

$\partial E(n)/\partial w_{ki}(n)$ can be written as

$$\frac{\partial E(n)}{\partial w_{ki}(n)} = \frac{\partial E(n)}{\partial e_k(n)}\frac{\partial e_k(n)}{\partial y_k(n)}\frac{\partial y_k(n)}{\partial \vartheta_k(n)}\frac{\partial \vartheta_k(n)}{\partial w_{ki}(n)} \quad (20)$$

$$\frac{\partial E(n)}{\partial w_{ki}(n)} = -e_k(n)\varphi_k'(\vartheta_k(n))y_i(n) \quad (21)$$

The partial derivatives $\frac{\partial E(n)}{\partial w_{ki}(n)}$ represents a sensitive factor. The correction $\Delta w_{ji}(n)$ applied to $w_{ki}(n)$ is defined by the delta rule

$$\Delta w_{ki}(n) = -\eta\frac{\partial E(n)}{\partial w_{ki}(n)} \quad (22)$$

Using equations (41), equation (42) can be written as

$$\Delta w_{ki}(n) = \eta\,\delta_k(n)y_i(n) \quad (23)$$

The local gradient $\delta_k(n)$ can be written as

$$\delta_k(n) = -\frac{\partial E(n)}{\partial \vartheta_k(n)} = -\frac{\partial e_k(n)}{\partial e_k(n)}\frac{\partial e_k(n)}{\partial y_k(n)}\frac{\partial y_k(n)}{\partial \vartheta_k(n)} \quad (24)$$

$$\delta_k(n) = e_k(n)\varphi_k'(\vartheta_k(n)) \quad (25)$$

The local gradient $\delta_k(n)$ for output neuron k is given by equation (25). When neuron k is not belongs to output layer and neuron k belongs to hidden layer then error signal would have to be determined recursively and may redefine the local gradient $\delta_k(n)$ j as

$$\delta_k(n) = -\frac{\partial E(n)}{\partial y_k(n)}\frac{\partial y_k(n)}{\partial \vartheta_k(n)} \quad (26)$$

$$\delta_k(n) = -\frac{\partial E(n)}{\partial y_k(n)}\varphi_k'(\vartheta_k(n)) \quad (27)$$

$$\delta_j(n) = \varphi_j'(\vartheta_j(n))\sum_k\delta_k(n)w_{kj}(n) \quad (28)$$

The hyperbolic tangent activation function is antisymmetric about $\vartheta = 0$ and has better and faster learning capabilities. It is defined as

$$\varphi_k(\vartheta_k(n)) = a\tanh(bv_k(n)),\ (a,b) > 0 \quad (29)$$

Where a,b is constant.

The proposed model is trained usimg the randomly generated 8000 samples. These samples are further divided into training, validation and testing. Number of training samples is 70% of the total samples, while the validation and testing data is made of 15% samples each. These samples are selected randomly from the total samples and differ each time we train the model.

4. SIMULATION RESULTS AND ANALYSIS

The MATLAB (Academic License no. 1075356) is used to run all simulations. Second-order non-linear plant (HE) given by *equations 4* and *5* is simulated. The best validation performance of the deep learning controller during training is shown in *figure 4*. The evaluation metric for the proposed network is set as mean squared error (MSE). The loss, thus is defined in terms of mse, where loss is the MSE of the output of NN and the real value. As it can be observed, at the initial phase of the training, the loss is high (~10), but as the training progresses through the epochs, the loss diminishes and finally settles at around 10-8. The losses observed during validation also follow the same curve. This implies that the model is trained perfectly and any kind of over or under fitting is absent. The testing losses also convey the same findings, i.e., the model is perfectly trained.

The best performance obtained is 5.5*10-09 at the 2000 epoch. The performance of the controller during training is shown in *figure 5*. This graphic compares the output of the NN to the real value. These two are coinciding, showing that the neural network is of good quality. Regression value during the training, validation and test all are 1, which indicate the best regression value, and there is a close relationship between the input and output. Input, plant output, NN output and error during training is shown in *figure 7*. The use of several samples and splits in the dataset allows us to assess the model performance. The training set of data simulated is used to train the model. NN output follows the plant output and during this steps, error obtained is 2*10-03. Similarly during testing and validation, error is 6*10-04 for both case is shown in figure 8 and *figure 9*. Training state is shown in *figure 10*. Validation data is fed into the model during training to provide it new data that hasn't seen before. A check was also put in place to detect if the learning of the model was becoming stagnant amd hence, the model was not learning. This was implemented by placing a “val_fail” function. The model peaked at the 2000 epochs.

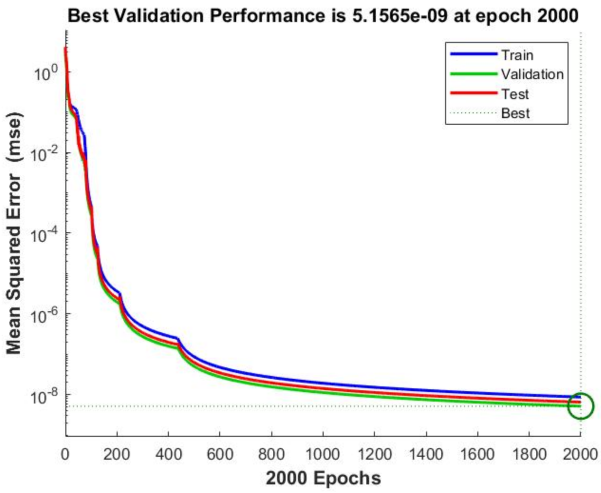

Figure 5: Performance of the deep learning controller

Heat exchanger performance has been evaluated using MATLAB and Simulink toolbox.

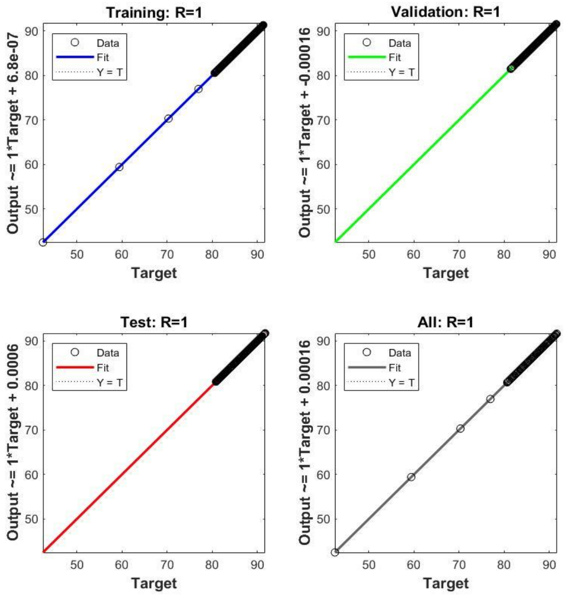

Figure 6 Regression of the controller during training

In this paper, the performance of heat exchanger has been classified into time response characteristics, signal statics and frequency response characteristics. These performance parameters are shown in *table 1, 2* and *3* and *figure 15, 16* and *17* for frequency response. The step response of magnitude 85 of the heat exchanger system with a deep learning controller has been shown in *figure 11*.

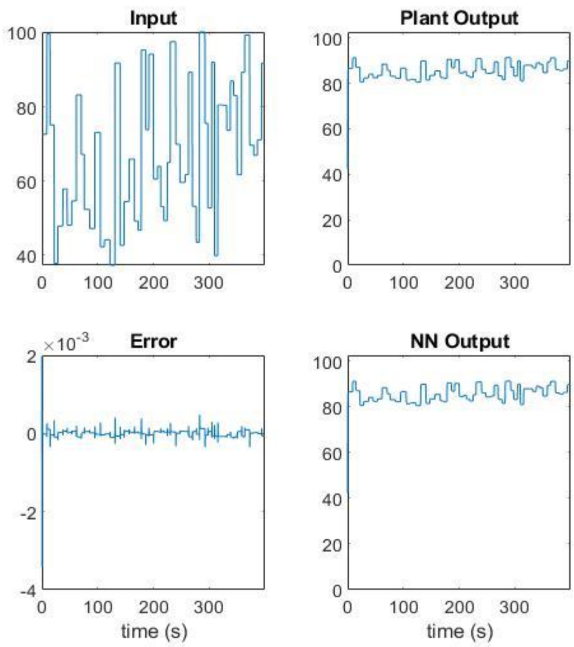

Figure 7: Training data

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 10, Issue 2 | Pages 327-334 | e-ISSN: 2347-470X

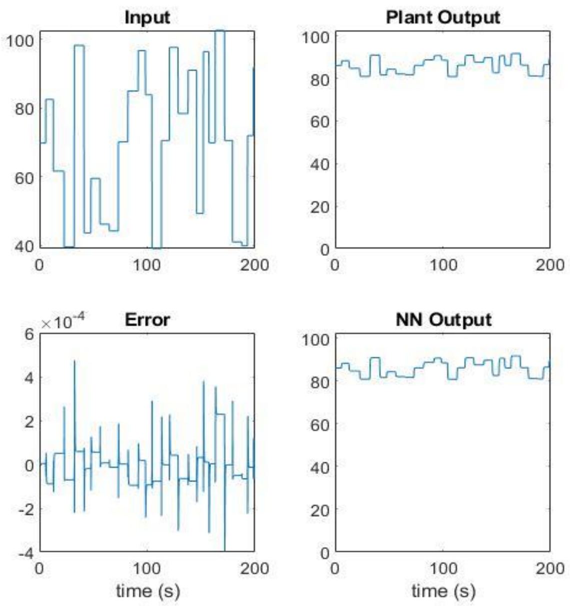

Figure 8: Testing data

Also, the step response of the system has been compared with a different conventional controller in *figure 12*. The same trained deep learning controller has been used for system response to a step of magnitude changes from 85 to 90 as shown in *figure 13*. The time response characteristics and signal statistics are listed in *table 1 and 2*. The frequency response analysis has also been done and the bode plot, Nichols digarma, Nyquist plot and pole zero map of the system have been plotted with a deep learning controller, as shown in *figure 14, 15,16* and *17* respectively. When the grid of pole zero map is turned on, it shows lines with a constant damping ratio (zeta) and lines with a constant natural frequency (wn).

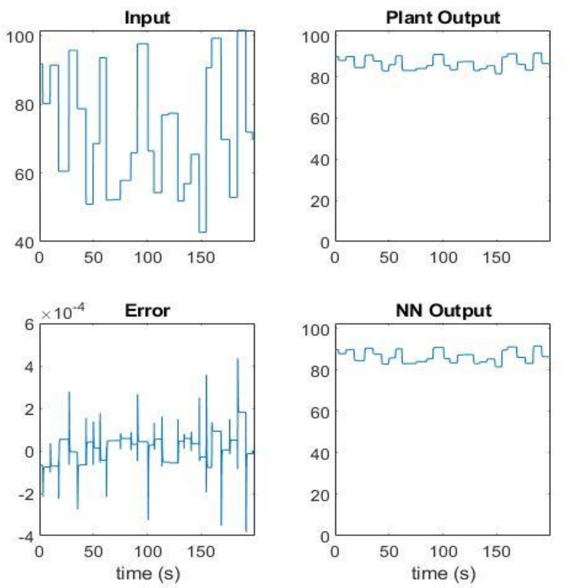

Figure 9: Validation data

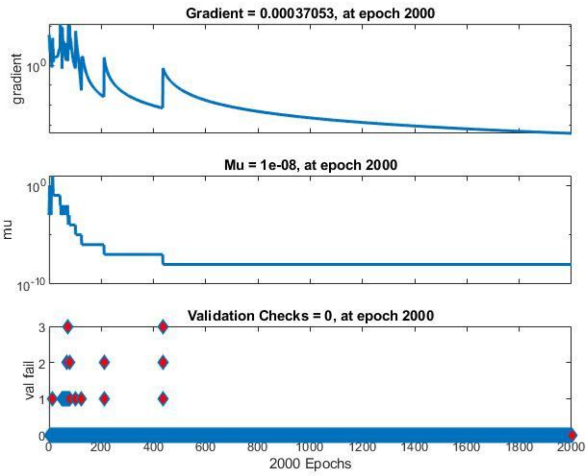

Figure 10: Training states diagram of neural network

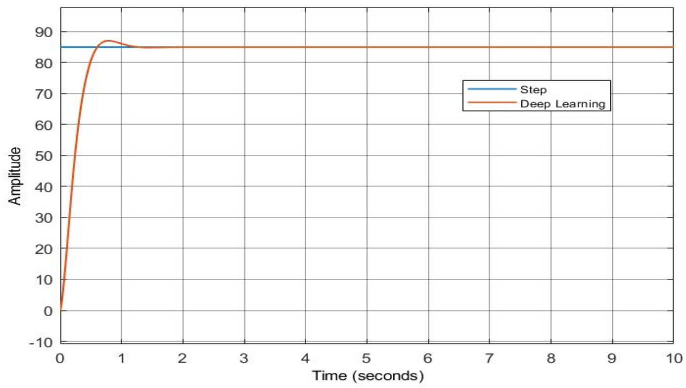

Figure 11: Simulation response of a system using deep learning controller for a set point of 85

The Nichols chart is an excellent tool for measuring a feedback system's stability and frequency response of closed loop system. The nyquist plot in the complex plane illustrates the relationship between the magnitude of frequency variation and the phase of the transfer function. The gain margin and phase margin have been evaluated and tabulated in the *table 3*. Results in *table 3* and simulation responses in *figure 10, 11, 12* and *13* indicate that proposed deep learning controller provides optimum settling time, eliminate overshoot in comparison with conventional controllers.

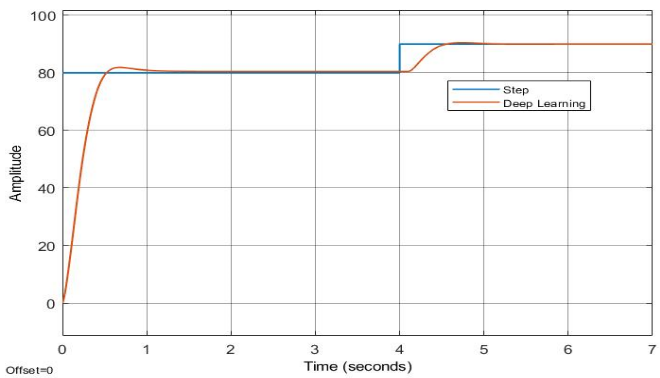

Figure 12: Response of the system using deep learning controller for a step change in set point

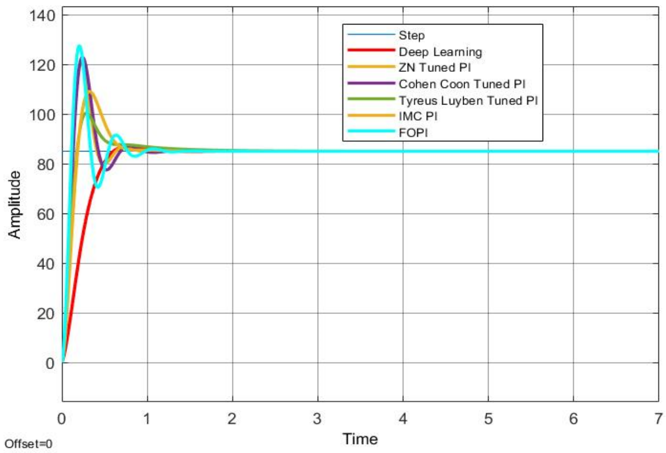

Figure 13: 9 Response of the system using deep learning controller and conventional controller

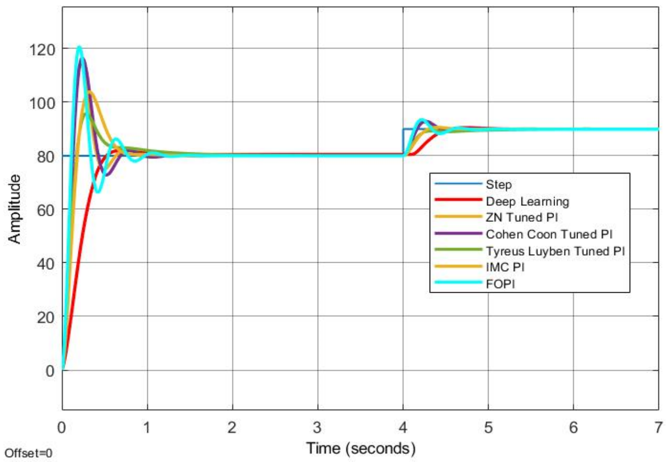

Figure 14: Response of the system using deep learning controller and conventional controller for a step change in set point

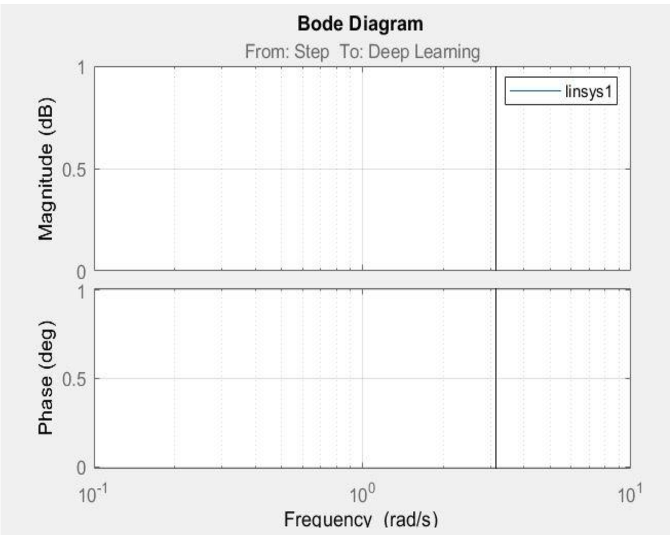

Figure 15: Bode plot of the system using deep learning controller

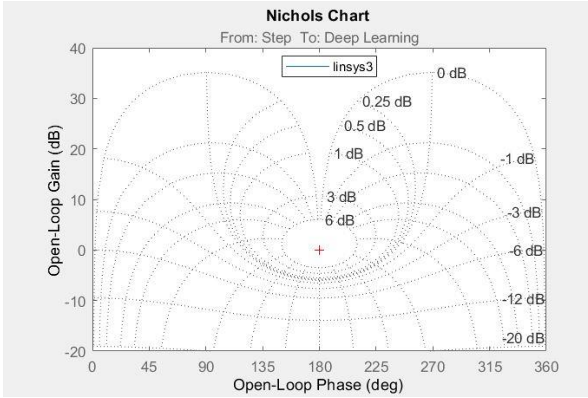

Figure 16: Nichols Chart of the system using deep learning controller

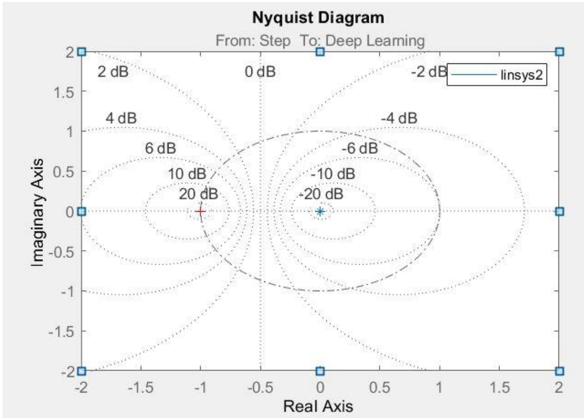

Figure 17: Nyquist diagram of the system using deep learning controller

Table 1: Time response performance characteristics of the system

Sr. No.	Parameters	Deep Learning Controller
1	Maximum value	87.07 unit
2	Rise time	380 ms
3	Peak time	790 ms
4	Peak overhoot (%)	2.5

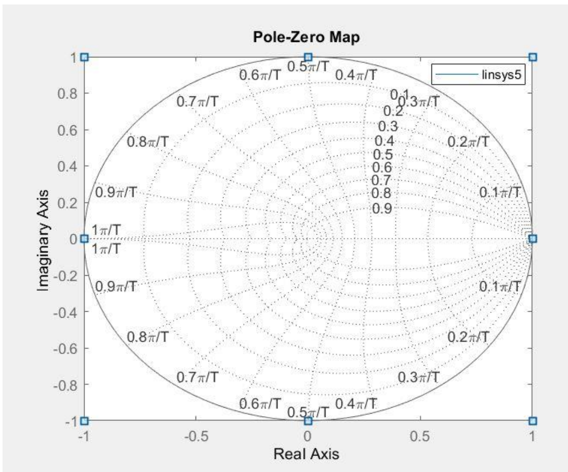

Figure 18: Pole zero map of the system using deep learning controller

Table 2: Signal statistics of the system

Sr. Number	Parameters	Deep Learning Controller
1	Mean value	83.60 unit
2	Median value	85.00 unit
3	RMS value	84.00 unit
4	Variance	1.96
5	Standard deviation	1.4

Table 3: Gain and phase margin of the controller

Parameters	Gain margin (dB)	Phase margin (deg)
Deep Learning	Infinite	Infinite
IMC PI	11.1	59
Fractional order PI	3.1	12
Ziegler Nichols tuned PI	5.05	28.8
Cohen coon tuned PI	5.77	29.6
Tyreus luyben tuned PI	9.52	66.9

5. CONCLUSION AND FUTURE SCOPE

A HE system has been simulated and the generated input and output data have been used to train a DLC using Backpropagation. The control horizon falls between one and the prediction horizon, and here control horizon is taken as 7 and the cost horizon as 35. During the training process, the best validation performance 5.15*10-9. Dataset has been divided into three sets a training set, validation set and test set. The advantage of using a validation data set is to avoid overfitting of the model. Train, validation and test data sets show that they are the best-fitted curve. Training, testing and validation data are also shown in the result. They show that neural networks learn very good and perform best during testing and validation. The performance of the heat exchanger system is measured by time response characteristics and frequency response characteristics through closed-loop simulation in MATLAB. The step response of magnitude 85 of the heat exchanger system with a DLC has been evaluated and also tested for set point changes. After time response and frequency response based analysis were carried out it is observed that the deep learning controller provides a satisfactory performance and outperforms as a comparison to the PID controller. Further work in this direction for the proposed work will have a great advantage in the real systems. Optimization techniques such as particle swarm optimization, genetic algorithm and intelligent techniques such as neural fuzzy interference system can be used in fine-tuning of deep learning controllers for better performance.

REFERENCES

[1] V. M. Chadeev and N. I. Aristova, "Control of industrial automation," Proc. 2017 10th Int. Conf. Manag. Large-Scale Syst. Dev. MLSD 2017, 2017, doi: 10.1109/MLSD.2017.8109604.
[2] E. L. Itskovich, "Methodology for attaining the rational level of plant automation," Autom. Remote Control, vol. 72, no. 5, pp. 1080–1088, 2011, doi: 10.1134/S000511791105016X.
[3] N. A. S. Alwan and Z. M. Hussain, "Deep learning control for digital feedback systems: Improved performance with robustness against parameter change," Electron., vol. 10, no. 11, 2021, doi: 10.3390/electronics10111245.
[4] F. L. Lewis and S. S. Ge, "Neural Networks in Feedback Control Systems," Mech. Eng. Handb., pp. 791–825, 2006, doi: 10.1002/0471777455.ch19.
[5] B. Prasad, R. Kumar, and M. Singh, "Performance Analysis of Model Predictive Control for Cascaded Tank Level Control System," 2021 IEEE 2nd Int. Conf. Electr. Power Energy Syst. ICEPES 2021, pp. 10–15, 2021, doi: 10.1109/ICEPES52894.2021.9699765.
[6] P. J. Werbos, "Neural networks for control and system identification," Proc. IEEE Conf. Decis. Control, vol. 1, pp. 260–265, 1989, doi: 10.1109/cdc.1989.70114.
[7] J. Schmidhuber, "Deep Learning in neural networks: An overview," Neural Networks, vol. 61, pp. 85–117, 2015, doi: 10.1016/j.neunet.2014.09.003.
[8] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.
[9] K. Cheon, J. Kim, M. Hamadache, and D. Lee, "On Replacing PID Controller with Deep Learning Controller for DC Motor System," J. Autom. Control Eng., vol. 3, no. 6, pp. 452–456, 2015, doi: 10.12720/joace.3.6.452-456.
[10] A. M. Zaki, A. M. El-Nagar, M. El-Bardini, and F. A. S. Soliman, "Deep learning controller for nonlinear system based on Lyapunov stability criterion," Neural Comput. Appl., vol. 33, no. 5, pp. 1515–1531, 2021, doi: 10.1007/s00521-020-05077-1.
[11] K. Erenturk, "Design of 'Deep Learning Controller,'" Int. J. Eng. Appl. Sci., vol. 5, no. 10, pp. 122–124, 2018, doi: 10.31873/ijeas.5.10.31.
[12] V. Prasad and B. W. Bequette, "Nonlinear system identification and model reduction using artificial neural networks," Comput. Chem. Eng., vol. 27, no. 12, pp. 1741–1754, 2003, doi: 10.1016/S0098-1354(03)00137-6.
[13] L. Ljung, C. Andersson, K. Tiels, and T. B. Schön, "Deep learning and system identification," IFAC-PapersOnLine, vol. 53, no. 2, pp. 1175–1181, 2020, doi: 10.1016/j.ifacol.2020.12.1329.
[14] B. M. Åkesson and H. T. Toivonen, "A neural network model predictive controller," J. Process Control, vol. 16, no. 9, pp. 937–946, 2006, doi: 10.1016/j.jprocont.2006.06.001.
[15] N. Sharma and K. Singh, "Model predictive control and neural network predictive control of TAME reactive distillation column," Chem. Eng. Process. Process Intensif., vol. 59, pp. 9–21, 2012, doi: 10.1016/j.cep.2012.05.003.
[16] A. Bamimore, N. B. Sobowale, A. S. Osunleke, and O. Taiwo, "Offset-free neural network-based nonlinear model predictive controller design using parameter adaptation," Neural Comput. Appl., vol. 33, no. 16, pp. 10235–10257, 2021, doi: 10.1007/s00521-021-05788-z.
[17] D. Hunter, H. Yu, M. S. Pukish, J. Kolbusz, and B. M. Wilamowski, "Selection of proper neural network sizes and architectures-A comparative study," IEEE Trans. Ind. Informatics, vol. 8, no. 2, pp. 228–240, 2012, doi: 10.1109/TII.2012.2187914.
[18] R. Kumar, B. Singh, D. T. Shahani, A. Chandra, and K. Al-Haddad, "Recognition of Power-Quality Disturbances Using S-Transform-Based ANN Classifier and Rule-Based Decision Tree," IEEE Trans. Ind. Appl., vol. 51, no. 2, pp. 1249–1258, 2015, doi: 10.1109/TIA.2014.2356639.
[19] S. Padhee, "Controller design for temperature control of heat exchanger system: Simulation studies," WSEAS Trans. Syst. Control, vol. 9, no. 1, pp. 485–491, 2014.
[20] A. A. A. Emhemed, A. Alsseid, and D. Hanafi, "Modelling and Controller Design for Temperature Control of Power Plant Heat Exchanger," vol. 5, no. 3, pp. 49–53, 2017, doi: 10.13189/ujca.2017.050302.
[21] D. M. Vega and H. G. Acevedo, "Advanced Control System Design for a Plate Heat Exchanger," 2020 9th Int. Congr. Mechatronics Eng. Autom. CIIMA 2020 - Conf. Proc., 2020, doi: 10.1109/CIIMA50553.2020.9290306.
[22] A. Vasičkaninová, M. Bakošová, L. Čirka, and M. Kalúz, "Robust controller design for a heat exchanger," Chem. Eng. Trans., vol. 52, no. 2002, pp. 247–252, 2016, doi: 10.3303/CET1652042.
[23] S. E. L. Coughanowr, Donald R, "Process systems analysis and control third edition," McGraw-Hill, p. 5, 2009, [Online]. Available: http://www.slideshare.net/accelerate786/process-systems-analysis-and-control-third-edition

[24] G. Stephanopoulos, "Chemical Process Control An Introduction to Theory and Practice Problem Sotving in Chemical Engineering with Numerical Methods Fluid Mechanics for Chemical Engineering," Elements, p. 696, 1983.
[25] A. Sahoo, T. K. Radhakrishnan, and C. S. Rao, "Modeling and control of a real time shell and tube heat exchanger," Resour. Technol., vol. 3, no. 1, pp. 124–132, 2017, doi: 10.1016/j.reffit.2016.12.001.
[26] K. R. Sundaresan and P. R. Krishnaswamy, "Estimation of time delay time constant parameters in time, frequency, and laplace domains," Can. J. Chem. Eng., vol. 56, no. 2, pp. 257–262, 1978, doi: 10.1002/cjce.5450560215.
[27] S. Skogestad, "Simple analytic rules for model reduction and PID controller tuning," Model. Identif. Control, vol. 25, no. 2, pp. 85–120, 2004, doi: 10.4173/mic.2004.2.2.
[28] "Design Neural Network Predictive Controller." https://in.mathworks.com/help/deeplearning/ug/design-neural-network-predictive-controller-in-simulink.html
[29] P. Zhang, "Advanced Industrial Control Technology British Library Cataloguing in Publication Data," 2010.
[30] C. Simon Haykin (McMaster University, Hamilton, Ontario, "Neural Networks - A Comprehensive Foundation - Simon Haykin.pdf." p. 823, 2005.