LoopRank: A Novel Tool to Evaluate Loop Connectivity

M. Farenzena and J. O. Trierweiler

Group of Integration, Modelling, Simulation, Control and Optimization of Processes (GIMSCOP)
Department of Chemical Engineering, Federal University of Rio Grande do Sul (UFRGS)
Rua Luiz Englert, s/n CEP: 90.040-040 - Porto Alegre - RS - BRAZIL,
Fax: +55 51 3308 3277, Phone: +55 51 3308 4163
E-MAIL: {farenz, jorge}@enq.ufrgs.br

Abstract: Since the number of loops in refineries or petrochemical plants is very large and the number of loops with poor performance is equally large, to prioritize their maintenance is essential to ensure plant profitability. This work proposes a methodology called *LoopRank* to compute the importance factor of each loop, aiming to prioritize their maintenance. The algorithm is based on the connection among them, which is computed using partial correlation. The algorithm is based on *PageRank*, which analyses connections among nodes recursively and computes a rank for each node using partial correlation. The *LoopRank* assigns an individual score for each loop ranging from 0% to 100%. Based on this score, the loop maintenance can be ranked. The *LoopRank* algorithm is computationally efficient, thus allowing its industrial large-scale application. The proposed algorithm was applied both on simulation and industrial case studies, providing fruitful results.

Keywords: Performance Monitoring, Data correlation, Loop Rank, Partial Correlation, Data-mining.

1. INTRODUCTION

Nowadays, it is a common knowledge the positive impact of control loop performance assessment tools over industrial plants. In the last twenty years, many methodologies and tools have been developed to diagnose the main loops problems:

- Poor performance (Harris, 1989, Huang *et al.*, 1997, Jelali, 2006);
- Plant-wide disturbances (Jiang *et al.*, 2007, Thornhill and Horch, 2007, Xia *et al.*, 2005);
- Valve hysteresis (Choudhury *et al.*, 2004, Hagglund, 2002, Hagglund, 2007, Rossi and Scali, 2005, Ruel, 2000);

It is also well known that most of industrial loops do not perform well (Paulonis and Cox, 2003). However, improve and maintain all loops in their optimal performance are impossible and economically infeasible because of the small number of engineers responsible to maintain a large number of loops. Therefore, a methodology to prioritize loop maintenance is required.

Methodologies to prioritize loop maintenance or to evaluate loop interaction are scarce in the literature. Tangirala et al. (2005) proposed a method based on spectral correlation between loops. Thornhill et al. (2002) proposed tools based on spectral principal component analysis.

The scope of this work is to provide an importance score for each control loop to prioritize its maintenance. Fig. 1 shows one simple case with four loops.

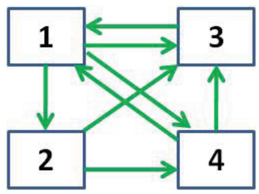

Fig. 1: Case study with four control loops interconnected.

In Fig. 1 scheme, it is easy to see that loop 3 has the most connections from others (3). So, is it the most important loop? On the other hand, loop 1 receives 2 connections, where one of them is very important, coming from loop 3, and it is the only loop 3 connection. Which is more important, loop 1 or 3? It is clear that an algorithm to systematize this procedure that provides an importance score for each loop is strongly required.

In this work is proposed an algorithm, called *LoopRank* that provides a grade based on loops connections. The loops that receive more connections from others, i.e. the loops that have stronger correlation with the remaining should have more importance than a loop that does not have any correlation with the others. To quantify these bounds, partial correlation is used. Subsequently, the priority of each loop is ranked using *PageRank* algorithm (Bryan and Leise, 2006).

The paper is segmented as follows: in section 2 the necessary background will be summarized. In section 3, the methodology to prioritize loop maintenance, proposed in this work is described. In section 4, the methodology is applied in simulation and industrial case studies. The paper ends with the concluding remarks.

2. BACKGROUND

This section provides the necessary background to understand the proposed methodology described in section 3.

2.1 Correlation and Partial Correlation

Correlation can be described as the linear dependence between two random variables (Bilodeau and Brenner, 1999). The correlation (ρ_{XY}) between two variables can be computed as follows:

$$\rho_{XY} = \frac{cov(X,Y)}{\sqrt{var(X)var(Y)}} \tag{1}$$

Where $cov(X,Y)$ is the covariance between X and Y and $var\,var$ is the variance. The correlation is a measurement of variable interaction, independently of the scale which it is measured.

In the case where the inputs and outputs are correlated, a better measure of the interaction is the partial correlation. It provides the degree of association of X and Y, with the effect of a set controlling variables (Z) removed. The partial correlation between X and Y with Z fixed ($\rho_{XY|Z}$) is computed by:

$$\rho_{XY|Z} = \frac{\rho_{XY} - \rho_{XZ}\rho_{YZ}}{\sqrt{(1-\rho_{XZ}^2)(1-\rho_{YZ}^2)}} \tag{2}$$

2.2 Importance score

To rank the relative importance of elements is essential when resources are limited. Rank algorithms have a broad class of applications including financial decisions, searching tools, among others. One rank algorithm that has been highlighted recently is *PageRank* (Bryan and Leise, 2006), which is used by the Google®'s search engine to rank pages relevance. It gives an importance score for each webpage according to an eigenvector of a weighted link matrix. It is based on the links made to a given page from other pages, and the relative impact of each source page.

The algorithm can be summarized as follows. Suppose n elements where the relative connectivity of them (x_k) should be computed, where k is the indexing element ($1 \le k \le n$), where this value corresponds to the arrows in each element. In the example (Fig. 1) $x_1=2$, $x_2=1$, $x_3=3$, $x_4=2$. Thus, loops can be ranked as follows 3, 1 and 4, and 2, based only on the connections. Following, the relative importance of k (x_k) is computed using the number of back links for this page. If page j contains n_j links to other pages, and one of them links to element k, then it will be boosted by a score x_j/n_j. Let $L_k \subset \{1,2,\dots,n\}$ denote the set of pages with link to page k. The relative weight for each k is computed by:

$$x_k = \sum_{j \in L_k} \frac{x_j}{n_j} \tag{3}$$

It is also assumed that the link from a page to itself has zero weight. For loop 1, its impact can be written as $x_1 = \frac{x_3}{1} + \frac{x_4}{2}$, since pages 3 and 4 have back-links to 1 and loops 3 and 4 have 1 and 2 links, respectively.

This linear relation can be written as $Ax = x$, where A is called "link matrix" and $x = [x_1 \quad x_2 \quad x_3 \quad x_4]$. $A(i,j)$ provides the relative weight from loop j to loop i, where the rows show the relative weight of each connection that goes to a given loop and columns show the relative importance of the connections that come from the same loop.

In the scheme of Fig. 1, the A matrix can be written as:

$$A = \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix} \tag{4}$$

In the case of loop 1 (Fig 1): three arrows come out this loop. Then the relative importance added in each loop is $\frac{1}{3}$, as shown in the first column of A.

This procedure transforms the problem into a simple eigenvector problem ($Ax = \lambda x$). It can be proved that λ has always a unitary eigenvalue for this kind of matrices. Thus, the eigenvector x with eigenvalue 1 for matrix A is seek. The importance score (*IS*) for each page is given by the mentioned eigenvector just normalizing each elements by the sum of all components so that at the end the final sum is equal to 1. For the case study, the already normalized importance score are $x_1=0.387$, $x_2=0.129$, $x_3=0.290$, $x_4=0.194$.

More information about *PageRank* algorithm can be found in Bryan and Leise (2006).

3. METHODOLOGY DESCRIPTION

This section describes the *LoopRank* algorithm to evaluate loop importance.

Initially the loop output data is collected. Only routine data is required and no further information about the loop is required.

The first step is to compute the links between loops and its relative weight, where the impact of a single variable over each other should be computed. The measure of loops connectivity used in this work is the linear dependence among them.

Industrial loops generally have high correlation between them. To overcome this constraint and isolate the individual loop impact over each other, partial correlation is used. Simple correlation between loops has also been tested and results were poorer. This comparison will be shown in the case studies section. Thus, the relative weight between loops i

and j is provided by the partial correlation between these loops, removing the effect of the remaining loops ($\rho_{ij|LI}$), where $L = \{1,2,\ldots,n\}$, $LA = \{i,j\}$, and $LI = L \cap LA$. Each element of the relative weight matrix (A_{ij}) is given by the partial correlation between loop i and loop $_j$ ($\rho_{ij|LI}$) :

$$A_{ij} = \rho_{ij|LI} \tag{5}$$

The next step is to evaluate *the LoopRank (LR)*, based on *relative weight matrix* (*A*), using *PageRank* algorithm that can quantify the relative importance of each loop, allowing to rank the loops for maintenance purposes. This class of algorithm was chosen because of its capacity to prioritize elements based on the connections among them and its computational/numerical efficiency. The *LoopRank* output is then normalized to limit each grade between 0 and 100%, where always the worst important loop has $LR = 0\%$ and the most important $LR = 100\%$.

Some loops can have more impact in plant profitability or help to smooth the operation. The loops that have connections with these "important ones" should have stronger weights. Thus, it is necessary to assign a *loop weight* (w_k), which is dependent on the source loop. The w_k is assigned heuristically, depending on loop type and its profitability. One heuristics is here suggested: flow and level loops are least important ($w=1$), pressure loops have middle importance ($w=1.5$) and temperature and composition are the highest importance ($w=2$). The *connection weight* k is then multiplied by all links where the source is loop k, in A.

The application of *LoopRank* algorithm can be summarized as follows:

1. Collect routine operating data of the loops;
2. Compute the partial correlation between each loop and build the *relative weight matrix (A)*;
3. Based on *A*, compute the *LoopRank*, using *PageRank* algorithm;
4. To the result in 3, multiply each loop by the corresponding loop weight (*w*);
5. Normalize the final results to express the result in relative percentage, where each importance is bounded between 0% and 100%.

4. CASE STUDIES

This section shows the application of *LoopRank* algorithm both on simulation and industrial case studies.

4.1 Simulation example I

In the first case study, a set of 10 loops will be analyzed. The first one has oscillatory behaviour. The other loops have been generated from the first loop by a simple addition of different noise levels followed by normalization in the amplitude. Fig. 2 shows the time trends for all loops.

Applying the proposed algorithm with $w = 1$ for all loops produces the results shown in Tab. 1. As it was already expected, the results of Tab. 1 clearly indicates that the source of oscillatory behaviour is related to the most important loop, which is in this case is the first one.

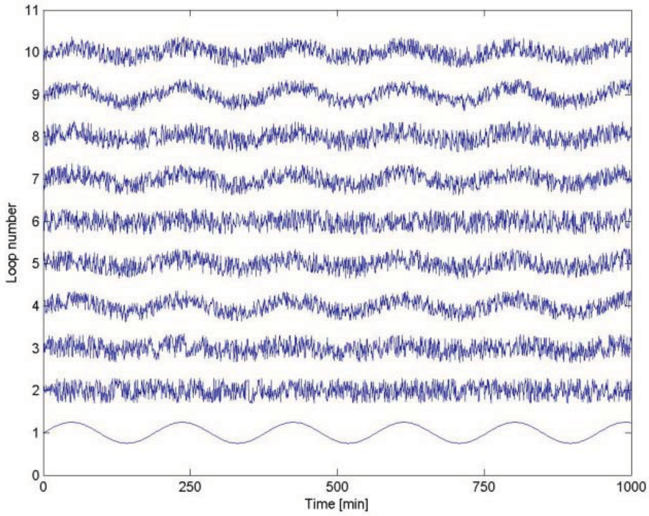

Fig. 2: Time trends of 10 data series.

Tab. 1: *LoopRanks* for case study 1.

k	LR_k	k	LR_k
1	100	**6**	16
2	3	**7**	10
3	1	**8**	20
4	21	**9**	11
5	13	**10**	0

Following, the impact of w_k will be analyzed. In this case study, w_5 will be increased and its impact over loop 4 (LR_4) and loop 8 (LR_8) are shown in Tab. 2. Since interaction between loop 5 and 1 is stronger than all remaining loop 5 connections, it is expected that increase the weight of loop 5, the *LR* of loop 1 will increase while the *LR* of all others will decrease, because the connections between loop 5 and all others will become weaker.

Tab. 2: Impact of w_5 in LR_4 and LR_8 *LoopRanks.*

w_5	LR_4	LR_8
1	21.0	20.0
1.5	20.0	18.6
2	19.3	17.3
5	16.1	12.3
10	13.7	8.3

The previous claim is corroborated by Tab. 2, where increasing w_5, the importance factors LR_4 and LR_8 decreased. LR_1 remains for all cases equal to 100%.

4.2 Simulation example II

In the second case study a set of 100 loops are be analyzed. The time trend for each loop is generated using the following procedure:

- Loop 1, loop 2, and loop 4 have oscillatory behaviour with different frequencies;
- Loop 3 and loop 5 data trends are obtained passing white-noise through a first order transfer function with different time constants;
- Loops 6, 7, and 8 are random signals;
- Other 92 data trends are generated by the linear combination of the first 8 data trends. Following, white noise is added in each one of the 92 data trends. Time trends 1, 2, and 3 impact all 92 loops using a random weight between 0.5 and 1.
- Time trends 4 to 8 impact some of 92 loops using a random weight between 0.5 and 1. The probability of each time trend to impact each loop is 50%.

Fig. 3 shows the 8 time trends for the source loops and Fig. 4 shows loops 9 to loop 18 time trends, generated using the previous loops.

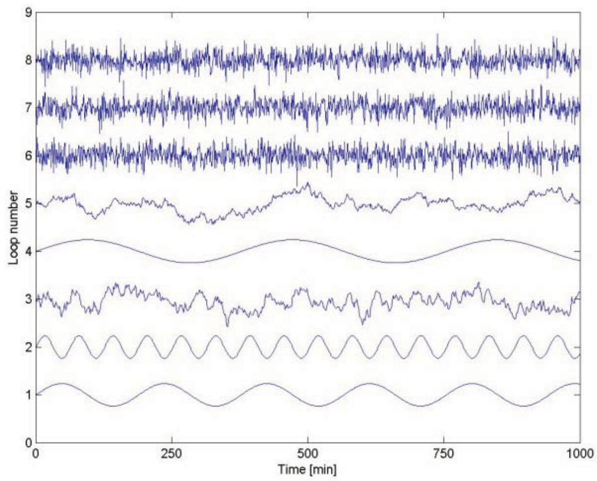

Fig. 3: Time trends for loops 1 to 8 in case study 2.

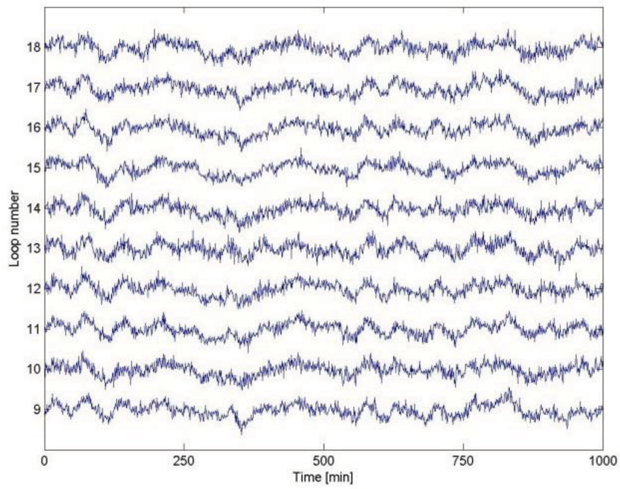

Fig. 4: Time trends for loops 9 to 18 in case study 2.

Applying the *LoopRank* algorithm, the following importance, shown in Fig. 5, is computed:

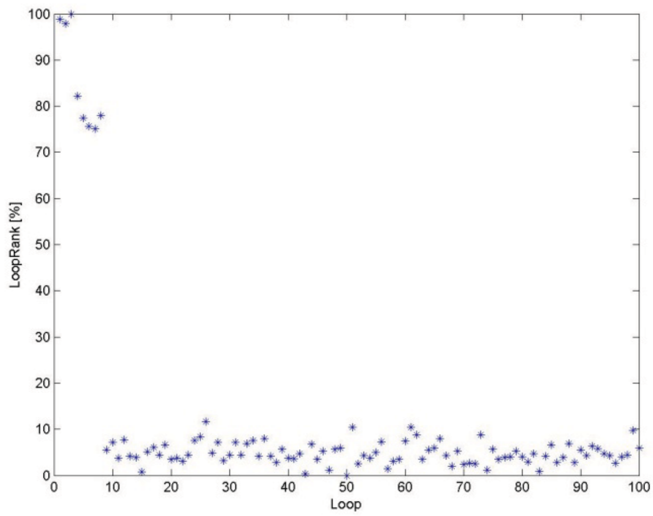

Fig. 5: *LoopRank* for case study 2, using 100 loops.

Fig. 5 reflects the expected result – loops 1, 2, and 3 have the highest importance, because of their impact in all loops. Loops 4 to 8 are less important than loops 1 to 3, but they are more important than the remaining. The remaining loops are less important, because the impact of a single one is not transferred to others.

One question can arise: If instead of partial correlation the correlation would be used, the results would be different? The comparison between *LoopRanks* using partial correlation and correlation is shown in Fig. 6.

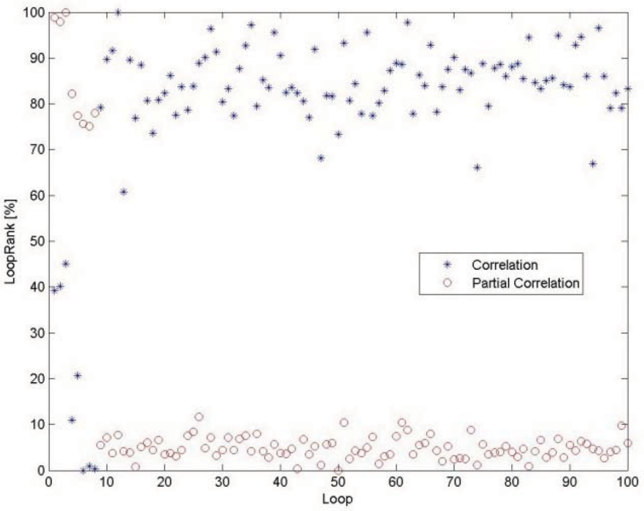

Fig. 6: *LoopRank* for case study 2, using both correlation and partial correlation.

Fig. 6 shows contrasting results, while correlation shows small importance of loops 1-8, partial correlation showed that these loops are the most important. Similar results have been seen in all tests, where correlation cannot point out the loops with major interaction among them, reason why partial correlation is used.

4.3 Industrial data

An industrial data set was provided by the courtesy of a Brazilian refinery. The system is an atmospheric distillation column of a petroleum refinery. The provided data set consists of 25 process variables: 6 level, 12 flow, 5 pressure, and 2 temperature controllers. The whole dataset has 1000 samples with a sampling time of 1 min. Fig. 7 shows the time trends of the variables.

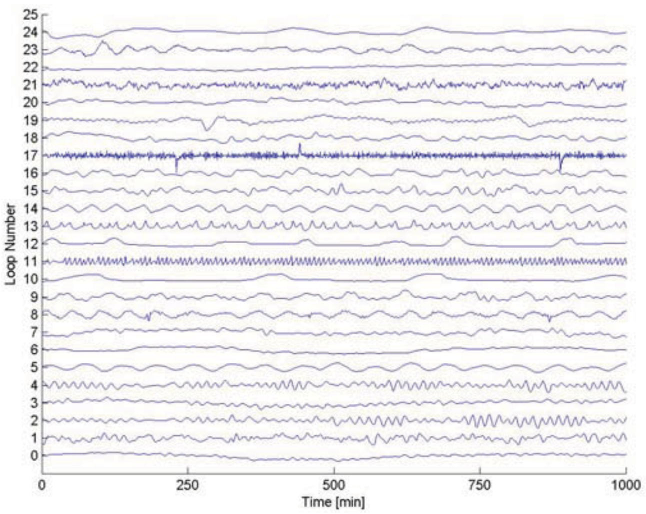

Fig. 7: *Time trends of industrial case study with 19 process variables.*

The *LoopRank* algorithm is then applied, providing the ranking shown in Fig. 8.

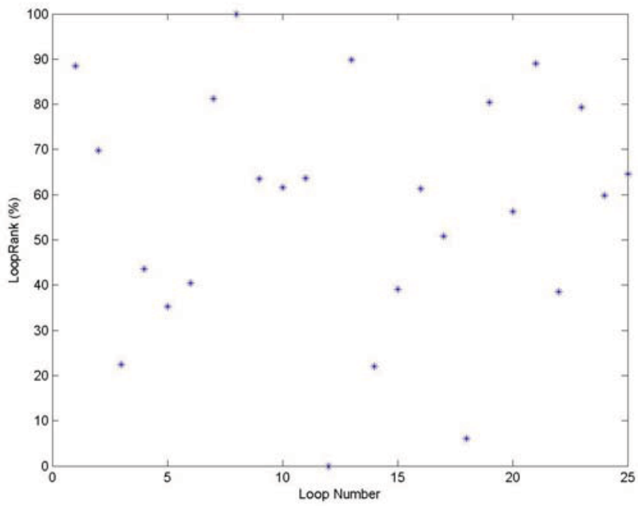

Fig. 8: *LoopRank* for industrial case study.

Fig. 8 clearly ranks the loop importance. It shows that loops 1, 8, 13, and 21 are the most important, because of its impact over the others. Loops 3, 12, 14, and 18 are the least important.

There previous results can be explained by the positions in the process flow diagram. The most important loop, in this case, is the flow of the intermediate recycle in the atmospheric tower (loop 8). The remaining most important loops are:

- Loop 1: crude oil inlet flow;
- Loop 13: Total reflux flow;
- Loop 21:Kerosene flow side-withdraw.

5. CONCLUSIONS

The main conclusions of the proposed work can be summarized as:

Loop ranking is an important tool for loop maintenance – Loop ranking for maintenance is required because of the large number of loops with poor performance in process plants. Unfortunately, the number of methodologies to evaluate loop impact is scarce. In this work a methodology for loop ranking aiming their maintenances, called *LoopRank*, is proposed.

It is better use partial correlation than correlation – *LoopRank* is based on loop interaction, measured by partial correlation. Correlation was also tested, however the results were poorer, therefore should not be used.

The proposed algorithm is similar to the Pagerank algorithm used by Google search engine – the relative importance score for each loop is computed using the *PageRank* algorithm. When the impact of a given loop should be emphasized, a loop weight can be assigned.

Successful applications of the LoopRank algorithm – the proposed algorithm was applied in 3 case studies, where reliable results were provided. One industrial case study was presented to demonstrate the efficacy of the algorithm. The computational time for all case studies was negligible.

ACKNOWLEDGMENTS

The authors would like to thank CAPES / Brazil for supporting this work.

REFERENCES

Bilodeau, M. and Brenner, D. (1999) *Theory of multivariate statistics,* New York, Springer.

Bryan, K. and Leise, T. (2006) The $25,000,000,000 eigenvector: The linear algebra behind Google. *SIAM Review,* 48**,** 569-581.

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**,** 1719-1728.

Hagglund, T. (2002) A friction compensator for pneumatic control valves. *Journal of Process Control,* 12**,** 897-904.

Hagglund, T. (2007) Automatic on-line estimation of backlash in control loops. *Journal of Process Control,* 17**,** 489-499.

Harris, T. J. (1989) Assessment of control loop performance. *Canadian Journal of Chemical Engineering,* 67**,** 856-861.

Huang, B., Shah, S. L. and Kwok, E. K. (1997) Good, bad or optimal? Performance assessment of multivariable processes. *Automatica,* 33**,** 1175-1182.

Jelali, M. (2006) An overview of control performance assessment technology and industrial applications. *Control Engineering Practice,* 14**,** 441-466.

Jiang, H., Shoukat Choudhury, M. A. A. and Shah, S. L. (2007) Detection and diagnosis of plant-wide oscillations from industrial data using the spectral envelope method. *Journal of Process Control,* 17**,** 143-155.

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

Rossi, M. and Scali, C. (2005) A comparison of techniques for automatic detection of stiction: simulation and application to industrial data. *Journal of Process Control,* 15**,** 505-514.

Ruel, M. (2000) Stiction: The hidden menace. *Control Magazine*.

Tangirala, A. K., Shah, S. L. and Thornhill, N. F. (2005) PSCMAP: A new tool for plant-wide oscillation detection. *Journal of Process Control,* 15**,** 931-941.

Thornhill, N. F. and Horch, A. (2007) Advances and new directions in plant-wide disturbance detection and diagnosis. *Control Engineering Practice,* 15**,** 1196-1206.

Thornhill, N. F., Shah, S. L., Huang, B. and Vishnubhotla, A. (2002) Spectral principal component analysis of dynamic process data. *Control Engineering Practice,* 10**,** 833-846.

Xia, C., Howell, J. and Thornhill, N. F. (2005) Detecting and isolating multiple plant-wide oscillations via spectral independent component analysis. *Automatica,* 41**,** 2067-2075.