12th IFAC International Symposium on
Advanced Control of Chemical Processes
July 14-17, 2024. Toronto, Canada

Control Valve Stiction Detection using Learning Vector Quantization Neural Network

Seshu K. Damarla*. Biao Huang*.

*Chemical and Materials Engineering, University of Alberta, Alberta, Canada
(damarla@ualberta.ca, bhuang@ualberta.ca).

Abstract: The performance of a process control loop can be limited when nonlinear problems like deadband, hysteresis, backlash, stiction, etc. exist in control valve. Stiction occurs more frequently than the other valve problems and has potential to cause adverse oscillations in the control loop, resulting in poor quality products, excessive use of raw materials and energy, and an environmental footprint. Timely detection of sticky control valves can help control engineers to take appropriate actions (retuning the controller or using stiction compensation methods) to prevent further degradation of the performance of the control loop. In connection with the aforesaid fact, this work proposes a novel stiction detection method founded on learning vector quantization neural network (LVQNN). Simulated database is generated and used to train the LVQNN with the training algorithm: LVQ2.1. To further enhance the performance of the method, transfer learning is adopted to retrain the pre-trained LVQNN model by using industrial data. The retrained LVQNN is tested on practical data obtained from a wide variety of industries. Results highlight that the proposed method can outperform the existing methods.

Keywords: Learning vector quantization; valve stiction; oscillations; neural network; control loops.

1. INTRODUCTION

Control performance monitoring is acknowledged as a key factor in increasing the profitability of industries. A proficient monitoring system ought to have the capability to pinpoint control loops that are underperforming and then differentiate between different sources of malfunction to recommend the most relevant measures to take (Paulonis and Cox (2003)). The poor performance of the control loops can be ascribed to tight control tuning, process upsets, sensor malfunctioning and control valve problems. Control valves are frequently utilized in process control loops, which control fluid flow, pressure, and temperature in a wide range of industrial processes. They are vital mechanical devices for the control loops to maintain the key process variables at their respective desired values. However, control valve operation can be challenging in the presence of stiction (Desborough and Miller (1998)). Valve stiction introduces oscillations in the control loops, which can reduce control loop performance and product quality. Researchers from academia and practicing engineers have paid a great deal of attention to this industrial problem due to its role in improving performance of the control loops.

Manual stiction detection is impractical in plants with numerous valves, necessitating the use of non-intrusive and automated methods. Since Horch (1999) developed the first automatic non-invasive detection technique, several non-intrusive methods have been proposed. Even though there are many of these approaches, new ones are nevertheless suggested every year, which involve the usage of multivariate statistical techniques, machine and deep learning algorithms, statistical techniques, system identification methodologies and global optimization algorithms. The book written by Jelali and Huang (2009) discusses various automatic stiction detection methods reported in the literature up to 2009. Details on the methods developed afterwards can be found in Zheng et al. (2021) and Bacci di Capaci and Scali (2018).
However, some of the existing methods either exhibit poor stiction detection performance on industrial case studies while the other methods are too complicated to be used in practice. This inspires the authors of this study to propose an innovative stiction detection approach utilizing a supervised machine learning algorithm: the learning vector quantization neural network (LVQNN). The rest of the paper is organized as follows. In Section 2, the problem of control valve stiction is revisited. The proposed methodology is discussed in detail in Section 3. In Section 4, the training of the LVQNN and its application to industrial benchmark case studies are discussed. Section 5 concludes the work.

2. CONTROL VALVE STICTION

Fig. 1 portrays the behaviour of a control valve impacted by stiction. If the control valve functions without abnormality, commands (controller output (OP)) generated by the controller are perfectly executed by the control valve. In this case, control valve position (MV) precisely matches OP, which is denoted by the straight line passing through the origin. But because of valve stiction, the previously existing linear correlation between OP and MV has dissipated. As a result, they are in nonlinear relationship represented by the parallelogram that contains three phases: stiction band, slip-jump and moving phase. The stiction band indicates the dormant phase of the control valve and there is no change in the valve position despite continuous variation in OP. When an aggregate change in OP is high enough to move the valve out of stiction, then MV suddenly moves (this sudden jump is called slip-jump) and keeps changing until OP changes its direction. During operation, the control valve can subject to stiction multiple times with various amounts of stiction band.

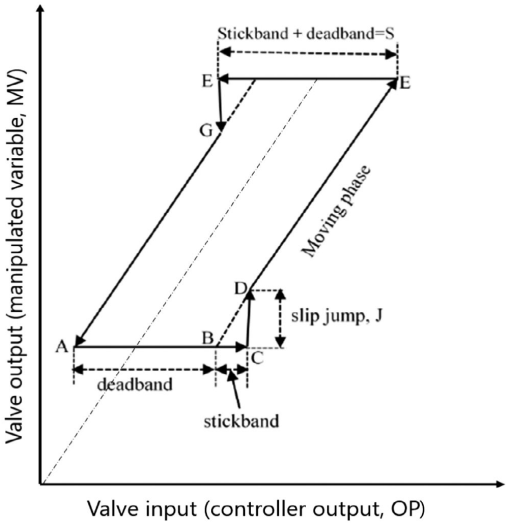

Figure 1. Sticky control valve.

3. THE PROPOSED METHOD

The main purpose of the present work is to develop a non-contact (or non-invasive) type stiction detection methodology, which is based on LVQNN. In most process industries, OP and process variable (PV) are routinely measured whereas MV is rarely obtained. Hence, the proposed method uses those frequently measured signals only. This choice is justifiable because control valve stiction unquestionably affects control loops, so, PV deviates from its setpoint and cycles continuously or intermittently with constant or variable amplitude. Therefore, information regarding control valve stiction can be obtained from OP and PV signals (Shoukat Choudhury et al. (2008)). In the present work, the preprocessing method introduced in (da Silva Mendonça et al. (2017)) is adopted to convert a pair of PV and OP signals into the D signal defined in the equation given below.

$$D_i = \sqrt{(PV_i - PV_m)^2 + (OP_i - OP_m)^2}, \qquad (1)$$

where $i = 1,2,\ldots,P$, P is the number of data points in PV or OP and PV_m is the mean of PV and OP_m is the mean of OP.

Kohonen (1997) created LVQNN to use labeled data to solve multi-class classification problems [9]. The operation of LVQNN is similar to that of k-nearest neighbors' algorithm. The prime goal of LVQNN is to find groups in the input data of training data, and assigns a given test input to one of those groups, which is nearest to the test input (da Silva et al. (2017)). Fig. 2 shows the structure of LVQNN.

The input layer establishes a connection between the network and the actual environment i.e. detecting stiction. The neural network gets training samples (D signals derived from pairs of OP and PV) via the input layer. All of the D signals in the training data are assumed to have the same number of data points. The input layer can have the same number of neurons as the length of a single D signal. The neurons in the input layer simply forward the data they get to the neurons in the hidden layer. Codebook vectors are the weights that link the input layer to the hidden layer.

. Codebook vectors are the weights that link the input layer to the hidden layer.

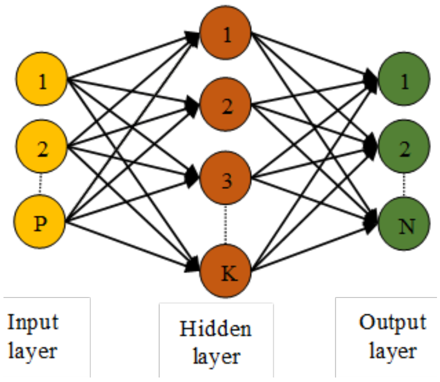

Figure 2. Architecture of LVQNN

The next equation provides a matrix of codebook vectors.

$$W^{(1)} = \left[w_1^{(1)}, w_2^{(1)}, \ldots, w_K^{(1)} \right]^T, \qquad (2)$$

where T denotes transpose.

In the above equation, the weights of the i[th] neuron in the hidden layer are given by $w_i^{(1)}$.

The hidden layer is also referred to as competitive layer since the neural network utilizes winner-take-all learning strategy to modify the weights of the middle layer neurons. Each neuron in the output layer is associated with a subset of the hidden layer neurons. The number of neurons in the output layer is equal to the number of groups in the input space; hence each output layer neuron signifies one class region. There can be the same number of subclasses for each of the output layer neurons. While training of the neural network is in progress, the codebook vectors undergo continuous modification until there is no difference between predicted class labels and the actual class labels. After the weights of the output layer neurons are initialized, they will not be updated.

Assume that the competitive layer has 6 neurons, and the output layer has 3 neurons, i.e. there are three class regions or groups in the training data. The first two hidden layer neurons belong to the first class (i.e. the first output neuron), the third and the fourth hidden layer neurons represent the second class (the second output neuron) and the last two hidden layer neurons are in class region 3 (the third output neuron). Under this circumstance, the weights of the output layer neurons can be fixed as shown below.

$$W^{(2)} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}. \qquad (3)$$

For the first training sample, the Euclidean distance between the sample and each of the codebook vectors is calculated. The codebook vector with the shortest Euclidean distance to the sample is declared the winner. The output of winner neuron is one, and the output of the remaining neurons is zero. The output of the hidden layer for the first training sample is

$$a^{(1)} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (4)$$

Here it is assumed that the second hidden layer neuron wins the competition.

The output of the LVQ neural network is

$$a^{(2)} = W^{(2)} a^{(1)} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}. \quad (5)$$

The LVQ neural network predicts that the first training sample belongs to class region 1. This process of computing neural network outputs for the remaining training samples is continued. Among the learning algorithms available to train the LVQNN, LVQ2.1 is found to be the most effective in minimizing misclassification rate. This training algorithm is described in the following.

LVQ2.1 is an improvement of LVQ2 training algorithm. LVQ2.1 simultaneously updates the codebook vectors of two hidden layer neurons which are the nearest neighbours to the training sample. One additional constraint which needs to be satisfied by the training sample is that the training sample must fall into a window defined around midplane of the codebook vectors of two neurons closest to it. It is to be noted that the winning neurons are from different classes.

If the following inequality relation holds, then the training sample is considered to be in the window.

$$\min\left(\frac{d_j}{d_i}, \frac{d_i}{d_j}\right) > \left(\frac{1-w}{1+w}\right), \quad (6)$$

where d_j is the Euclidean distance between the codebook vector of neuron j and the training sample, d_i is the Euclidean distance between the codebook vector of neuron i and the training sample, w is the window length. It is to be recalled that neuron j and neuron i win the competition.

The rule to change the codebook vectors of neuron j and neuron i are

$$\left.\begin{aligned} w_j^{(1)} &= w_j^{(1)} + \alpha\left(x - w_j^{(1)}\right) \\ w_i^{(1)} &= w_i^{(1)} - \alpha\left(x - w_i^{(1)}\right) \end{aligned}\right\}. \quad (7)$$

$$\left.\begin{aligned} w_j^{(1)} &= w_j^{(1)} - \alpha\left(x - w_j^{(1)}\right) \\ w_i^{(1)} &= w_i^{(1)} + \alpha\left(x - w_i^{(1)}\right) \end{aligned}\right\}. \quad (8)$$

In the above equations, α is called the learning rate. The learning rule in Eq. (7) is used when neuron j correctly classifies the training sample and neuron i incorrectly classifies the training sample. If neuron j incorrectly classifies and neuron i correctly classifies the training sample, then the learning rule in Eq. (8) is employed.

4. RESULTS AND DISCUSSIONS

As the proposed method relies on the supervised learning algorithm, labeled data is needed to learn the codebook vectors of the hidden layer neurons. The authors in Jelali and Huang (2009) formed a database called ISDB containing practical data acquired from various industries. However, the available practical data is still not sufficient to adequately train LVQNN. Therefore, a simulated database was generated by creating the control loop, shown in Fig. 3, under various scenarios.

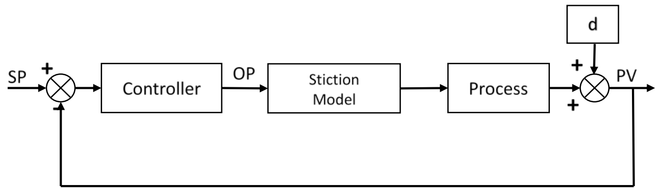

Figure 3. Process control loop

Both self-regulating (concentration) and integrating (level) processes were considered to produce oscillatory PV and OP. Proportional integral controllers was employed in each control loop. The following equations provide the mathematical models for the process and the controller of the concentration loop and the level loop, respectively.

$$G_p = \frac{3e^{-10s}}{10s+1}, \; G_c = 0.2\left(\frac{10s+1}{10s}\right) \quad (9)$$

$$G_p = \frac{1}{s}, \; G_c = 0.4\left(\frac{2s+1}{2s}\right) \quad (10)$$

The data-based stiction model introduced in (Shoukat Choudhury et al. (2005)) was utilized in the present work to create stiction-caused oscillations in the control loops. White noise with variance V was added to PV. Table 1 provides ranges selected for stiction band (S), slip-jump (J) and noise variance (V).

Table 1 Parameter range for producing stiction-induced oscillations

Parameter	Range
S	[0.5: 0.25: 10]
J	[0.1: 0.25: 5]
V	0.01

As discussed before, proportional integral controllers with excess integral action can also introduce oscillations in the control loops. According to the parameter values given in Table 2, oscillatory data was generated. In this case too, PV was corrupted with noise.

Table 2 Parameter range for producing oscillatory data (tightly-tuned controllers)

Parameter	Range
K_c	$[0.1{:}\, 0.01{:}\, 0.3]$
τ_I	$[0.01{:}\, 0.01{:}\, 0.27]$
V	0.01

Table 3 Parameter range for creating oscillatory data (external oscillatory disturbances)

Parameter	Range
A	[1,1.5,2,2.5]
F	$0.01{:}\, 0.01{:}\, 0.27$
V	0.01

Table 4 Configuration and training details

Parameter	**Value**
No. of hidden layers	1
No. of hidden neurons	20
No. of epochs	200
Length of each D signal	300
Learning rate	0.015
Training algorithm	LVQ2.1
Objective function	MSE

To simulate external oscillatory disturbances, sinusoidal signal with amplitude A and frequency F was added to the control loop at the disturbance input d. By using different values for A and F as given in Table 3, several oscillatory PV and OP signals were created. Once the simulation database was created, D signals were produced from each pair of PV and OP signals belonging to stiction and non-stiction datasets. If a D signal is obtained from a pair of PV and OP representing stiction condition, then target vector for the neural network is [1 0]. The target vector changes to [0 1] for the D signals signifying non-stiction conditions. Table 4 provides hyper-parameters and training details of the LVQNN. The LVQNN was trained using LVQ2.1 delineated above.

As the simulate data generally do not contain all the characteristics of the industrial data, some of the data available in ISDB were used to retrain the trained LVQNN model. The retrained LVQNN was tested on the remaining industrial data obtainable in ISDB. The details of the industrial control loops and the results are given in Table 5. The full forms for the acronyms used in Table 5 are provided in Table 6. As per Tables 5 and 7, the proposed method provided correct diagnosis for the 22 control loops out of the 26 control loops studied. It can be noticed from Table 8 that the LVQNN based stiction detection methodology outperformed the existing methods considered for the comparison.

Table 5 Results for ISDB control loops

LN	**CL**	**AM**	**VIM**	**IDC**
CHE 1	FC	STN	STN	Yes
CHE 2	FC	STN	STN	Yes
CHE 3	TC	NSTN	NSTN	Yes
CHE 4	LC	NSTN	NSTN	Yes
CHE 5	FC	STN	STN	No
CHE 6	FC	STN	STN	Yes
CHE 10	PC	STN	STN	Yes
CHE 11	FC	STN	STN	Yes
CHE 12	FC	STN	STN	Yes
CHE 13	AC	NSTN	STN	Yes
CHE 14	FC	NSTN	NSTN	Yes
CHE 16	PC	NSTN	NSTN	Yes
CHE 23	FC	STN	NSTN	No
CHE 24	FC	STN	STN	Yes
CHE 26	LC	STN	STN	Yes
CHE 29	FC	STN	STN	Yes
CHE 32	FC	STN	STN	Yes
CHE 33	FC	NSTN	NSTN	Yes
CHE 34	FC	NSTN	NSTN	Yes
CHE 58	FC	NSTN	NSTN	Yes
MIN 1	TC	STN	STN	Yes
PAP 2	FC	STN	STN	Yes
PAP 4	CC	NSTN	NSTN	Yes
PAP 5	CC	STN	STN	Yes
PAP 7	FC	NSTN	STN	No
PAP 9	TC	NSTN	STN	No

Table 6 Configuration and training details

Acronym	**Full form**
IDC	Is diagnosis correct
AM	Actual malfunction
CL	Control loop
LN	Loop name
FC	Flow control
PC	Pressure control
CC	Concentration control
AC	Analyzer control
TC	Temperature control
STN	Stiction
NSTN	No stiction
VID	Verdict issued by method
LC	Level control

Table 7 Performance of proposed method

Performance Metric	Value
True positive	14
True negative	8
False positive	1
False negative	3
Precision	0.933
Recall	0.8235
Specificity	0.8889
F1 score	0.875
Accuracy	0.8462

Table 8 Comparison with existing methods

Method	AL	NAL	NCD
BIC (Jelali and Huang (2009))	24	2	19
Horch Method 1 (Horch (1999))	24	2	14
Horch Method 2 (Jelali and Huang (2009))	25	1	16
Rossi And Scali's Method (Jelali and Huang (2009))	26	0	17
He's Method (Jelali and Huang (2009))	25	1	12
Singhal and Salsbury's Method (Jelali and Huang (2009))	26	0	11
Le's Method (Jelali and Huang (2009))	26	0	18
Karra and Karim's Method (Jelali and Huang (2009))	26	0	18
SLOPE Method (Jelali and Huang (2009))	25	1	14
ZONES Method (Jelali and Huang (2009))	25	1	15
BSD Method (Kamaruddin et al. (2020))	26	0	20
Proposed method	26	0	22

AL – applicable loops, NAL – not applicable loops, NCD – number of correct diagnoses

5. CONCLUSIONS

Sticky control valves are a common source of oscillations in industrial control loops, which ultimately result in production loss and reduced profits. Timely identification of sticky control valves is crucial. In the present work, a simple stiction detection method was devised with the help of the LVQNN. Compared to the existing methods, the proposed method demonstrated superior performance by providing correct diagnosis for the 22 control loops out of the 26 control loops.

REFERENCES

Bacci di Capaci, R. and Scali, C. (2018). Review and comparison of techniques of analysis of valve stiction: From modeling to smart diagnosis. *Chem. Eng. Res. Des.*, 130, 230-265.

da Silva Mendonça, R., Duarte Cavalcante, A. L., and de Lucena, V. F. (2017). Proposal of a simulator for evolutionary production systems. *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA),* Limassol, Cyprus, pp. 1-4.

da Silva, I. V., Spatti, D. H., Flauzino, R. A., Liboni, L. H. B., Liboni, L. H. B., dos Reis Alves, S. F. (2017). *Artificial neural networks: A practical course,* 1-309. Springer International Publishing, Switzerland.

Desborough, L., and Miller, R. (1998). Increasing customer value of industrial control performance monitoring-honeywell's experience. *In AIChE symposium series*, 169-189, New York, 1998.

Horch, A. (1999). A simple method for detection of stiction in control valves. *Control Eng. Pract.*, 7, 1221-1231.

Jelali, M., and Huang, B. (2009). *Detection and diagnosis of stiction in control loops: State of the art and advanced methods*, 1-389. Springer-Verlag, London.

Kamaruddin, B., Zabiri, H., Mohd Amiruddin, A. A. A., Teh, W. K., Ramasamy, M., and Jeremiah, S. S. (2020). A simple model-free butterfly shape-based detection (BSD) method integrated with deep learning cnn for valve stiction detection and quantification. *J. Process Control*, 87, 1−16.

Kohonen, T. (1997). *Self-organising maps*. Springer, Berlin.

Paulonis, M. A., and Cox, J. W. (2003). A practical approach for large-scale controller performance assessment, diagnosis, and improvement. *J. Process Control*, 13, 155-168.

Shoukat Choudhury, M. A. A., Shah, S. L., and Thornhill, N. F. (2008). *Diagnosis of process nonlinearities and valve stiction: Data driven approaches*, 1-293. Springer-Verlag, Berlin Heidelberg.

Shoukat Choudhury, M. A. A., Thornhill, N. F., Shah, S. L. (2005). Modelling valve stiction. *Control Eng. Pract.*, 13, 641−658.

Venceslau, A. R., Guedes, L. A., and Silva, D. R. in Proc. of 2012 IEEE 17th Int. Conf. on Emerging Technologies & Factory Automation (ETFA 2012) (Ed: E. A. Gutierrez-D), IEEE, Krakow, Poland 2012, pp. 1–4.

Zheng, D., Sun, X., Damarla, S. K., Shah, A., Amalraj, J., and Huang, B. (2021). Valve stiction detection and quantification using a k-means clustering based moving window approach. *Ind. Eng. Chem. Res.*, 60(6), 2563-2577.