Quantification of Valve Stiction using Particle Swarm Optimisation with Linear Decrease Inertia Weight

Pongsurachat Aksornsri[1] and **Sarawan Wongsa**[2], Non-members

ABSTRACT

Valve stiction is one of the most common problems on industrial process control loops. The detection and quantification of valve stiction in control loops is therefore important to ensure the high quality of the products and maintain the reliable performance of control loops. This paper presents an algorithm for quantifying valve stiction in control loop based on linear decrease inertia weight particle swarm optimisation to obtain more accurate estimates of stiction parameters. The amount of stiction present in the valve is estimated by identifying parameters of Kano model which is a two-parameter data-driven stiction modelling based on the parallelogram of MV-PV phase plot. Simulation results have demonstrated the efficacy of this algorithm in valve stiction quantification and also its robustness to oscillations due to inappropriate controller tuning and external disturbances. Results are confirmed by application to real process industrial data.

Keywords: Valve Stiction Quantification, Particle Swarm Optimisation, Linear Decrease Inertia Weight, Kano Model

1. INTRODUCTION

Valve stiction problem has been known to be one of the main causes of performance deterioration in control loops. Stiction induces oscillations in process variables and cannot be eliminated by controller detuning. Therefore, detecting and quantifying this valve problem is essential to identify the sticky valves so that they can be isolated and repaired before degrading the performance of the control loop and affecting the product quality [1, 2].

In the literature, there are a significant number of methodologies to detect and quantify the valve stiction [3], including the nonlinear analysis [1], model-based methods [4], and the shape analysis [5, 6]. The methods belonging to the nonlinear analysis utilises the fact that the presence of stiction in a control valve introduces nonlinearity in the control loop and often produces non-Gaussian time series. Therefore, the very first step of detecting valve stiction is to analyse whether there is nonlinearity and non-Gaussian data in the control loop. The most-used techniques for detection of valve or process nonlinearity are higher-order statistical methods. In Choudhury et al. [7], a non-Gaussianity index (NGI) and a nonliearity index (NLI) have been defined using the bicoherence of the signal to quantify the size of the non-Gaussianity and nonlinearity in control loops. However, in order to detecting valve stiction the nonlinearity detection technique has to be combined with other methods, typically with ellipse fitting [1].

The shape-based methods have been shown to be reliable for different types of control loops including flow control and level control loops [5, 6]. The shape-based methods use only routine operation data for detecting stiction. The shape is considered from the relationship between controller output (OP) and manipulated variable (MV) in the two-dimensional space. In practice, flow rate is used as MV instead of the valve position if MV is not available. When stiction occurs, it produces a special shape in the phase plot of OP and MV. Based on the obtained shape, a pre-determined stiction metric is calculated to quantify the valve stiction. In Kano et al. [5], stiction is detected by detecting a parallelogram shown in Fig. 1. When stiction occurs, MV stays constant while OP decreases or increases. The degree of stiction can be evaluated by taking the length where MV stays constant into account.
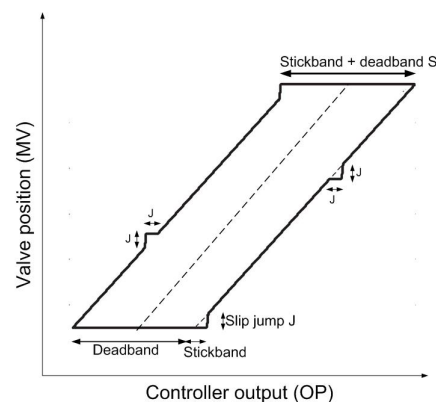

Fig.1: *Typical MV-OP characteristic of a sticky valve.*

The main difficulty of using stiction metrics proposed in [5] in practice is that there are several predefined parameters that need to be selected, such as the thresholds for the difference between the maximum and the minimum of OP and MV when stiction occurs. However, there is no systematic way to select appropriate threshold values presented in their work. Moreover, the proposed metrics are unscaled. This makes it hard to gauge how strong the stiction is after the stiction size is quantified in the form of these metrics. In order to address this problem, it might be better to diagnose the valve stiction by directly identifying the stiction parameters which are more physically meaningful.

In this work, we propose a stiction quantification method that incorporates the stiction parameter estimation by particle swarm optimation (PSO) with the two-parameter data-driven stiction modelling. We apply the concept of linear decrease inertia weight in PSO to obtain more accurate estimates. As the identified parameters are the direct measure of the amount of valve stiction, they intuitively provide the interpretation of how strong the stiction is. We show through simulation how the proposed stiction quantification can be carried out and that the algorithm is not affected by controller tuning.

2. STICTION MODELLING & QUANTIFICATION

Consider a typical valve-controlled loop in Fig. 2, where a valve is included between the process and the controller. The controller compares the process variable (PV), for example the level, flow rate, and pressure, to the desired process condition or the set point (SP) and sends the controller output (OP) signal to correctively adjust the manipulated variable (MV) which is the valve travel position in this case (units of % opening), such that the PV well tracks the SP.
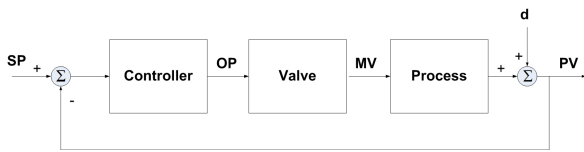

Fig.2: *Typical scheme of feedback valve-controlled loop. (SP: set point; OP: controller output; MV: manipulated variable/valve position; PV: process variable.*

A normal valve has the linear relationship between MV and OP. However, in the presence of stiction the relationship is changed to a nonlinear shape as shown in Fig. 3. Stiction describes the situation where the valve's stem is sticking when small changes are attempted. It occurs when the static friction exceeds the dynamic friction inside the valve [5]. As a consequence, the valve position cannot be changed un-
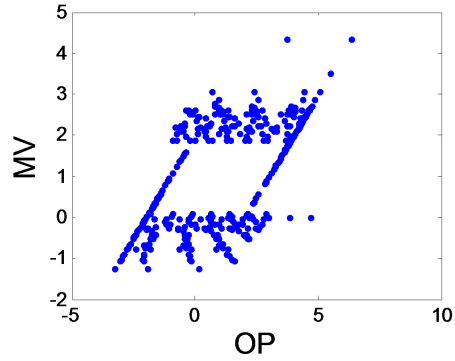

Fig.3: *Typical input-output behavior of a sticky valve.*

til OP overcomes the static friction. In presence of stiction, the movement of the position becomes unsmooth and jumpy.

2.1 Stiction Modelling

In the literature, there are 2 major categories of stiction modelling: the physically-based modelling and the data-driven modelling. The physically-based models attempt to describe the friction phenomenon that causes the valve stiction by using the balance of forces and Newton's second law of motion. Examples are the model equations that describe the sticking or sliding body [8] and dynamic friction model proposed in [9]. For more details, see [2] and references therein. A detailed physically-based stiction model has a major problem to be applied in a real plant because it requires the knowledge of several critical parameters that are difficult to estimate.

Instead of relying on the first principles to model the stiction, data-driven stiction models use extensive collection of data to establish the detection and quantification of the valve stiction. The data-driven modelling requires only a few parameters to identify and has become more favorable in the literature in the recent years. Some of the data-driven models need only one parameter to be identify, such as the stiction model proposed by Stenman [4]. The problem with the one-parameter data-driven model however is that it cannot capture some input-output behaviours, such as the phase plot, of the real sticky valve during the fast stoke test [1].

The two parameter data-driven model has been proposed to more accurately capture the dynamics of the valve stiction. The Choudhury model [10] and the Kano model [5] require two parameters: the J and S parameters which represent the size of the stem slip and the combination of the stickband and deadband, respectively, while the He model [6] describes the valve stiction through the parameters f_S (static friction band) and f_D (kinetic friction band).

Garcia [11] has performed tests according to ISA

standards on several different valve stiction models, including the Choudhury, the Kano, and the He models, under different input signals and with valve with different friction coefficients. He demonstrated that among these three models, the Kano model was the only model that could represent the expected stick-slip phenomenon of the sticky valve. The Kano model also has other advantages, including its ability to cope with both the stochastic and deterministic inputs and to change the degree of stiction according to the direction of the valve movement [5]. Therefore in this work, the Kano model is used to describe and simulate the stiction nonlinearity.

The Kano model describes the valve stiction using the parallelogram of the MV-OP plot, shown in Fig. 3. The deadband and stickband represent the behaviour of the valve when it is not moving through the controller output keeps changing. The magnitude of the deadband and stickband is estimated as the S parameter. The slip jump J presents the abrupt release due to the conversion of potential energy stored in the actuator to kinetic energy.

The magnitudes of S and J determine the characteristics of oscillation caused by valve stiction. The deadband provokes oscillations and reduces performance of the control loops. The magnitudes of J are also crucial to determine the amplitudes and frequencies of limit cycles [1]. Stiction is measured in percent of the controller output necessary to move the valve stem. Generally, 1% of stiction is considered enough to cause performance problems [3].

The algorithm of Kano model can be presented in a flowchart shown in Fig. 4 [5]. The input u and output y of this model are the OP and MV, respectively. The variable $stp = \{0, 1\}$ represents the moving ($stp = 1$) or resting states ($stp = 0$) of the stem, u_s represents the controller output at the moment the valve state changes from moving to resting, and $d = \pm 1$ indicates the direction of frictional force. When the valve stops or changes its direction while its state is moving, us is updated and changes its state to the resting state. Then, in the resting state the valve will change its state to moving if i) the valve changes its direction and overcomes the maximum static friction and/or ii) the valve moves in the similar direction and overcomes friction. After the valve changes its state to moving, the valve position is updated via the following equation:

$$y(t) = u(t) - \frac{d(S - J)}{2} \qquad (1)$$

2.2 Stiction Detection Method based on Parallelogram

The stiction detection algorithm, proposed by Kano et al. [5], uses the parallelogram in Fig. 1 to consider the sections where the valve does not change even though the controller output changes.
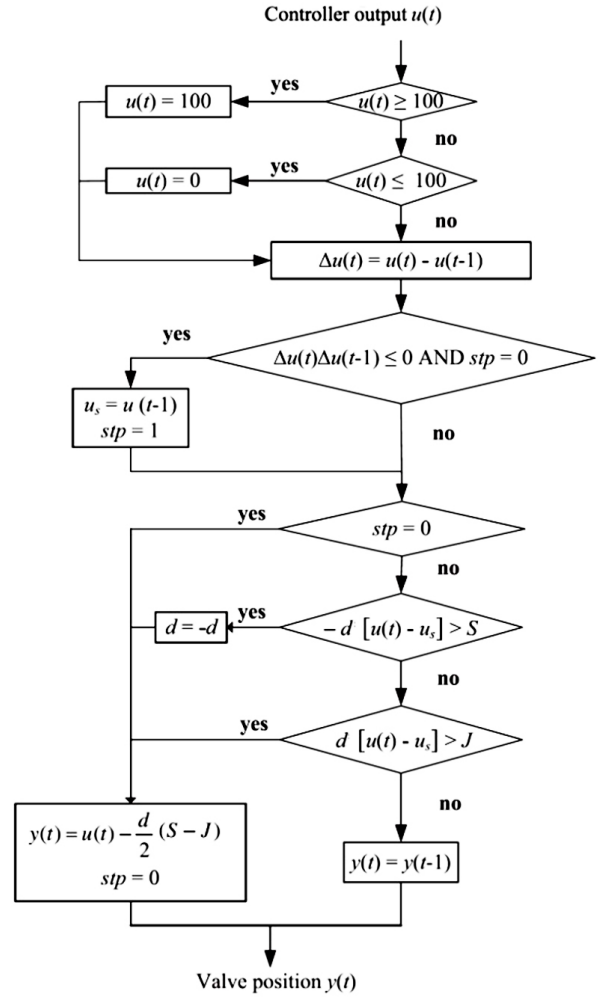

Fig.4: *Flowchart for the Kano model [5].*

The longer such sections are, the stronger the stiction is. The possibility of stiction is estimated as a ratio between the total length of intervals when stiction occurs to total length of all intervals. The stiction size is quantified by calculating the mean of the difference between the maximum and the minimum of the controller output (defined as $\tilde{u}$) when stiction occurs.

Even though this algorithm has been illustrated its success in stiction detection and quantification via both simulation studies and real chemical processes, the main problems are that there is no information on how to optimally tune several parameters, including the thresholds ε, ε_u and ε_y, and the resulting indicator σ, which quantifies the degree of stiction, has no direct connection to the parallelogram. Therefore, it is hard to gauge how strong the stiction is once the parameter σ is obtained.

To solve these limitations we propose a particle swarm optimation-based technique to directly estimate the stiction parameters J and S which have a more direct interpretation to the valve stiction.

3. PROPOSED METHOD

Particle swarm optimisation (PSO) is a stochastic population-based optimisation method, motivated by social interaction behaviour of bird flocking. The advantages of PSO are its requirements of a few parameters to adjust and easiness in implementation. Moreover, it does not require linearity in the parameters that makes it suitable to the identification nonlinearities properties [12].

In the basic PSO algorithm, m particles are placed in the n-dimensional search space. The current position of each particle represents the potential solution of the problem evaluated by a pre-defined fitness function. In every iteration, each particle determines its movement according to the history of its own current and best locations, and the best location obtained so far by any particle in the population (global best PSO). The equations for updating velocity and position of each particle are:

$$v_{id} = v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) \quad (2)$$

$$x_{id} = x_{id} + v_{id} \quad (3)$$

where
$d = 1, 2, \ldots, n$ represents the dimension.
$i = 1, 2, \ldots, m$ represents the particle index, where m is the size of the swarm.
g is the index of the best particle in the swarm.
c_1 and c_2 are constants, called cognitive and social scaling parameters, respectively.
r_1 and r_2 are random numbers drawn from a uniform distribution from 0 to 1.
$\boldsymbol{x_i} = (x_{i1}, x_{i2}, \ldots, x_{in})^T$ represents the position vector of the i particle.
$\boldsymbol{v_i} = (v_{i1}, v_{i2}, \ldots, v_{in})^T$ represents the velocity vector of the i particle.
$\boldsymbol{p_i} = (p_{i1}, p_{i2}, \ldots, p_{in})^T$ represents the best previous position of the i particle.

The concept of the inertia weight was introduced in 1998 by Shi and Eberhart [16] and the proposed velocity update equation is:

$$v_{id} = w v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) \quad (4)$$

where w is the inertia weight.
The inertia weight is added to balance global exploration and local exploitation of the searching process. A large w facilitates a global search while a small one facilitates a local search. There have been a number of strategies proposed for adjusting the value of the inertia weight during a course of run, such as adaptive inertia weight strategy [13], chaotic inertia weight [14], and linearly decreasing strategy [15, 16]. According to the results of the comparative study of

15 different inertia weight strategies over five optimisation problems presented in [17], the linear decreasing inertia weight produced the best performance in terms of the minimum error in comparison to other methods.

In the linear decreasing inertia weight, the value of w is linearly decreased from an initial value (w_{max}) to a final value (w_{min}) according to the following equation:

$$w(iter) = \frac{iter_{max} - iter}{iter_{max}}(w_{max} - w_{min}) + w_{min} \quad (5)$$

where $iter$ is the current iteration of the algorithm and $iter_{max}$ is the maximum number of iterations allowed.

In this work, we use the PSO with the linear decreasing inertia weight to estimate the stiction parameters, J and S, of the control valve. The proposed framework is illustrated in Fig. 5.
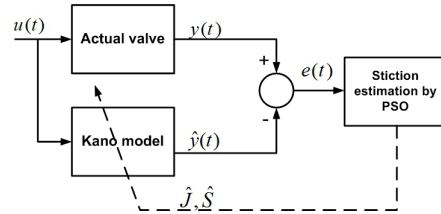

Fig.5: *PSO based parameter estimation procedure.*

The algorithms in Fig. 5 is summarised as follows:
i) Set up all PSO parameters and initialise the inertia weight to w_{max}.

ii) Initialise a population array of m particles with random positions and velocities on two dimensions, representing the parameters J and S of the stiction, in the search space.

iii) The fitness function for each particle in the initial population is evaluated. We select the mean squared error (MSE) as the fitness function for determining how well the estimates fit the system. The MSE_i of estimation for particle i is defined as

$$MSE_i = \frac{\boldsymbol{e_i^T e_i}}{N} \quad (6)$$

where $\boldsymbol{e_i} \in R^N = \boldsymbol{y_i} - \boldsymbol{\hat{y}_i}$ is error vector of the i particle, $\boldsymbol{\hat{y}_i} \in R^N = \{\hat{y}_i(1), \hat{y}_i(2), \ldots, \hat{y}_i(N)\}$ is the vector of estimated valve positions obtained from the Kano model, $\boldsymbol{y_i} \in R^N = \{y_i(1), y_i(2), \ldots, y_i(N)\}$ and is the vector of measured valve positions, and N is the number of input-output datapoints used in the identification. p_{id} is set to each initial searching point. The initial best evaluated value among p_{id} values is set to p_g.

iv) Update the velocity and position of each particle using (4) and (3).

v) Search with new position and the fitness func-

tions are calculated. If the fitness function of each particle is better than the previous p_{id} ,the value is set to p_{id}. If the best p_{id} is better than p_{gd}, the value is set to p_{gd}. All p_{gd} are stored as the current estimates of J and S.

vi) Update the inertia weight according to (5).

vii) Repeat from step iv) until the maximum number of iteration is exceeded.

4. SIMULATION EXAMPLES

The following experiments demonstrate the optimal configuration of the proposed stiction quantification method and its performance. The flow control system proposed in [5] is used to represent the industrial control loop. Fig. 6 shows the block diagram of the system.
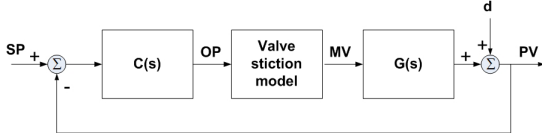

Fig.6: *Block diagram of the flow control system.*

The process transfer function of flow $G(s)$ is given by

$$G(s) = \frac{1}{0.2s + 1} \qquad (7)$$

The controller $C(s)$ is the proportional-integral (PI) controller which is implemented in the following form:

$$C(s) = K_p \left(1 + \frac{1}{\tau_i s}\right) \qquad (8)$$

The proportional gain (K_p) and reset time (τ_i) are set to 0.5 and 0.3 min, respectively. The sampling interval for flow control is 0.5 min.

Table 1: *Stiction Parameters used to Evaluate the Proposed Method.*

Case	$J(\%)$	$S(\%)$
No stiction	0	0
Weak stiction	0.3	1
Strong stiction	1	5

Table 1 summarises all three cases investigated in this study, including the normal case where there is no stiction, the weak stiction where $J = 0.3\%$ and $S = 1\%$ and the strong stiction where $J = 1\%$ and $S = 5\%$. For each pair of parameter $\{J, S\}$, we generate the input-output datapoints by simulation to represent the actual data. These data are then used in our stiction quatification procedure to determine the estimates of $\{J, S\}$ in the performance evaluation process.

It should be noted that generally there is no particular rule of thumb to strictly pinpoint how much stiction would represent each state of valve stiction. In practice, we could consider a valve to be *normal* even when it exhibits some behaviour of stiction if such magnitude of stiction does not greatly deteriorate the loop performance. However in order to be able to simulate three different states of the valve, in this simulation we selected the magnitudes of J and S similar to those in [5] to represent the normal, weak stiction, and strong stiction states. For each stiction case, a band-limited white noise is forced into the control loop as the setpoint to simulate the control loop and 100 sampling points of MV-OP data are used in our algorithm to quantify the stiction parameters J and S.

Table 2: *Parameters of LDIW-PSO Stiction Estimation.*

Parameter	Value
Population size	9, 25, 49
Number of iterations	40
Initial and final inertia weight	{0.9, 0.4}
c_1 and c_2	2

The parameters of the LDIW-PSO (linear decrease inertia weight-particle swarm optimisation) are listed in Table 2. The initial and final values of the inertia weight are set to 0.9 and 0.4, respectively, according to the suggestion in [16]. The coefficients c_1 and c_2 are set to the recommended values of 2. Note that the estimated J and S are constrained to be positive, meaning that when negative estimates are obtained, they are set to zero.

In the first experiment, we vary the number of particles or the population size to 9, 25, and 49 to investigate the estimation performance and use it as a guideline to select the appropriate population size for this problem. For each stiction case, 50 trials are carried out and the average MSE of parameter estimation are calculated. Note that this MSE is different from the one we used for calculating the fitness function of PSO in (6) and the MSE of the stiction estimation is defined as:

$$MSE_{stiction} = \frac{1}{50} \sum_{i=1}^{50} (\theta_i - \hat{\theta}_i)^2 \qquad (9)$$

where θ_i and $\hat{\theta}_i$ represent the actual values and the estimates of the parameters $\{J, S\}$ at the trial number i, respectively.

From the results of the average MSEs shown in Figs. 7 - 9, it can be clearly seen that the population size of 25 is sufficient for this problem because its MSE values are greatly reduced when compared with the MSE of the population size of 9 and by increasing it to be 49, the performances are not significantly improved, or for the normal case are even degraded. So

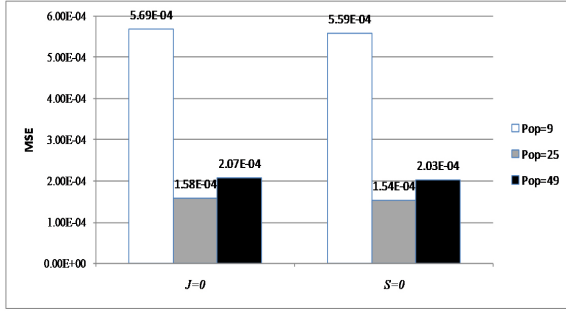the population size of 25 is used for further analyses.

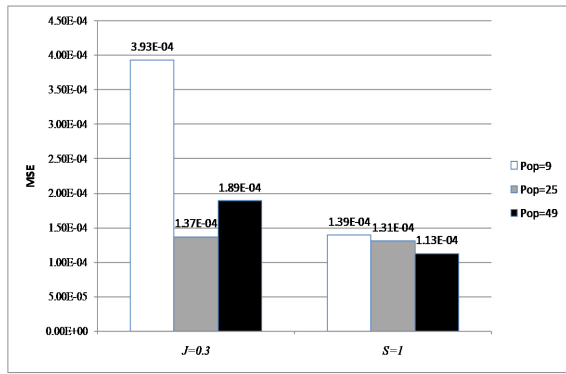**Fig.7:** *MSE in each population size for normal case.*

Fig.8: *MSE in each population size for weak stiction case. (a) J estimates (b) S estimates.*
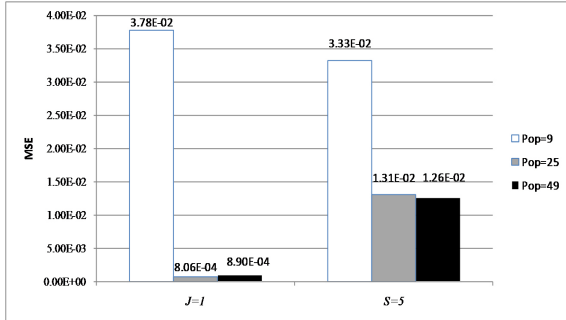

Fig.9: *MSE in each population size for strong stiction case. (a) J estimates (b) S estimates.*

In the next experiment, the LDIW-PSO is compared with the conventional PSO which uses the constant inertia weight. Two constant inertia weights, w = 0.9 and 0.4, corresponding to the initial and final values of the inertia weight of LDIW-PSO are selected for conventional PSO and the results are compared based on the MSE of the stiction estimation in (9). All PSO variants use the population of 25. Table 3 summarises the results in terms of the MSE from 50 trials. The best result for each test case is shown in bold. In 5 out of 6 stiction parameter values, LDIW-PSO performs as the best approach. Even in the case

of $S = 5$ where the LDIW-PSO does not perform the best, but its obtained MSE is just slightly smaller than the best one. So it may be concluded that overall the LDIW-PSO outperforms the constant inertia weight scheme.

Table 3: *MSE for Each Stiction Case Versus the PSO Variants.*

Case	MSE		
	Fixed w=0.4	*Fixed w=0.9*	*LDIW-PSO*
J=0	1.99E-01	5.91E-04	**1.58E-04**
S=0	1.94E-01	5.32E-04	**1.54E-04**
J=0.3	5.25E-01	1.57E-02	**1.37E-04**
S=1	1.94E-01	5.32E-04	**1.54E-04**
J=1	9.84E-04	6.70E-03	**8.06E-04**
S=5	**1.23E-02**	**1.23E-02**	1.31E-02

In the last experiment, we investigate how the proposed technique performs in detecting oscillation other than stiction. Sources of oscillation commonly present in industrial processes include incorrect controller tuning and periodic external disturbances.
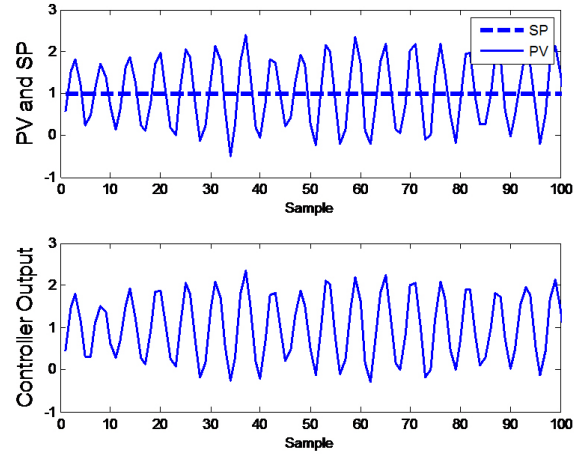

Fig.10: *Oscillatory control loop simulated with aggressive tuning without stiction for flow control loop.*

A very common scenario of inappropriate controller tuning is when an aggressively tuned controller is applied. In such scenario, the stiction quantification algorithm should be able to confirm that the cause of the oscillation is not valve stiction. We simulate this scenario by deliberately apply an aggressive tuning specification to the flow control loop by setting the integral time to 0.26 min and keeping the proportional gain to the value used in the previous simulations. The measurement noise with variance of 0.01 (SNR around 10) is injected to the measured flow to simulate corrupted data. This setting causes the loop to be oscillating as shown in Fig. 10 even when the valve is free from stiction.

To simulate the loop oscillation due to an external disturbance, a sinusoidal signal with amplitude

of 0.1 and period of 20 min is added to the measured flow. Even when the stiction is absent, the injected disturbance causes a regular oscillation to the process variable as illustrated in Fig. 11.
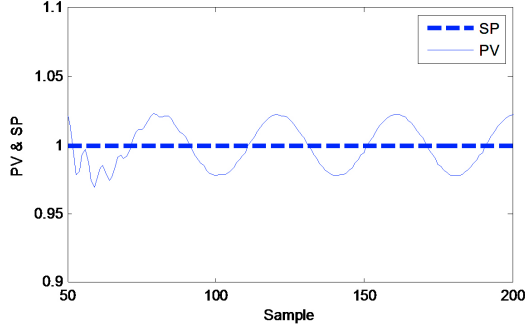

Fig.11: *Oscillatory control loop simulated with external sinusoidal disturbance without stiction for flow control loop.*

To evaluate the robustness of the proposed methodology, the LDIW-PSO is applied to different sources of oscillation for the cases of normal valve, weak stiction, and strong stiction. The values of J and S for all simulated stiction levels are similar to the ones that we used in the previous experiments.

The average estimates of the stiction parameters over 50 runs are summarised in Table 4. It can be seen that for all oscillation cases, the method successfully estimates valve stiction with high accurate quantification. Moreover, in the case of normal valve the proposed method is robust to non-stiction oscillations as the obtained values are zero for both J and S, indicating that it can quantify the stiction size correctly.

Table 4: *Average Estimates and Standard Deviations(in parentheses) of Different Stiction Levels for Different Sources of Loop Oscillation.*

Case	Actual J	Actual S	Est.J	Est.S
Normal + aggressive tuning	0	0	0(0)	0(0)
Weak stiction + aggressive tuning	0.3	1	0.3105 (0.0237)	1.0105 (0.0237)
Strong stiction + aggressive tuning	1	5	1.0045 (0.0097)	5.0045 (0.0097)
Normal + sinusoidal disturbance	0	0	0(0)	0(0)
Weak stiction + sinusoidal disturbance	0.3	1	0.2998 (0.0042)	0.9998 (0.0042)
Stroing stiction + sinusoidal	1	5	0.9676 (0.0470)	4.9675 (0.0470)

Fig. 12 shows the MV-OP plot of the prediction from the model for normal valve with aggressive tuning as an example. From the figure, we can clearly observe a linear dependence between the estimated manipulated variable and the controller output and there is no sign of any stiction pattern as it should be.
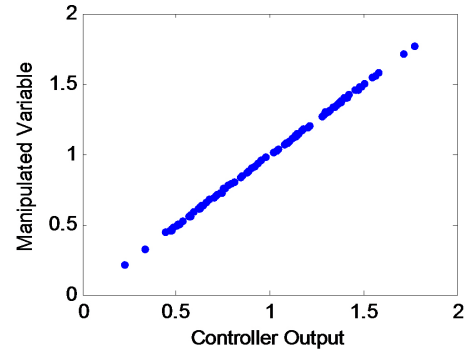

Fig.12: *MV-OP plot from the estimated stiction model of the normal valve.*

5. APPLICATION TO INDUSTRIAL DATA

The objective of this section is to evaluate the performance of the proposed method when applied to real process industrial valves. A set of data, collected from five valves and provided by a large-scale petrochemical plant in Thailand, were analysed. Valves EX1 - EX4 in Table 5 were selected from normally-operated control loops. Bear in mind that the normal valves considered in this study are not completely free from stiction. Instead, we accept a valve as normal as long as it gives a satisfactory control loop performance. For example, the plot of MV-OP of valve EX1 which is considered as normal clearly shows a non-zero deadband plus side slip. However, as this valve could still perform its control duty with excellent performance it was regarded as normal valve in this study.

Table 5: *Industrial Control Valves in This Study.*

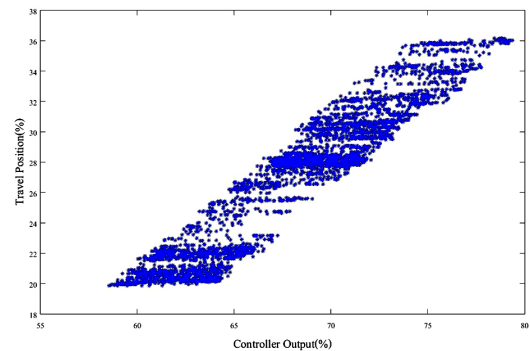| Valve	Size	Valve Type	Control Loop
EX1	1"	Globe	Tank level
EX2	1.5"	Globe	Tank level
EX3	4"	Globe	Gas flow
EX4	10"	Butterfly	Level of a distillation column
EX5	14"	Ball	Gas flow

Fig.13: *MV-OP plot of valve EX1.*

For valve EX5, it was taken from the control loop which exhibited severe oscillations. We suspected that the oscillations in this loop were due to the sticky behaviour of this valve. To confirm the condition of valve EX5, it was completely disassembled and closely examined during the maintenance period. Apparently, as seen in Fig. 14, the clear damage on the surfaces of upper and lower trunnions of the valve was visually evident. Therefore, it can be confirmed that the loop oscillations were indeed a result of stiction in valve EX5.
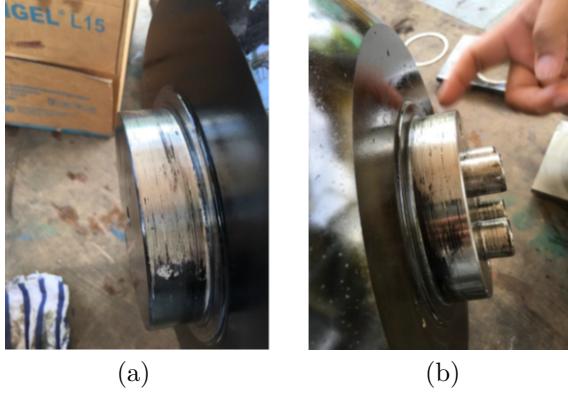

(a)   (b)

Fig.14: *Damage on the surfaces of (a) upper trunnion (b) lower trunnion of the ball valve EX5.*

For each loop, the controller output (OP) and the valve stem travel position (MV) were collected with the sampling interval of 1 second and scaled between 0 to 100%. To illustrate the ability of the proposed method to perform the online valve stiction analysis, we divided the time-series data into sub-windows with window size of 1 hour, corresponding to 3600 data points in each window. 96 samples were extracted from the normal valves (EX1-EX4) and 29 samples from the sticky valve. 68 samples of normal data and 20 samples of abnormal data were randomly selected as training data, while the remaining data were used for testing. The slip jump (J) and deadband plus slip band (S) of all samples are estimated using the proposed method. To perform the online stiction detection, the estimated J and S are compared to a set of predefined thresholds. If any of the estimated J and S is greater than its threshold, the sample is classified as faulty and the valve is stuck. Once the valve is defined as a sticky valve, its stiction level can readily be classified into weak or strong stictions by considering the magnitudes of the estimated J and S. Here, we used two sets of thresholds in order to indicate the stiction levels of the valve as described below:

$$\text{Stiction index}(i) = \begin{cases} \text{normal,if } \hat{J}_i < J_1 \ \& \ \hat{S}_i < S_1 \\ \text{weak stiction,if } J_1 \le \hat{J}_i \le J_2 \ \text{ or } \ S_1 \le \hat{S}_i \le S_2 \\ \text{strong stiction,if } \hat{J}_i > J_2 \ \text{ or } \ \hat{S}_i > S_2 \end{cases}$$

(10)

where Stiction index (i) is the verdict of valve stiction for data in window i, J_i and S_i are the estimated J and S of the data in window i, J_1 and $J_2(J_2 > J_1)$ are the thresholds of slip jump for detecting weak and strong stiction respectively, and S_1 and $S_2(S_2 > S_1)$ are the thresholds of deadband plus slip band for detecting weak and strong stiction respectively.

Obviously, the thresholds influence the robustness of this stiction detection and classification scheme. Unfortunately, from the experience of the authors, there is no single value of threshold that can guarantee to successfully detect stiction of all control loops. In this work, we obtained the thresholds empirically from the training dataset. Specifically, J_1 and S_1 were obtained from the maximum values of the estimated J and S of the normal samples in the training dataset, while J_2 and S_2 were obtained from the minimum values of the estimated J and S of the severely sticky valve in this example to represent the lower limits of strong stiction.
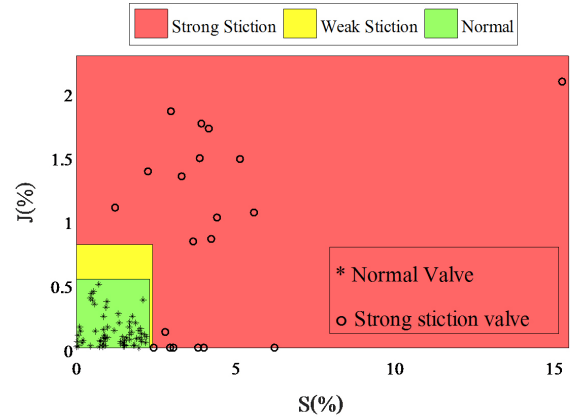

Fig.15: *Stiction condition zones and estimated J and S of the training data.*

With $\{J_1, S_1\} = \{0.53, 2.3\}$ and $\{J_2, S_2\} = \{0.8, 2.4\}$, we could plot the decision boundaries for normal, weak stiction and strong sticion as shown in Fig. 15 for the training data. To provide a convenient and quick analysis of valve stiction level, the $J - S$ plot area is divided as a green-yellow-red condition zone. The colours green, yellow, and red specify the zones for normal, weak stiction, and strong stiction. When plotting the estimates of test dataset on the $J - S$ chart in Fig. 16, we see that the stiction conditions of all samples are correctly predicted.

6. CONCLUSIONS

In this paper, a method for quantification of valve stiction has been proposed. The slip-jump J and the deadband plus stickband S could be simultaneously estimated. Stiction parameters were identified accurately through Kano model through MV-OP plot
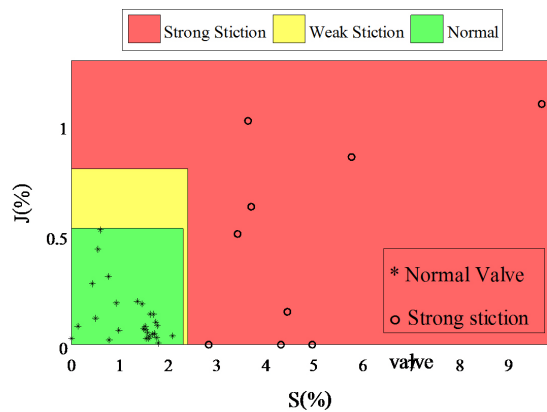
Fig.16: *Stiction condition zones and estimated J and S of the test data.*

with the use of particle swarm optimisation. The linear decrease inertia weight was suggested to be used during the update process of PSO to improve the identification convergence and accuracy. The experimental results clearly show the superiority of the LDIW-PSO over the fixed inertia weight PSO. The comparisons are made in terms of solution accuracy. In addition, the robustness of the proposed method to incorrect controller tuning and external disturbances was demonstrated. The simulations have shown that the estimated parameters obtained from this method can clearly indicate if the oscillation is caused by valve stiction or other problem. Industrial examples have demonstrated the implementation of this technique in an online fashion and confirmed the effectiveness of the proposed stiction quantification method in practice.

References

[1] M.A.A. S. Choudhury, M. Jain and S. L. Shah, "Stiction - Definition, Modelling, Detection and Quantification," *Journal of Process Control*, Vol. 18, No. 3-4, pp. 232-243, March 2008.

[2] M. Jelali, *Control Performance Management in Industrial Automation*, 1st ed., UK: Springer-Verlag London, 2013, pp. 266.

[3] M. Jelali and B. Huang, *Detection and Diagnosis of Stiction in Control Loops: State of the Art and Advanced Methods*, 1st ed., UK:Spinger-Verlag London, 2010.

[4] A. Stenman, F. Gustafsson, and K. Forsman, "A Segmentation-based Method for Detection of Stiction in Control Valves," *International Journal of Adaptive Control and Signal Processing*, Vol. 17, No. 7-9, pp. 625-634, September - November 2003.

[5] M. Kano, H. Maruta, H. Kugemoto, and K. Shimizu, "Practical Model and Detection Al-

gorithm for Valve Stiction," *Proceedings of 7th IFAC DYCOPS*, Cambridge, USA, 2004.

[6] Q. P. He, J. Wang, M. Pottmann, and S. J. Qin, "A Curve Fitting Method for Detecting Valve Stiction in Oscillating Control Loop," *Industrial & Engineering Chemistry Research*, Vol. 46, No. 13, pp. 4549-4560, 2007.

[7] M.A.A.S. Choudhury, S.L. Shah, N.F. Thornhill, "Diagnosis of Poor Control Loop Performance using Higher Order Statistics," *Automatica*, Vol. 40, No. 10, pp.1719-1728, 2004.

[8] D. Karnopp, "Computer Simulation of Stick-Slip Friction in Mechanical Dynamic Systems," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 107, No. 1, pp. 100-103, 1985.

[9] C. C. de Wit, H. Olsson, K. J. Åström, and P. Lischinsky, "A New Model for Control of Systems with Friction," *IEEE Transactions of Automatic Control*, Vol. 40, No 3, pp. 419-425, 1995.

[10] M. A. A. S. Choudhury, N. F. Thornhill, and S. L. Shah, "A Data- Driven Model for Valve Stiction," *Proceedings of 5th ADCHEM*, Hong Kong, China, 2004, pp. 261-266.

[11] C. Garcia, "Comparison of Friction Models Applied to a Control Valve," *Control Engineering Practice*, Vol. 16, No. 10, pp. 1231-1243, October 2008.

[12] S. Sivagamasundari and D. Sivakumar, "Estimation of Valve Stiction Using Particle Swarm Optimization," *Sensors & Transducers Journal*, Vol. 129, pp. 149-162, 2011.

[13] A. Nikabadi and M. Ebadzadeh , "Particle Swarm Optimization Algorithms with Adaptive Inertia Weight : A Survey of the State of the Art and a Novel Method," *IEEE Journal of Evolutionalry Computation*, 2008.

[14] Y. Feng, G. F. Teng, A. X. Wang, and Y. M. Yao, "Chaotic Inertia Weight in Particle Swarm Optimization," *Proceedings of 2nd International Conference of Innovative Computing, Information and Control*, Kumamoto, 2007, p. 475.

[15] J. Xin, G. Chen, and Y. Hai., "A Particle Swarm Optimizer with Multistage Linearly-Decreasing Inertia Weight," *Proceedings of International Joint Conference on Computational Sciences and Optimization*, Sanya, Hainan, 2009, pp. 505-508.

[16] Y. Shi and R. C. Eberhart, "Empirical Study of Particle Swarm Optimization," *Proceeding of the 1999 Congress on Evolutionalry Computation*, Washington, DC, 2002, pp. 1945-1950.

[17] J. C. Bansal, P. K. Singh, M. Saraswat, A. Verma, S. S. Jadon, and A. Abraham, "Inertia Weight Strategies in Particle Swarm Optimization," *Proceedings of 3rd World Congress on Nature and Biologically Inspired Computing*, Salamanca, 2011, pp.640-647.

Pongsurachat Aksornsri received the BEng degree in Electrical Engineering from Chiangmai Universtiy, 2005. He graduated with a master degree in Control Systems and Instrumentation Engineering from King Monkut's University of Technology Thonburi, Bangkok, Thailand in 2015. His research interests include process monitoring for industrial applications.

Sarawan Wongsa received the BEng degree in Control Systems and Instrumentation from King Monkut's University of Technology Thonburi, Bangkok, Thailand in 1998, and the MSc and PhD degrees in Automatic Control and Systems Engineering from Sheffield University, United Kingdom, in 2002 and 2007, respectively. She is currently an assistant professor in the Department of Control Systems and Instrumentation Engineering at King Monkut's University of Technology Thonburi. Her research interests include intelligent fault detection and diagnosis.