Prediction Interval-based ANFIS Controller for Nonlinear Processes

Mohammad Anwar Hosen, Abbas Khosravi, Saeid Nahavandi and Lachlan Sinnott

***Abstract*—Prediction interval (PI) has been appeared as a promising tool to quantify the uncertainties and disturbances associated with point forecasts. Despite of its numerous applications in prediction problems, the use of PIs in control application is still limited. In this paper, a PI-based ANFIS controller is proposed and designed for nonlinear systems. In the proposed algorithm, a PI-based neural network model (PI-NN) is developed to construct the PIs, and this model is used as an online estimator of PIs for the controller. The PIs along with other traditional inputs are used to train the inverse ANFIS model. The developed PI-based ANFIS model is then used as a nonlinear PI-based controller (PIC). The performance of the proposed PIC is examined for a nonlinear numerical plant. Simulation results revealed that the proposed PIC performance is superior over the traditional ANFIS-based controller.**

***Keywords*—Prediction interval, Nonlinear controllers, ANFIS, Neural networks, PI-NN, PIC.**

I. Introduction

PREDICTION INTERVAL (PI)-based modelling technique has been vigorously used in different fields of science to quantify the process's uncertainties and disturbances, these include power generation industries, transportation, food industries, and chemical and manufacturing processes [1]–[5].

In contrast to traditional point-based forecasts model, PI-based model predicts an interval consisting of upper bound and lower bound as seen in Fig. 1. The figure shows the PI (PI-based model output) along with respective target and traditional model output. In this technique, it is assumed that the target value should lies between upper and lower bounds. Usually, a PI-based model is develop with a predefined confidence level (1-α). This confidence level (CL) denotes the accuracy of the model. For instance, if a model is developed with a CL of 90%, the constructed PIs from that model should covered at least 90% of the target values. In contrast to PI-based model, traditional point forecast-based model does not bear any information related to their accuracy.

As described in [4], PIs is more appropriate than traditional point-based forecasts to quantify the uncertainties and disturbances. Moreover, PIs bear more information than point-based forecasts. However, the uses of PI-based model and/or informative PIs in control applications are still limited [6]. It is a challenging task to integrate the PIs in control systems. More precisely, it is unknown how to cater the information from the PIs and feed to the controller to improve the control signal. As seen in Fig. 1, three extra variables can be found from the PI-NN model. These include, upper bound ($\overline{y}$), lower bound ($\underline{y}$) and width of the interval ($\overline{y}-\underline{y}$). It is expected that these variables can be utilized in expert system-based controllers, such as fuzzy logic controllers, (FLC). However, it is difficult to develop the fuzzy rules based on the PIs. Another technique, namely adaptive neuro-fuzzy interface system (ANFIS) may be appropriate solution to integrate the PIs in control application.

ANFIS is a hybrid system that combined artificial NN and fuzzy interface system. In this method, NN produces a correlation between system's inputs and output with respect to the individual fuzzy rules. This method is very helpful where the relation of the system's inputs and outputs are unknown. More precisely, this method can be applied to a system without any prior knowledge. Therefore, ANFIS has appeared as a powerful tool for nonlinear system identification [7], [8]. This technique is also widely used as an effective controller to control nonlinear system [9], [10].

In the present work, a PI-based ANFIS controller is proposed and designed for a nonlinear system to get the better tracking performance of the controller. In the proposed algorithm, a PI-based model, PI-NN is developed to get the PIs for every sample instance. The PI-NN model is then used as an online estimator of PIs in control system. The PIs along with other tradition inputs are used to train and optimise the ANFIS. Finally, the proposed PI-based control system, PIC is examined for a nonlinear numerical plant.

The rest of the paper is structured as follows. Section II describes the proposed PI-based controller. This section also includes the development procedure of the PI-based model, PI-NN to construct the PIs. The case study and data generation technique is includes in Section III. Section IV demonstrates the prediction performance of the ANFIS model, also describes the tracking performance of the proposed PI-based controller.

II. Proposed PI-based Controller, PIC

A. PI-based NN model, PI-NN

As described in Section I, a PI-based model is required to design PIC system. In recent years, NNs have been extensively used to develop PI-based model [4]. As described in [11] and [12], lower upper bound estimation (LUBE) technique that proposed by Khosravi $et.al$ [13] is the best method so far to construct the quality and informative PIs. Therefore, LUBE method is used to develop the PI-NN model. Unlike traditional error-based cost functions, a PI-based cost function, namely coverage width criterion (CWC)

Mohammad Anwar Hosen, Abbas Khosravi and Saeid Nahavandi are with the Institute for Intelligent Systems Research ans Innovation (IISRI), Deakin University, Australia (email: {anwar.hosen, abbas.khosravi, saeid.nahavandi}@deakin.edu.au), and Lachlan Sinnott is with HPS Engineering, VIC, Australia (email: lachlans@hpseng.com.au).

This research was fully supported by the Institute for Intelligent Systems Research ans Innovation (IISRI) at Deakin University, Australia and HPS Engineering, VIC, Australia.

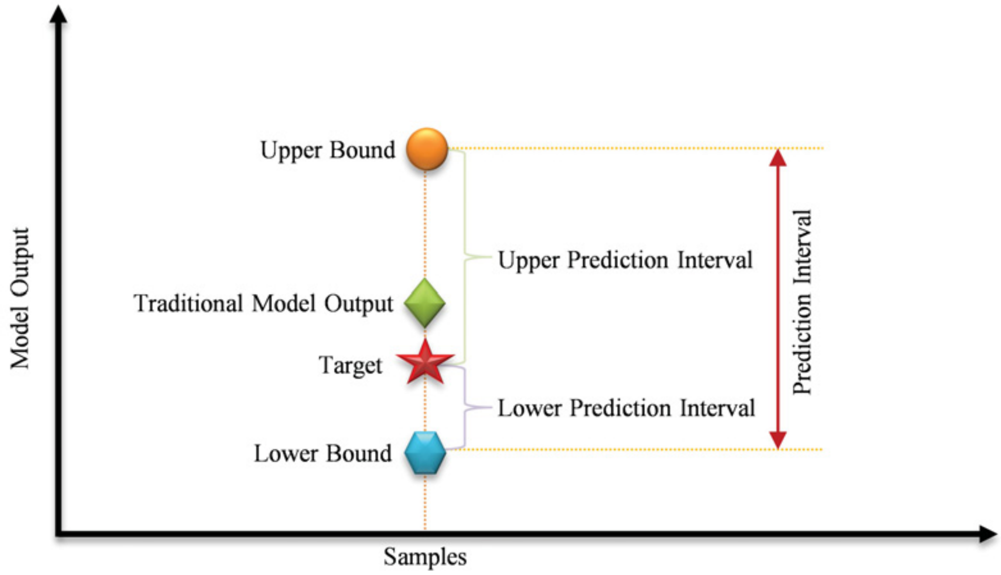

Fig. 1. Prediction Interval-based model

is utilised in the LUBE method to optimise the NN structure. The PI-based cost function, CWC can be defined as:

$$CWC = PINAW + (PICP)e^{-\eta(PICP-\varphi)}. \quad (1)$$

where $PINAW$ is the normalise average width of the PIs, $PICP$ is the PIs coverage probability, φ is the predefined CL, and η is the hyper-parameter that magnify the penalty if $PICP <$ nominal CL. Here $PICP$ and $PINAW$ are the two quality indexes of PIs. $PICP$ indicates the percentage of the target values that covered by constructed PIs and $PINAW$ is correspond to the width of the PIs. $PICP$ and $PINAW$ can be defined as [13]:

$$PICP = \frac{1}{n}\sum_{j=1}^{n} c_j \quad (2)$$

where

$$c_j = \begin{cases} 1, tr_j \in [\underline{y}_j, \overline{y}_j] \\ 0, tr_j \notin [\underline{y}_j, \overline{y}_j], \end{cases} \quad \text{and}$$

$$PINAW = \frac{1}{R}\left(\frac{1}{n}\sum_{j=1}^{n}(\overline{y}_j - \underline{y}_j)\right). \quad (3)$$

where $R = (max(t_r) - min(t_r))$, and j is the sample index of PIs.

In this method, NN predicts two outputs, one output is corresponds to the upper bound, and the other one is lower bound. In recent years, Hosen *et. al* [2] extended the LUBE method through forecasts aggregation. Actually the authors ensemble several PI-NN models to get the better quality of PIs. This work followed the same procedure to develop the PI-NN model for PIC system.

B. PI-based ANFIS Controller

ANFIS uses NN to search for fuzzy decision rules. Takagi-Sugeno type if-then fuzzy rules are used in ANFIS system. Fig. 2 shows the basic structure of ANFIS model for first order Takagi-Sugeno interface system [7]. This model consists of two inputs, x and y, and one output, f. The two rules for this ANFIS are as follows:

- **Rule 1**: if x is A_1 and y is B_1; then $f_1 = p_1x + q_1y + r_1$
- **Rule 2**: if x is A_2 and y is B_2; then $f_2 = p_2x + q_2y + r_2$

where A and B are the fuzzy sets for inputs x and y, respectively, and p, q and r are the parameters of the output function. These output parameters are optimize during the training process. As seen in Fig. 2, ANFIS structure consists of five layers, first and second layers correspond to the fuzzification and rule layers, respectively. Third layer is the normalisation layer, where fourth and fifth layers are the defuzzification and output layers, respectively. The detailed description of the ANFIS structure can be found in [7].

In expert system, fuzzy interface-based controller (FLC), error and rate of error (ROE) are usually used as inputs to predict the control signal/manipulated variable [14]. However, likewise, NN-based controller, the effective plant outputs (controlled variable) and inputs (delayed manipulated variable) can be used as ANFIS inputs. The inputs-output structure for this kind of ANFIS controller is $[y(k), y(k-1), u(k-1); u(k)]$. where y, u, and k are the model output, input and sampling index, respectively. From the PI-NN model, an extra input, PI can be found to train the ANFIS controller. The inputs-output structure for the proposed PI-based ANFIS controller is as follows: $[y_{set}, y(k), y(k-1), PI(k), u(k-1); u(k)]$ as seen in Fig. 3. The idea is that ANFIS system extracts the information from the PIs through

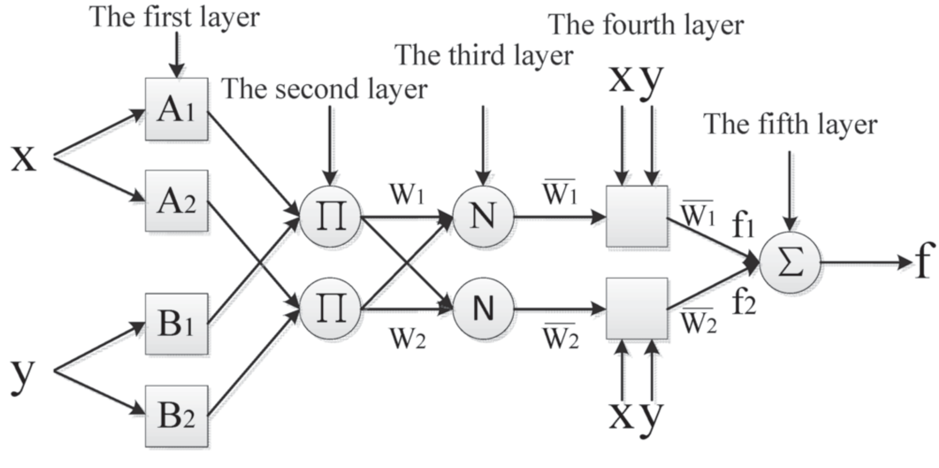

Fig. 2. Basic ANFIS Structure

fuzzy rules and NN mapping, and improves the controller signal.

III. Case Study

The proposed PI-based ANFIS controller is examined for a nonlinear numerical plant model. The model can be expressed mathematically as follows:

$$y(k+1) = \frac{y(k)}{1+y(k)^2} + u(k)^3. \tag{4}$$

where y and u are the controlled and manipulated variables, respectively. This model is nonlinear in nature and used as a benchmark model for control study [15], [16]. In this work, the model in (4) is simulated to generate the training data for PI-NN and ANFIS controller. The following setpoint is used to generate the training data:

$$y_{set} = 2.5\sin\left(\frac{10\pi k}{N}\right) + 2.5\sin\left(\frac{4\pi k}{N}\right). \tag{5}$$

where N=total number of samples. White noise (variance=1) is added in the plant output as unknown disturbances. Closed-loop system is employed to simulate the model and collected total 5000 data in the form of $[y, u]$ (where sampling time= 1). A PID controller is developed and tuned based on Matlab PID Tuner tool to generate the closed-loop data.

IV. Results and Discussion

First of all, a PI-NN model is developed to construct the PIs to train ANFIS. The inputs-output structure for PI-NN is $[y(t), y(t-1), u(t), u(t-1); \overline{y}(t-1), \underline{y}(t-1)]$. The generated data in Section III are rearranged as PI-NN's inputs-output structure. After normalising the data, randomly split into training (60% of total data), validation (20% of total data) and testing (20% of total data) data. Simulated annealing optimization algorithm is used to optimise the NN structure by minimizing a PI-based cost function, CWC defined in (1). The parameters used in PI-NN development process are taken from [2]. Total 10 PI-NNs are developed by varying the hidden neuron size to diversify the structure of NNs. Finally, best five PI-NNs are selected for ensemble process as described in [5]. The developed ensemble PI-NN model is then used to to construct the PIs for ANFIS. This model also used in PIC system as an online estimator of PIs.

The next step is to generate the data for ANFIS. In this connection, the developed PI-NN model is connected in parallel with the mathematical model that defined in (4). The same setpoint and PID controller described in Section III are used to generate the training data for ANFIS. This time, simulate the models for 5,000sec and collected total 20,000 data for each variable where sampling time = 0.25sec . The variables include $y_{set}, y, u, \overline{y}, \underline{y}$. y_{set} is the setpont for PID controller. Finally, the data are rearranged as the structure of ANFIS system, and 90% and 10% of total data are selected randomly for training and testing, respectively. The inputs-output data structure for ANFIS controller is as follows:

$$u(t) = ANFIS\left\{y_{set}(t), y(t-1), y(t), \overline{y}(t-1) - \underline{y}(t-1), \right.$$
$$\left. \overline{y}(t) - \underline{y}(t), u(t-1)\right\}. \tag{6}$$

where $\overline{y}(t) - \underline{y}(t)$ is the width of the prediction interval. As seen in (6), six inputs are used against one output for ANFIS.

ANFIS Toolbox in Matlab R2015b is used to train the ANFIS. Grid partition is used to classify the input data and generate the fuzzy rules. Hybrid optimization method is utilised to optimize the ANFIS parameters through minimising the error-based cost function, namely root mean square errors (RMSE). Hybrid optimization algorithm is a combination of least-square estimation method and backpropagation method [17]. Trail and error method is employed to select the number and type of membership functions (MFs) for each variable. Simulation results demonstrated that ANFIS with two MFs (gaussian type) for each input variable produces better results than others. A linear function is used as output MF.

A traditional ANFIS controller is also developed to compare the results with the proposed PI-based ANFIS controller. The inputs-output structure for traditional ANFIS

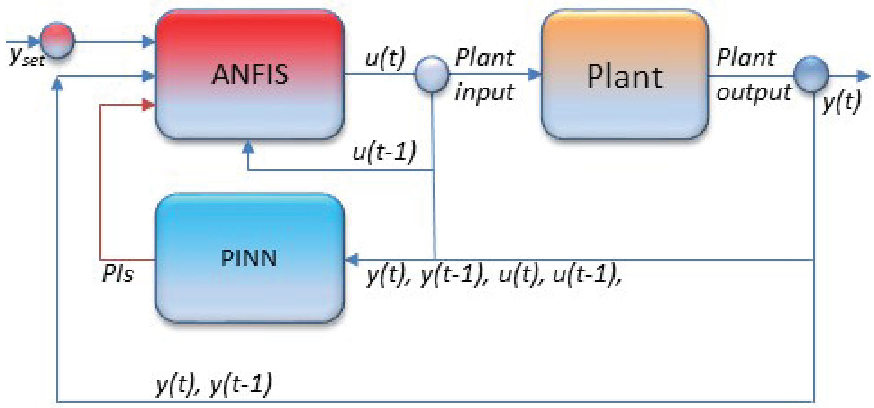

Fig. 3. PI-based ANFIS control system

is as follows: $[y_{set}(t), y(t-1), y(t), u(t-1); u(t)]$. The same procedure as PI-based ANFIS is applied to develop traditional ANFIS. Fig. 4 depicts the traditional ANFIS and PI-based ANFIS performance for test data set in terms of errors. As seen in Fig. 4 the errors, $(y_{plant} - y_{predicted})$ are very low that indicate very good fitting of ANFIS. The RMSE for ANFIS and PI-ANFIS are 4.37×10^{-5} and 4.62×10^{-5}. This means that the fitting ability of the ANFIS and PI-ANFIS is excellent for this nonlinear plant.

After developing the PI-ANFIS, the developed PI-NN model is added in parallel to the PI-ANFIS as seen in Fig. 3. The whole PIC system is then simulated to track the different setpoint for nonlinear numerical plant.

A. Tracking performance of PI-ANFIS Controller

The proposed PI-ANFIS controller is examined for different setpoint tracking, such as constant, step changes and wavy setpoint, and Integral absolute error (IAE) is used as the performance criterion of the controller. The performance of the proposed PIC is compared with the traditional ANFIS controller.

Fig. 5 depicts the controllers' performance for constant setpoint. A constant setpoint = 4 is used to simulate the controllers for 5,000sec. As seen in Fig. 5, the tracking performance of the both controllers are quite good. However, the settling time for PI-ANFIS is less than the traditional ANFIS controller though there is little overshoot observed for PI-ANFIS. The performance of the PI-ANFIS is better than the ANFIS in terms of IAE as seen in Table I. The IAE values for PI-ANFIS and ANFIS are 3.076 and 18.25, respectively. This means that the tracking performance can be improved by 83% using PI-ANFIS than the traditional ANFIS.

The stability and robustness of the proposed controller are evaluated by using upward and backward step changes as well as wavy setpoints. Fig. 7 and 6 demonstrate the tracking performance of the both controllers for step changes and wavy setpoints, respectively. Upward step changes are made at time 1,000 and 4,000sec where downward step change is made at 2,500sec as seen in Fig. 7. Despite the negligible overshoot for the step changes setpoint tracking,

TABLE I. TRACKING PERFORMANCES OF PI-ANFIS AND TRADITIONAL ANFIS CONTROLLERS IN TERMS OF IAE

Type of setpoint	Controllers	IAE	Improvement (%)
Constant	PI-ANFIS	3.076	83
	ANFIS	18.25	
Step changes	PI-ANFIS	24.28	39
	ANFIS	40.11	
Wavy	PI-ANFIS	55.81	74
	ANFIS	216.40	

smooth setpoint tracking is observed for both step changes and wavy setpoints using PI-ANFIS. However, a considerable overshoot at 2,500sec (step changes setpoint tracking) and minor offsets (wavy setpoint tracking) are observed for traditional ANFIS tracking. The *IAE* values for PI-ANFIS tracking are 24.28 and 55.81 for step changes and wavy setpoints, respectively as seen in Table I. The *IAE* values for ANFIS are 40.11 and 216.40 for step changes and wavy setpoints, respectively that are higher than the PI-ANFIS tracking. According to Table I, 39% (for step changes setpoint) and 74% (for wavy setpoint) improvements can be achieved in terms of *IAE* by using the proposed PI-ANFIS controller over the traditional ANFIS.

V. CONCLUSIONS

Despite the vast applications of PIs in forecasting problems to quantify the uncertainties and disturbances, the use of PIs in control applications are very limited. It is well established that PIs bear more information than point forecasts. This means that the use of PIs in a control system will improve the tracking performance of the controllers.

This paper investigates the possible use of PIs in control applications. As ANFIS system automatically generates the correlation between system's inputs and output based on the process data using fuzzy rules and NN models, PI-based ANFIS controller is proposed and developed for tracking nonlinear processes. First of all, a PI-based model is developed using extended LUBE method to construct the PIs for ANFIS system. PIs along with other traditional inputs are used to train the PI-based ANFIS controller.

The performance of the proposed PI-ANFIS controller is examined for a nonlinear numerical plan model and

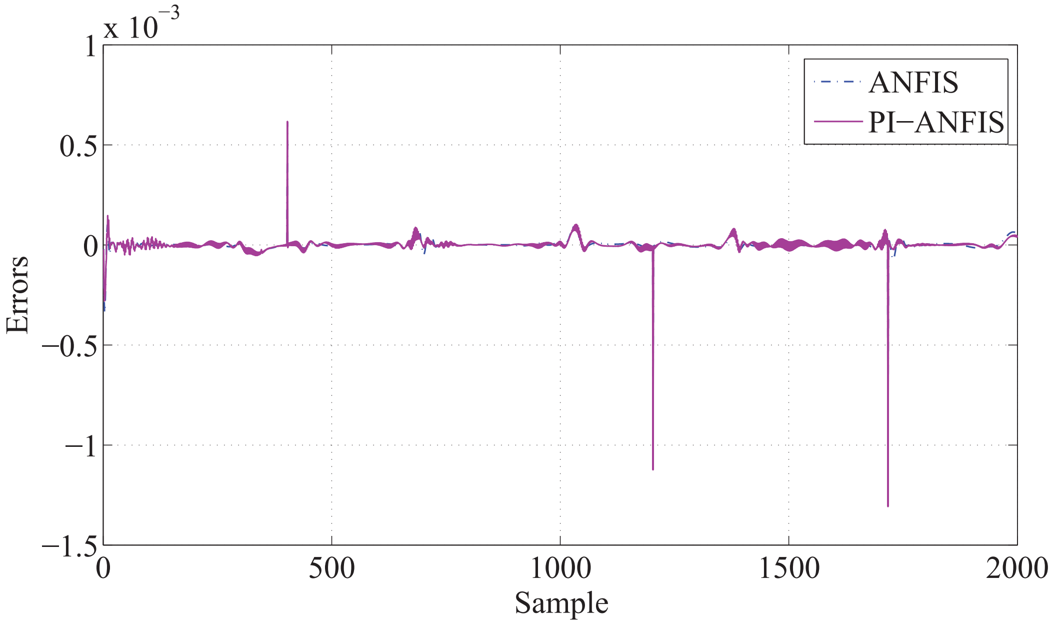

Fig. 4. Fitting performance of traditional ANFIS and PI-based ANFIS for test data set

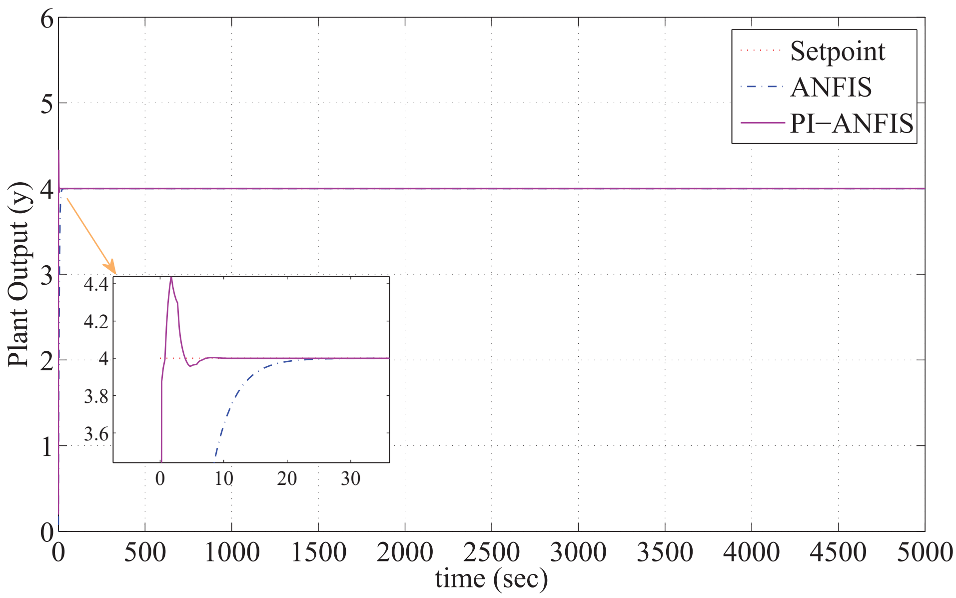

Fig. 5. Constant setpoint tracking using PI-ANFIS and traditional ANFIS controller

compared the results with the traditional ANFIS controller. Simulation results show that the proposed PIC improved the tracking performance by 83% for constant setpoint, 39% for step changes setpoint and 74% for wavy setpoint over the traditional ANFIS.

References

[1] H. Quan, D. Srinivasan, and A. Khosravi, "Incorporating wind power forecast uncertainties into stochastic unit commitment using neural network-based prediction intervals," *Neural Networks and Learning Systems, IEEE Transactions on*, no. 99, pp. 1–1, 2014.

[2] M. A. Hosen, A. Khosravi, S. Nahavandi, and D. Creighton, "Improving the quality of prediction intervals through optimal aggregation," *Industrial Electronics, IEEE Transactions on*, vol. 62, no. 7, pp. 4420–4429, 2015.

[3] P. L. Espinheira, S. L. Ferrari, and F. Cribari-Neto, "Bootstrap prediction intervals in beta regressions," *Comput. Stat.*, pp. 1–15, 2014.

[4] A. Khosravi, S. Nahavandi, D. Creighton, and A. F. Atiya, "Comprehensive review of neural network-based prediction intervals and new advances," *Neural Networks, IEEE Transactions on*, vol. 22, no. 9, pp. 1341–1356, 2011.

[5] M. A. Hosen, A. Khosravi, S. Nahavandi, and D. Creighton, "Aggregation of pi-based forecast to enhance prediction accuracy," in *Neural Networks (IJCNN), 2014 International Joint Conference on*. IEEE, 2014, pp. 778–784.

[6] M. A. Hosen, A. Khosravi, S. Nahavandi, D. Creighton, and S. M. Salaken, "Prediction interval-based neural network controller for non-

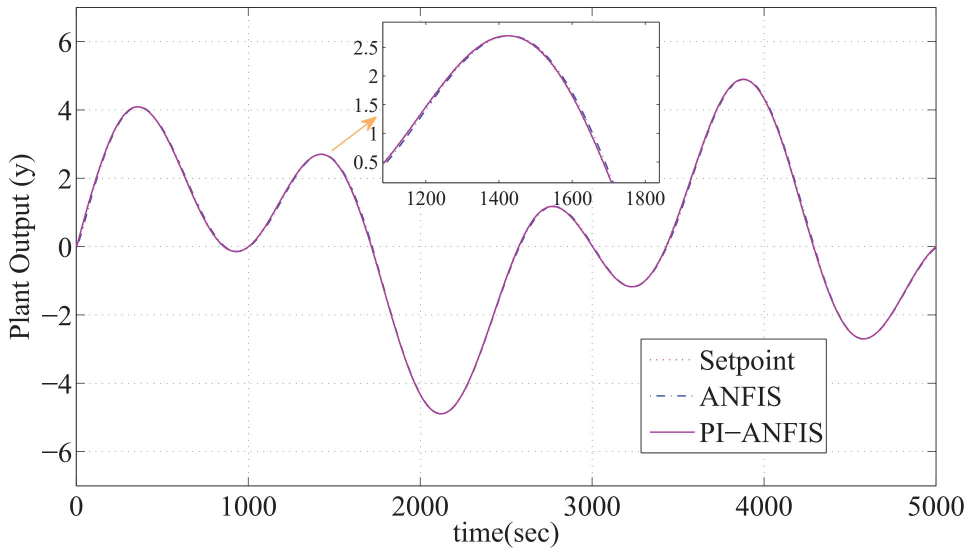

Fig. 6. Wavy setpoint tracking using PI-ANFIS and traditional ANFIS controller

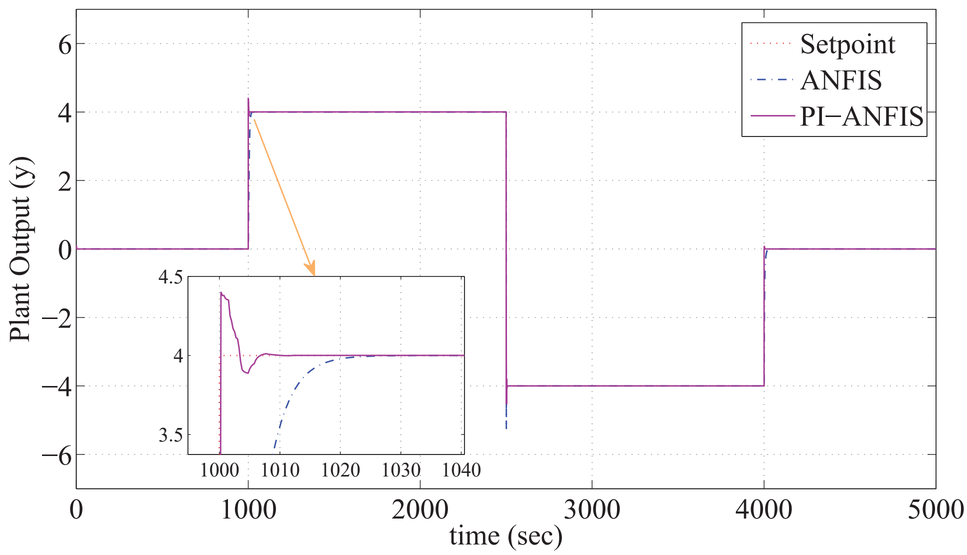

Fig. 7. Step changes setpoint tracking using PI-ANFIS and traditional ANFIS controller

linear processes," in *Neural Networks (IJCNN), 2015 International Joint Conference on*. IEEE, 2015, pp. 1–6.

[7] J.-S. R. Jang, "Anfis: Adaptive-network-based fuzzy inference system," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 23, no. 3, pp. 665–685, 1993.

[8] T. R. Borah, K. K. Sarma, and P. H. Talukdar, "Retina recognition system using adaptive neuro fuzzy inference system," in *Computer, Communication and Control (IC4), 2015 International Conference on*. IEEE, 2015, pp. 1–6.

[9] S.-C. Chen, D.-K. Le, and V.-S. Nguyen, "Adaptive network-based fuzzy inference system (anfis) controller for an active magnetic bearing system with unbalance mass," in *AETA 2013: Recent Advances in Electrical Engineering and Related Sciences*. Springer, 2014, pp. 433–443.

[10] A. Kharola, "Design of a hybrid adaptive neuro fuzzy inference system (anfis) controller for position and angle control of inverted pendulum (ip) systems," *International Journal of Fuzzy System Applications (IJFSA)*, vol. 5, no. 1, pp. 27–42, 2016.

[11] M. A. Hosen, A. Khosravi, S. Nahavandi, and D. Creighton, "Prediction interval-based neural network modelling of polystyrene polymerization reactor : a new perspective of data-based modelling," *Chemical Engineering Research and Design*, vol. 92, no. 11, pp. 2041–2051, 2014.

[12] M. A. Hosen, A. Khosravi, D. Creighton, and S. Nahavandi, "Prediction interval-based modelling of polymerization reactor: A new modelling strategy for chemical reactors," *Journal of the Taiwan Institute of Chemical Engineers*, vol. 45, no. 5, pp. 2246–2257, 2014.

[13] A. Khosravi, S. Nahavandi, D. Creighton, and A. Atiya, "Lower upper bound estimation method for construction of neural network-based prediction intervals," *Neural Networks, IEEE Transactions on*, vol. 22, no. 3, pp. 337–346, 2011.

[14] M. A. Hosen, M. A. Hussain, F. S. Mjalli, A. Khosravi, D. Creighton, and S. Nahavandi, "Performance analysis of three advanced controllers for polymerization batch reactor: An experimental investigation," *Chemical Engineering Research and Design*, vol. 92, no. 5, pp. 903–916, 2014.

[15] R. Salman, "Neural networks of adaptive inverse control systems," *Applied Mathematics and Computation*, vol. 163, no. 2, pp. 931–939, 2005.

[16] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *Neural Networks, IEEE Transactions on*, vol. 1, no. 1, pp. 4–27, 1990.

[17] Z. Jiang, H. Zheng, N. Mantri, Z. Qi, X. Zhang, Z. Hou, J. Chang, H. Lu, and Z. Liang, "Prediction of relationship between surface area, temperature, storage time and ascorbic acid retention of fresh-cut pineapple using adaptive neuro-fuzzy inference system (anfis)," *Postharvest Biology and Technology*, vol. 113, pp. 1 – 7, 2016.