World Academy of Science, Engineering and Technology
International Journal of Mechanical and Mechatronics Engineering
Vol:4, No:8, 2010

A Black-Box Approach in Modeling Valve Stiction

H. Zabiri, N. Mazuki

***Abstract*—**Several valve stiction models have been proposed in the literature to help understand and study the behavior of sticky valves. In this paper, an alternative black-box modeling approach based on Neural Network (NN) is presented. It is shown that with proper network type and optimum model structures, the performance of the developed NN stiction model is comparable to other established method. The resulting NN model is also tested for its robustness against the uncertainty in the stiction parameter values. Predictive mode operation also shows excellent performance of the proposed model for multi-steps ahead prediction.

***Keywords*—**Control valve stiction, neural network, modeling.

I. Introduction

OSCILLATIONS in process variables are widely encountered in process plants [1]. The presence of oscillations in a control loop enhances the variability of the process variables hence creating inferior quality products, higher rejection rates, increased energy consumption and reduced average throughput. Interactions among process units further facilitate the propagation of oscillations across the plant.

There are many causes that may contribute to the oscillatory behavior observed in control loops. These include poorly tuned controllers, presence of oscillatory disturbances and nonlinearities [2]. A survey reported in [1] found that 30% of the loops are oscillatory due to control valve problems. Control valves constitute an important element in chemical process control systems. Through a control valve, control actions are implemented on the process. They manipulate energy flows, mass flows or forces as a response to low energy input signals, for example, electrical voltages or currents, pneumatic and hydraulic pressures or flows [3].

Due to their continuous motions, control valves tend to undergo wear and aging. In general, they contain static and dynamic nonlinearities including saturation, backlash, stiction, deadband and hysteresis [4]. Among the many types of nonlinearities in control valves, stiction is the most commonly encountered in the process industry [4]. In general, stiction is a phenomena that describes the valve's stem (or shaft) sticking when small changes attempted [4]. Stiction causes fluctuation of process variables, which lowers productivity. The variability of process variables makes it difficult to keep operating conditions close to their constraints, and hence causes excessive or unnecessary energy consumption. It is therefore desirable to understand and study the dynamics behavior of stiction so that necessary actions can be implemented to eliminate or hinders its deleterious effect before it propagates.

H. Zabiri is with Universiti Teknologi PETRONAS, Bandar Sri Iskandar, Tronoh 31750 Perak, Malaysia. (corresponding author phone: 605-368-7625; fax::605-366-6176; e-mail: haslindazabiri@petronas.com.my).

N. Mazuki is with with Universiti Teknologi PETRONAS, Bandar Sri Iskandar, Tronoh 31750 Perak, Malaysia (e-mail: sukma09@gmail.com).

Several valve stiction models have been proposed in the literature. Muller [5] described a detailed physical model that formulates the stiction phenomenon as precisely as possible. However this type of model is not only impractical, it is also time-consuming since there are a number of unknown physical parameters that must be solved. On the other hand, [4] proposed a data-driven model that describes the relationship between a controller output and a valve position. An extended version of [4] model that includes the flexibility of processing deterministic and stochastic signals has been proposed in [6]. However, both these empirical approaches involved with a rather complex logic making them difficult to implement.

This paper presents an alternative approach via a pure black-box modeling strategy based on artificial Neural Network. An artificial neural network (ANN) or commonly just neural network (NN) is an interconnected group of artificial neurons that uses a mathematical model or computational model for information processing based on a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network. There are various types of NN available, however, in this paper we will focus only on Recurrent Nonlinear AutoRegressive with eXogenous Inputs (NARX) Neural Networks. It is noted in [7] that NARX NN is capable of modeling heat exchangers, waste water treatment plants, catalytic reforming systems and various artificial nonlinear systems.

The outline of this paper is as follows: Section II describes stiction in general. In Section III, six Neural-Network algorithms considered in this paper are presented while Section IV illustrates the proposed methods in numerical simulations and benchmarked against the proven and validated model developed by Choudhury et. al [3] as well as the robustness study. Finally, the conclusions are drawn.

II. Control Valve Stiction

Fig. 1 shows the general structure of a pneumatic control valve. Stiction happens when the smooth movement of the valve stem is hindered by excessive static friction at the packing area. The sudden slip of the stem after the controller output sufficiently overcomes the static friction caused undesirable effect to the control loop.

Fig. 2 illustrates the input-output behavior for control valve with stiction. The dashed line represents the ideal control valve without any friction.

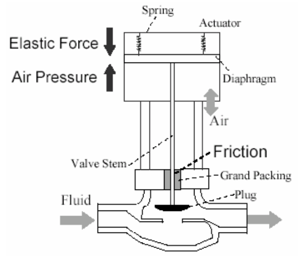


Fig. 1 Structure of pneumatic control valve adapted from [6].

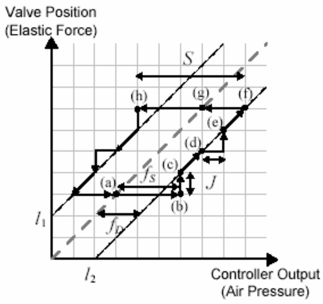


Fig. 2 Typical input-output behavior of a sticky valve adapted from [6].

Stiction consists of primarily of deadband, stickband, slip jump and the moving phase [8]. For control valve under stiction resting at point (a), the valve position remains unchanged even when the controller output increases due to the deadband caused by the static friction. Only when the controller output exceeds the maximum static frictional force, f_S, the valve starts to response (point(b)). A slip jump of magnitude J is incurred when the valve starts to move at point (b) when the frictional force f_S converts to kinetic force f_D. From (c) to (d), the valve position varies linearly. The same scenario happens when the valve stops at point (d), and when the controller output changes direction. Parameter S represents the deadband plus stickband regions.

There are four types of stiction, namely, deadband ($J=0$), stiction undershoot ($S>J$), stiction no offset ($S=J$) and stiction overshoot ($S<J$).

III. Neural-Network

An artificial neural network (ANN) or commonly just neural network (NN) is an interconnected group of artificial neurons that uses a mathematical model or computational model for information processing based on a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network. In this paper, six types of NN for modeling the control valve stiction are investigated.

A. Feedforward-Backpropagation Neural Network

Feedforward backpropagation neural networks (FF networks) are the most popular and most widely used models in many practical applications [9]. They are known by many different names, such as "multi-layer perceptrons." The following diagram illustrates a FF networks network with three layers:

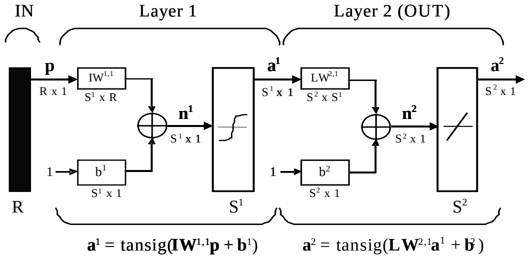


Fig. 3 Graphical representation of a BP network architecture.

Backpropagation (BP) network was created by generalizing the Widrow-Hoff learning rule to multiple-layer networks and nonlinear but differentiable transfer functions [10]. BP network with biases, a sigmoid ('tansig' or 'logsig') transfer functions at the hidden layers, and a linear transfer function at the output layer is capable of approximating any functions. BP networks architecture is slightly more complex than a single layer network. In addition to a single (hidden) layer consisting nodes with sigmoid transfer function, another layer called the output layer is required. The output layer is usually kept linear to produce output values in the similar range as the target values. However, the sigmoid transfer functions (either 'logsig' or 'tansig') are often used if the outputs need to be constrained to the range of [0, 1] or [-1, 1]. The minimum architecture of BP networks is illustrated as layer diagram in Fig. 3. The (R x 1) inputs p are fed to Layer 1 (hidden layer) consisting of S^1 'tansig' nodes. The resulting outputs a^2 with 'linear' transfer function retain the same size (S^2 x 1) as the net inputs n^2 to Layer 2 (output layer). With this architecture, the BP networks are capable of approximating any linear and nonlinear functions given adequate number of hidden nodes.

B. Cascade-forward Backpropagation Network

Feedforward networks have one-way connections from input to output layers. They are most commonly used for prediction, pattern recognition, and nonlinear function fitting. Supported feedforward networks include feedforward backpropagation and cascade-forward backpropagation. In CF network, each subsequent layer has weights coming from the input as well as from all previous layers.

Like FF networks, CF networks uses BP algorithm for updating of weights but the main symptoms of the network is that each layer neurons related to all previous layer neurons. In [11], several NN topologies were evaluated and it was found that the cascade forward NN with BP training provides the best performance in terms of convergence time, optimum network structure and recognition performance. The training of multi-layer perceptron (MLP) networks normally involves BP training as it provides high degrees of robustness and generalization [12].

C. Recurrent Neural Network

In Feedforward NN, the neurons in one layer receive inputs from the previous layer. Neurons in one layer deliver its output to the next layer; the connections are completely unidirectional; whereas in Recurrent NN, some connections are present from a layer to the previous layers. The next value of output is regressed on previous values of input signal (see Fig.4).

NARX Network

The nonlinear autoregressive network with exogenous inputs (NARX) is a recurrent dynamic network, with feedback connections enclosing several layers of the network.

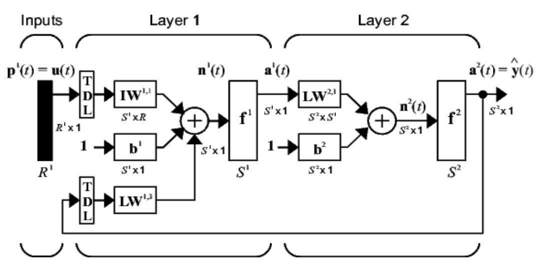
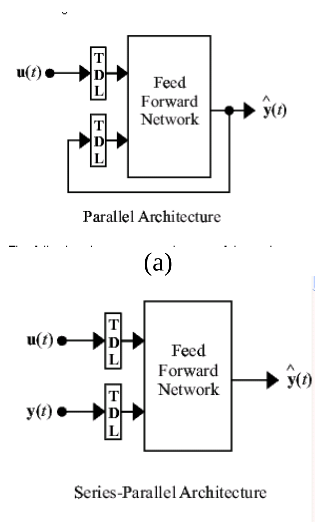


Fig. 4 Recurrent NARX NN structure.

The NARX model is based on the linear ARX model, which is commonly used in time-series modeling. The defining equation for the NARX model is shown in (1), where the next value of the dependent output signal *y(t)* is regressed on previous values of the output signal and previous values of an independent (exogenous) input signal.

$$y(t) = f(y(t-1), y(t-2), \ldots, y(t-n_y), u(t-1), u(t-2), \ldots, u(t-n_u)) \quad (1)$$



(a)

(b)

Fig. 5 NARX network architecture.

Standard NARX architecture is as shown in Fig. 5(a). It enables the output to be fed back to the input of the feedforward neural network. This is considered a feedforward BP network with feedback from output to input. In series parallel architecture (NARXSP), Fig. 5(b), the true output which is available during the training of the network is used instead of feeding back the estimated output. The advantage is that the input to the feedforward network is more accurate. Besides, the resulting network has a purely feedforward architecture, and static BP can be used for training.

Simple Recurrent Network (SRN)

Simple Recurrent Network (SRN) is also known as Elman network. In Elman network, the input vector is similarly propagated through a weight layer but also combined with the previous state activation through an additional recurrent weight layer. A two-layer Elman network is shown as in Fig.6.

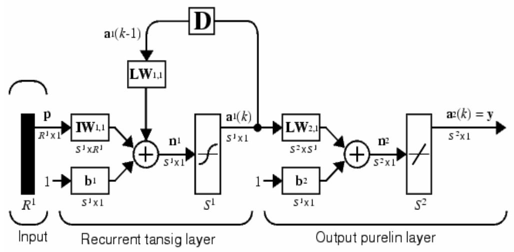


Fig. 6 Elman network structure.

The output of the network is determined by the state and a set of output weights, W,

$$y_k(t) = f(net_k(t))$$
$$net_k(t) = \sum_j^m y_j(t) w_{kj} + \theta_k \quad (2)$$

Elman network has activation feedback which embodies short-term memory. A state layer is updated through the external input of the network as well as the activation from the previous forward propagation. The feedback is modified by a set of weights as to enable automatic adaption through learning (e.g. BP). Elman network differs from conventional two-layer networks in that the first layer has a recurrent connection. The delay in this connection stores values from the previous time step, which can be used in the current time step. Because the network can store information for future reference, it is able to learn temporal patterns as well as spatial patterns. The Elman network can be trained to respond to, and to generate, both kinds of patterns.

Layer-recurrent Network (LRN)

An earlier simplified version of this network was introduced by Elman. In the LRN, there is a feedback loop, with a single delay, around each layer of the network except for the last layer. The original Elman network had only two layers. The original Elman network was trained using an approximation to the BP algorithm. Fig. 7 illustrates a two-layer LRN.

LRN generalizes the Elman network to have an arbitrary number of layers and to have arbitrary transfer functions in each layer. LRN is trained using exact versions of the gradient-based algorithms used in BP.

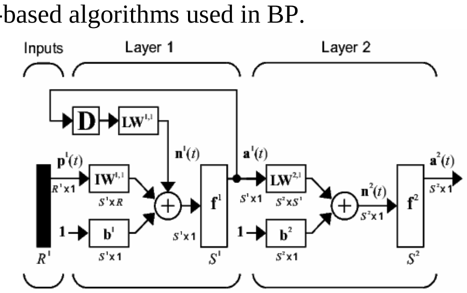


Fig. 7 Layer-recurrent neural network structure.

IV. Numerical Evaluations

This section is divided into three main parts. *Section A* presents results on the optimization of Neural Network model structures for valve stiction modeling. *Section B* illustrates the robustness study on the selected NARXSP-based stiction model whilst *Section C* compares the model and predictor modes of operation for the developed NN stiction model.

A. Optimization of Neural Network Model Structure

In this section, the six types of NN for modeling valve stiction are applied to simulated data generated using the validated and proven Choudhury's model of [3] from a simple sine wave function. The four cases of stiction are investigated, namely, deadband ($J=0$), stiction undershoot ($S>J$), stiction no offset ($S=J$) and stiction overshoot ($S<J$). The model structures for each of the NN types are initially analyzed and the optimized architecture is selected.

Figures 8 -13 show the results for stiction undershoot case for the six stiction models in comparison to the Choudhury's model. In stiction undershoot, the valve output can never reach the valve input, i.e., there will always be some offset. In this case, an imprecise matching between the Feedforward BP, Cascade, Elman, Layer Recurrent and NARX NN with the Choudhury's model output can be perceived. Feed forward BP NN failed to track the stiction behavior dexterously when it is unable to follow the shape of the data driven stiction model.

Both Cascade and Elman NN can merely follow the trend of the data driven model. There is an apparent deviation at the bottom peak when the signal changes direction. There is also a slight deviation at the top peak when the signal changes direction and both are unable to capture the sharp edges typical of stiction. The same observation can be seen when Layer-recurrent NN is applied. The NARX NN also cannot correctly match the Choudhury's model. Only NARXSP NN with real output fed to the network feature tracks the stiction behavior as efficient and as accurate as the Choudhury's model.

Statistical analysis verified the visual inspection. Root Mean Square Error (RMSE) of 0.3416, 0.3442, 0.2059, 0.206 and 0.27 are obtained for Feedforward BP, Cascade, Elman, Layer Recurrent, and NARX NN respectively, whilst NARXSP NN achieved RMSE value as low as 0.0907. However comparable Correct Directional Change (CDC) values are obtained for all networks as indicated by the satisfactory directional change tracking.

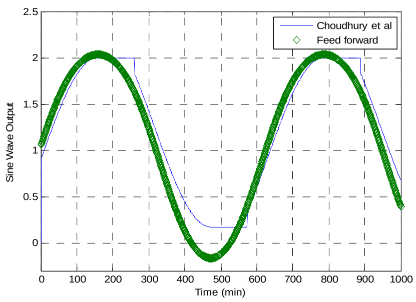


Fig. 8 Data driven vs. feedforward BP NN for stiction undershoot.

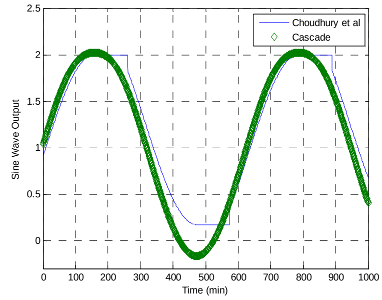


Fig. 9 Data driven vs. Cascade NN for stiction undershoot.

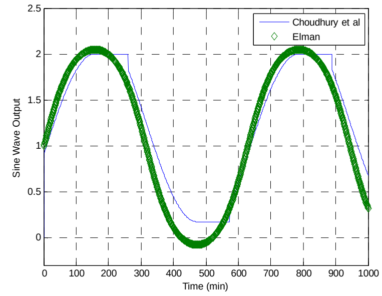


Fig. 10 Data driven vs. Elman NN for stiction undershoot.

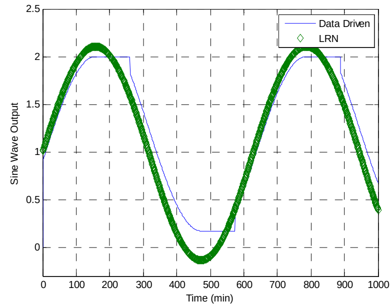


Fig. 11 Data driven vs. Layer-recurrent NN for stiction undershoot.

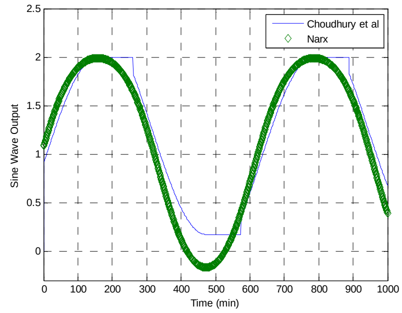


Fig. 12 Data driven vs. NARX NN for stiction undershoot.

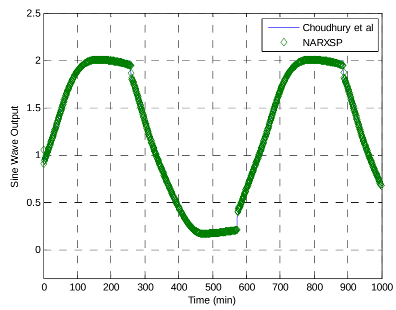


Fig. 13 Data driven vs. NARXSP NN for stiction undershoot.

The same observations are obtained for the deadband, stiction no offset and stiction overshoot cases. Only NARXSP can efficiently and accurately model the stiction behavior. The summary of the results are as tabulated in Table 1.

TABLE I
STATISTICAL ANALYSIS FOR NN ARCHITECTURE

Neural Network Model		*RMSE*	*CDC*
NARXSP	Deadband	0.0097	83%
	Stiction overshoot	0.0123	83%
	Stiction no offset	0.0234	69%
FF networks	Deadband	0.1221	83%
	Stiction overshoot	0.1312	83%
	Stiction no offset	0.1930	69%
CF networks	Deadband	0.1223	83%
	Stiction overshoot	0.1345	83%
	Stiction no offset	0.1929	69%
Elman	Deadband	0.0971	83%
	Stiction overshoot	0.1117	83%
	Stiction no offset	0.1975	69%
LRN	Deadband	0.1035	83%
	Stiction overshoot	0.1997	83%
	Stiction no offset	0.1573	69%
NARX	Deadband	0.1331	83%
	Stiction overshoot	0.1312	83%
	Stiction no offset	0.1962	69%

Results in this section clearly indicate that proper selection of the NN type, together with optimum configuration of the corresponding network architectures, obtained via heuristic approach, can efficiently and accurately model the stiction behavior. It is obvious that NARXSP-based stiction model gives the best performance for all four types of stiction. In the next section, the performance of this NARXSP-based stiction model is tested for robustness.

B. Robustness Study on NARXSP-based Stiction Models

As can be seen in the previous section, NARXSP NN is able to model stiction excellently for fixed *S* and *J* values. However, empirical modeling approaches such as NN are known to suffer from degradation in performance when operating conditions changes. It is imperative to investigate how robust such a model is under uncertainty in the values of *S* and *J*. In this section, the robustness of the developed NARXSP-based stiction models are tested against varying values of *S* and *J*, as well as against other types of stiction behavior.

Robustness analysis against varying values of S and J for same stiction types

For this analysis, a NARXSP-based stiction model is developed for the base case of $S = 3$ and $J = 1$ (i.e. stiction undershoot). Fig. 15 to 23 show the resulting behaviors when the base model is tested against varying values of *S* and *J*. It can be clearly observed that the base model is able to track the stiction behavior accurately and effectively for small *S* and *J* values. However, as the values of *S* and *J* increased, the performance of the base NARXSP model decreased.

Table 2 shows the summary of the performances of the other stiction NARXSP types. In general, the base model is able to tolerate mismatch if stiction is less than 6% of valve travel span for the same stiction type.

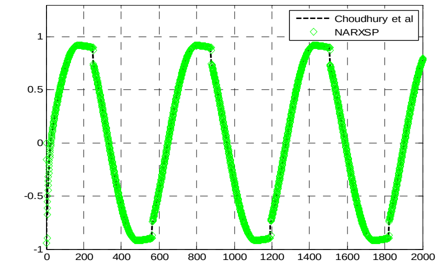


Fig. 15 NARXSP NN for test set *S=2* and *J=1*

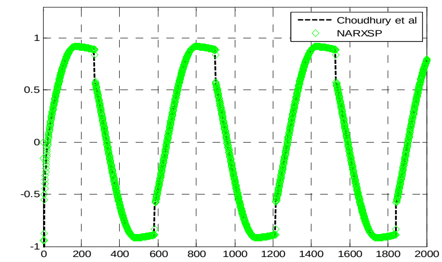


Fig. 16 NARXSP NN for test set *S=3* and *J=2*

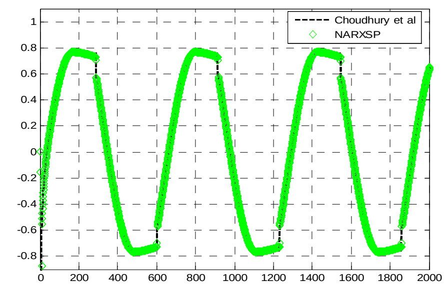


Fig. 17 NARXSP NN for test set *S=4* and *J=1*

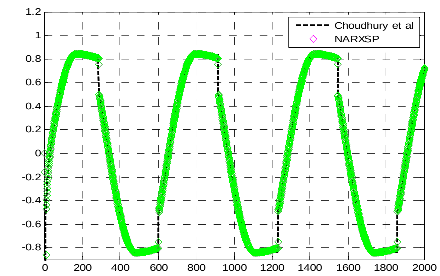


Fig. 18 NARXSP NN for test set *S=4* and *J=2*

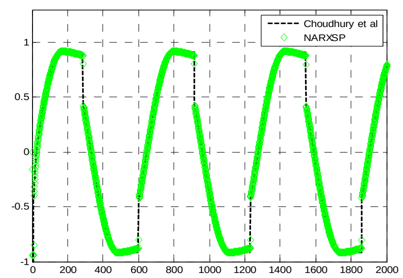


Fig. 19 NARXSP NN for test set *S=4* and *J=3*

TABLE II
PERFORMANCE SUMMARY FOR ANALYSIS IN ROBUSTNESS AGAINST VARYING S AND J VALUES OF SAME STICTION TYPES

Stiction Type	Base		Test		Performance
	S	J	S	J	
Pure deadband	1	0	5	0	Robust
			4	0	Robust
			3	0	Robust
			2	0	Robust
No offset	1	1	5	5	Not robust
			4	4	Not robust
			3	3	Not robust
			2	2	Robust
Stiction overshoot	1	3	1	5	Robust
			1	4	Robust
			1	3	Robust
			1	2	Robust
			2	5	Robust
			2	4	Robust
			2	3	Robust
			3	5	Robust
			3	4	Robust
			4	5	Robust
Stiction undershoot	3	1	5	4	Robust
			5	3	Robust
			5	2	Robust
			4	3	Robust
			4	2	Robust
			4	1	Robust
			3	2	Robust
			2	1	Robust

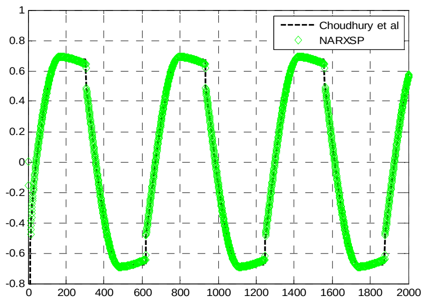


Fig. 20 NARXSP NN for test set *S=5* and *J=1*

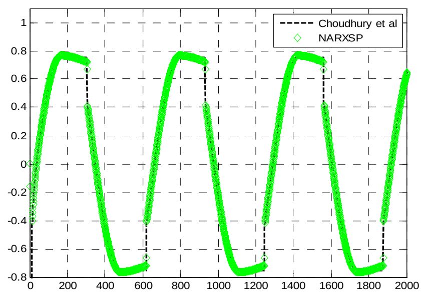


Fig. 21 NARXSP NN for test set *S=5* and *J=2*

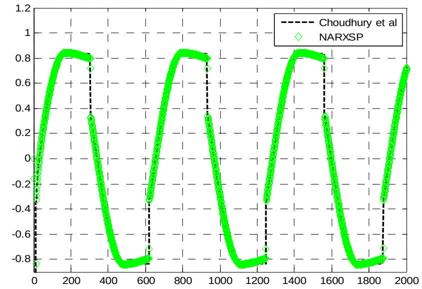


Fig. 22 NARXSP NN for test set *S=5* and *J=3*

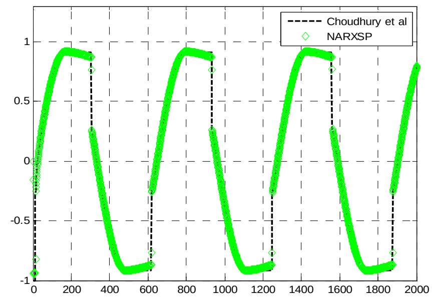


Fig. 23 NARXSP NN for test set *S=5* and *J=4*

Robustness analysis against other stiction types

In this section, the NARXSP base model that has been developed for stiction undershoot case ($S=3$ and $J=1$) is tested against three other types of stiction, namely stiction overshoot (*S=3*, *J=1*), pure deadband (*S=1*, *J=0*) and no offset (*S=J=1*). From Fig. 24-26, it can be clearly observed that there is a perfect match between NARXSP stiction model and Choudhury's stiction model. The stiction undershoot NARXSP model is capable to predict the behavior of the other stiction types excellently provided that the stiction parameter values, i.e. *S* and *J*, are within reasonable limits to the base model.

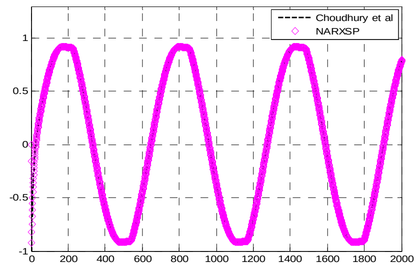


Fig. 24. Stiction undershoot NARXSP NN for test set *S=1* and *J=0*

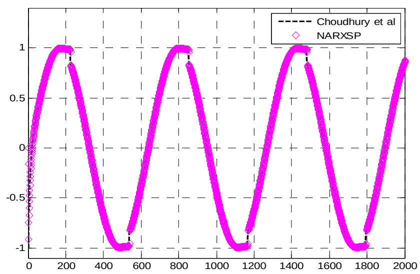


Fig. 25. Stiction undershoot NARXSP NN for test set *S=J=1*

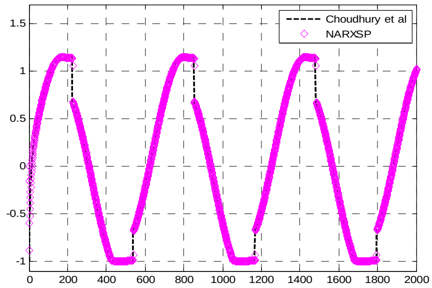


Fig. 26 Stiction undershoot NARXSP NN for test set *S=1* and *J=3*

Table 3 summarizes the same robustness analysis for other types of stiction. In general, the NARXSP model has satisfactory robustness properties when subjected to uncertainty in the type of stiction that exists in the system. However, one exception can be seen in the stiction no offset type. For this case, the robustness margin is very small.

TABLE III
PERFORMANCE SUMMARY FOR ANALYSIS IN ROBUSTNESS AGAINST OTHER STICTION TYPES.

Stiction Type	Base		Test		Performance
	S	J	S	J	
Pure deadband	1	0	1	1	Robust
			1	3	Robust
			3	1	Robust
No offset	1	1	1	0	Robust
			1	3	Not robust
			3	1	Not robust
Stiction overshoot	1	3	1	0	Robust
			1	1	Robust
			3	1	Robust
Stiction undershoot	3	1	1	0	Robust
			1	1	Robust
			1	3	Robust

C. *Predictor vs. model modes of operation*

It is widely accepted that NARXSP structure always results in excellent performance since the actual output available during training is fed back to the network as part of the inputs for prediction [13], i.e. the process outputs are predicted a single time step into the future. Consequently, a significant disadvantage of this mode of operation, termed the predictor mode, is the inability of the model to be used independently from the plant. An alternative as proposed by [13] is to use the trained NARXSP network in the parallel (feedback) architecture, where the predicted output from the network is being delayed and fed back along with the input to the network. This alternative mode of operation is called model mode.

In this study, the network is first trained using the series-parallel NARX network (NARXSP) or predictor mode. The network is then converted to the model mode (or parallel (feedback) form) using 'sp2narx' function in MATLAB. Its performance is then being evaluated against the validated and proven Choudhury's model for stiction undershoot case (i.e. *S=3* and *J=1*) for 10000 time steps into the future. Fig. 24 to 25 showed the performance of the model.

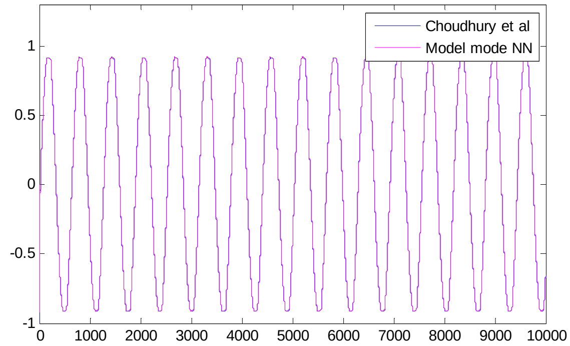


Fig. 24 NARXSP NN model vs. Choudhury et al for stiction undershoot

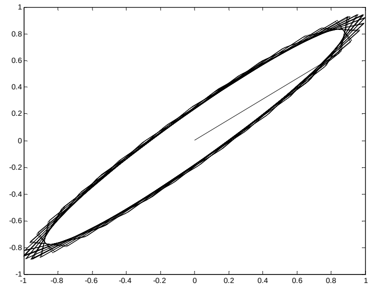


Fig. 25 PV vs. OP plot of NARXSP NN model

As can be observed from figures, the resulting stiction model structure is able to follow the stiction behavior as satisfactorily as the siction model of Choudhury. There is insignificant deviation at the top and bottom peak when the signal is changing direction.

V. Conclusion

In this work, a black box Neural Network-based modeling approach is proposed in modeling control valve stiction. Numerical evaluations showed that with optimized model structures, NARXSP NN stiction model is able to predict the control valve behavior in all four types of stiction to sufficient accuracy. Robustness analysis against the uncertainty in the stiction parameters (*S* and *J*) is tested under various conditions. It is shown that the NARXSP-based stiction model is robust when the stiction is less than 6% of the valve travel span for the same type of stiction behavior. For different types of stiction, the NARXSP-based stiction model is fairly robust to with the exception of stiction no offset. It is also found that parallel (feedback) network trained using the series-parallel form (NARXSP) is able to provide multi-steps ahead prediction with sufficient accuracy.

Acknowledgment

The authors would like to thank Universiti Teknologi PETRONAS for the funding provided for this research.

References

[1] M. A. A. S. Choudhury, S. L. Shah, & N. F. Thornhill, Diagnosis of poor control-loop performance using higher-order statistics, *Automatica*, *40*, 2004, 1719-1728.

[2] Choudhury, M. A. A. S., Shah, S. L., and Thornhill, N. F. (2004). Diagnosis of poor control-loop performance using higher-order statistics. Automatica, 40, pp. 1719-1728.

[3] Zabiri, H., and Samyudia, Y. (2006). A hybrid formulation and design of model predictive control for systems under actuator saturation and backlash. Journal of Process Control, 16, pp. 693-709.

[4] Choudhury, M. A. A. S., Thornhill, N. F., and Shah, S. L. (2005). Modeling valve stiction. Control Engineering Practice, 13, pp. 641-658.

[5] Muller, F. (1994). Simulation of an air operated sticky flow control valve. Proceedings of the 1994 Summer Computer Simulation Conferences, pp. 742-745.

[6] Kano, M., H., Kugemoto, H., and Shimizu, K. (2004). Practical model and detection algorithm for valve stiction. In Proceedings of the Seventh IFAC-DYCOPS Symposium, Boston, USA.

[7] Siegelmann, H. T., Horne, B. G., and Giles, C. L., Computational Capabilites of Recurrent Neural Networks, IEEE Transactions on Systems, Man. And Cybernetics – Part B: Cybernetics, 27 (2), pp. 208-215.

[8] Choudhury, M A. A. S., Kariwala, V., Shah, S. L., Douke, H., Takada, H., and Thornhill, N. F. (2005). A simple test to confirm control valve stiction. IFAC World Congress, Praha.

[9] Hagan M.T., Demuth H.B., Beale M.H. (1996). *Neural Network Design*, PWS Publishing Company, Boston, MA.

[10] Radhakrishnan, V. R., Zabiri, H., and Thanh, D. V. (2006). Application of Multivariable Modeling in the Hydrocarbon Industry, *International Conference on Computer Process Control*, Lake Louise, Alberta Canada, January 10-16.

[11] Qahwaji, R. and T. Colak, Neural Network-based Prediction of Solar Activities, in CITSA2006: Orlando. (2006).

[12] Kim, J., Mowat, A., Poole, P., and Kasabov, N., Linear And Non-Linear Pattern Recognition Models For Classification Of Fruit From Visible-Near Infrared Spectra, Chemometrics And Intelligent Laboratory Systems, 2000. 51: pp.,201-216.

[13] Gomm, J. B., D. Williams, J. T. Evans (1996). Development of a Neural-Network Predictive Controller, Liverpool John Moores University,UK.