Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

An Improved MILP Method for Data Rectifications with Gross Error Candidates

Jianlie Li, Gang Rong*, Xu Wang, Yiping Feng

**State Key Laboratory of Industrial Control Technology,*
Institute of Advanced Process Control, Zhejiang University, Hangzhou, 310027, P. R. China
(Tel: +86-571-86667032; e-mail:grong@mail.hz.zj..cn).

Abstract: MILP (Mixed Integer Linear Programming) method for simultaneous gross error detection and data reconciliation has been proved to be an efficient way to adjust process data with material and other balance constraints. But the efficiency will decrease significantly when the MILP method is applied in a large-scale data rectification problem because there are too many binary variables to be considered. In this paper, a strategy is proposed to generate a list of gross error candidates with reliability factors. The list of candidates are combined into the MILP objective function to improve the efficiency and accuracy through reducing the number of binary variables and giving accurate weights for suspected gross errors. Industrial examples are provided to show the efficiency of the algorithm.

1. INTRODUCTION

Control and optimization of an industrial process ultimately depend on the accuracy and reliability of process data. However, in most cases, process variables are corrupted during the measurement, processing, and transmission of the measured signals. Sometimes, errors in measured data can lead to significant deterioration in evaluating performance of a plant. After the idea of data reconciliation was brought in 1961(Kuehn and Davidson, 1961), the problem of improving the accuracy of process data to let them satisfy the material, component and energy balance with less adjustment and shorter time has been studied for decades. Besides the widely used solution of linear and nonlinear problem using matrix projection (Crowe, 1986, Crowe et al., 1983), many other methods such as MILP method(Soderstrom et al., 2001), PCA method(Tong and Crowe, 1995) ,MTNT method (Wang et al., 2004, Yang et al., 1995, Mei et al., 2006) and redundancy analysis method(Zhang et al., 2001) have been developed to solve the steady-state data rectification problem. More information about the history of data rectification methods can be obtained from a perspective written by Crowe (Crowe, 1996).

MILP framework of simultaneous data reconciliation and gross error detection is prompted to remove random errors of plant data as well as identify and compensate gross errors for the final solution. MILP method defines an associated binary variable for each measurement to indicate the existence of gross error and add penalty in the objective function to activate the binary variables. However, the efficiency will decrease significantly when the method is applied in the large-scale problems, as there will be too many binary variables in the calculation. Fortunately, we found that before performing the mixed integer linear optimization, the number of binary variables can be reduced on the basis of historic data. Moreover, the fit weighting factors in the objective function should also improve the results.

This article describes a method for searching gross error candidates in a directed diagram of process flowsheet and calculating their reliability factors. The gross error candidates are used to decide binary variables in the MILP objective function. This paper is organized as follows. In section 2, MILP method is briefly introduced. Then, the algorithm of gross error candidate generation based on graphic theory and bayesian method is described in section 3 and section 4, and the usefulness of the proposed algorithm is shown with some industrial examples in section 5. Conclusions and future research topics are addressed in section 6.

2. BASIC MILP METHOD

For a linear, time invariant and steady state system such as a network of flowrate, balance equation can be written as:

$$\sum_{i=1}^{s} a_{ki} F_i = 0 \qquad k = 1.....n \qquad (1)$$

Where n is the number of process nodes, s is the total number of streams. F_i is the true flow rates of stream i. a_{ki} is the correlative parameter whose value is 1, -1 or 0 depending on the stream i is enters, leaves or not correlate to node k.

If we consider measured value $\widetilde{F}_i$ consists with true value F_i, random component ε_i with zero mean and a know variance σ_i, and a deterministic bias δ_i then

$$\widetilde{F}_i = F_i + \varepsilon_i + \delta_i \qquad (2)$$

The MILP method is derived from the standard formulation of data reconciliation:

$$\begin{aligned} &\min \Phi = (\hat{F} - \tilde{F})^T \cdot Q \cdot (\hat{F} - \tilde{F}) \\ &s.t \qquad \mathbf{A} \cdot \hat{F} = 0 \end{aligned} \tag{3}$$

By substituting the equation (2) into (3) and define each measurement a binary variable to indicate the existence of a bias, the new objective function with MILP formulation can be written as:

$$\begin{aligned} &\sum_{i=1}^{n} \frac{1}{\sigma_i} \left| \hat{F}_i - (\tilde{F}_i - \delta_i) \right| + W_i B_i \\ &s.t \quad \mathbf{A} \cdot \hat{F} = 0 \\ &\qquad \zeta U_i B_i \le \left| \delta_i \right| \le U_i B_i \\ &\qquad B_i \in binary \end{aligned} \tag{4}$$

Here the binary variable B_i represents the existence of a bias in ith measurement, and W_i is the weighting factor to represent the importance of biased measurement, the second constraint ensures the upper limit of a bias and the activate zone of binary variables by choosing ζU_i as some fraction of the standard deviation of the measurement. In addition, equation (4) can be restated to remove the absolute operator to avoid discontinuity problem.

3. GENERATE GROSS ERROR CANDIDATES

Because the data reconciliation and gross error detection problems are formulated as MILP frameworks, it is easy to apply other techniques to enhance the performance. Soderstrom(2001) suggests using some standard statistical tests for bias as constrains, however, these modifications not offer much benefits over the basic method. And moreover, as an optimization problem, this technique is significantly more computationally intensive. Generate a list of gross error candidates and reduce the number of binary variables in the objective function is obviously an efficient way to improve the efficiency of the MILP method.

An efficient method based on graph theory and Bayesian method is prompted for generate gross error candidates with weighting factors and reduce the binary variables in the objective function. This method can generate gross error candidates with prior information and need less computation time.

In 1988, Tamhane, Jordache and Mah(Tamhane et al., 1988a, Tamhane et al., 1988b) used Bayesian approach to detect gross error, in which the prior probabilities of gross error occurrence are updating in the light of accumulating data. And the probability of gross error occurrence for instrument S_i is:

$$P_i(t) = 1 - \frac{\Gamma(l_i + m_i)\Gamma(m_i + \tau_i(t))}{\Gamma(l_i)\Gamma(l_i + m_i + \tau_i(t))} \qquad i = 1, 2, ..., n \tag{5}$$

Where l_i is the number of previous failure for instrument S_i and m_i is the sum of previous lifetimes for instrument S_i and $\Gamma(\cdot)$ denotes the gamma function.

Once the prior probabilities are confirmed, the next step is to use these probabilities to generate gross error candidates before using the MILP method. This step can be modified or even removed when there are expert experiences or no accumulated process data.

In the view of graph theory, the flowsheet can be regard as a directed graph. As illustrated in figure 1, spanning tree is the subgraph of a graph and this subgraph doesn't contain any loop. So the streams in the flowsheet can be divided into two sets: branches and chords of the spanning tree. Obviously, the true value of every branch can be obtained through true values of some chords. For an example, the branches of spanning tree No.1 in Fig1 is stream 2, 3 and 4, so the chords will be stream 1 and 5. So the true value of stream 2 and stream 4 is as same as the true value of stream 1, and true value of stream 3 is identical to the sum of true values of stream 1 and stream 5. In this case, we called these chords as independent variables (IV) and branches as dependent variables (DV).

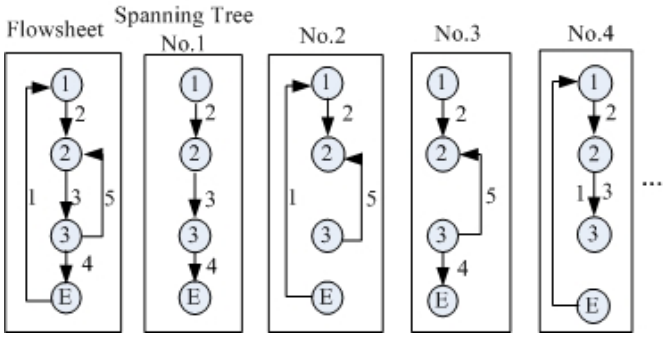

Figure 1 Spanning trees for a simple flowsheet

Usually, there are a large number of spanning trees in a given graph, so the choice of independent variables is not exclusive. In this method, because the independent variables are used as benchmarks in the gross error candidate generation, we choose the chords of graph's maximal spanning tree as independent variables. And the weight of the given flowsheet is defined as prior probabilities calculated by (5).

Because the maximal spanning tree is the spanning tree with the largest weight, the sum of the weights of all chords is minimal, which means these independent variables have lowest probabilities to have gross error. These independent variables will be used as benchmarks in the gross error generation. If a measured value doesn't contain any gross error, $\widetilde{F}_i$ will only consists with true value F_i and random component ε_i with zero mean and a know variance σ_i. So the sum of measured values is

$$\begin{aligned} &\sum_{i=1}^{k} \widetilde{F}_i = \sum_{i=1}^{k} F_i + \varepsilon \\ &\varepsilon \sim N(0, \sum_{i=1}^{k} \sigma_i^2) \end{aligned} \tag{6}$$

Because there are random component ε_i in the $\widetilde{F}_i$, the measured value of dependent variables only can be estimated from measured value of independent variables. If we still choose chords of spanning tree No.1 in Fig 1 as independent variables, the estimated value of stream 2 and stream 4 is the

measured value of stream 1 and the estimated value of stream 3 is the sum of measured values of stream 1 and stream 5. So the relationship between estimated value $\bar{F}_{IV}$ and measured value $\tilde{F}_{IV}$ can be calculated as:

$$\bar{F}_{IV} - \tilde{F}_{IV} = \sum_{DV \subset R} F_{DV} - F_{IV} + \sum_{DV \subset R} \varepsilon_{DV} - \varepsilon_{IV} = \sum_{DV \subset R} \varepsilon_{DV} - \varepsilon_{IV} \quad (7)$$

where R is the set of independent variables used to estimate the dependent variable. And $\varepsilon_i \sim N(0, \sigma_i^2)$ leads to

$$Z = \sum_{DV \in R} \varepsilon_{DV} - \varepsilon_{IV} \sim N(0, \sum_{DV \in R} \sigma_{DV}^2 + \sigma_{IV}^2)$$

Statistic Z is established to indicate the measurement bias for dependent variables that has an upper threshold limit of Z_α for a level of significance α.

$$|Z| \le Z_\alpha \quad (8)$$

There is a remaining problem that independent variables have certain probabilities to include gross error, and cause the Type I Error of gross error candidates in the dependent variables. In order to solve this problem, if an independent variable contains gross error, we assume that all the dependent variables corresponding to this independent variable will be suspected. The probability of gross error occurrence between this independent variable and all the corresponding dependent variables will be compared to decide the gross error candidate. Assume that the probability of gross error occurrence is 0.2 for streams 2, 3, 4 and 0.1 for stream 1 in Fig 1. If streams 2, 3 and 4, which are corresponding to stream 1, are suspected with gross errors, stream 1 will be added into gross error candidate list because the probability of all stream 2, 3 and 4 have gross error will be 0.2^3 which is smaller than the probability of gross error occurrence for stream 1. Once there are candidates in the independent variables, all dependent variables not in the candidate list will be removed from graph and new iteration will be performed in the subgraph till there are no new gross error candidates in the subgraph. Details of the gross error candidate generation method are described as follows:

Step1. Weight all measured streams with their probabilities of gross error occurrence calculated by (5).

Step2. Finding the maximal spanning tree of the graph, classify the streams into independent and dependent variables.

Step3. Establish statistic Z for each dependent variable according to (7), put the dependent variable into suspected set if $|Z| \ge Z_\alpha$.

Step4. Check if there are any independent variables that all the corresponding dependent variables are in suspected set. If there is such independent variable IV and $P_{IV} \ge \prod_{DV \in R} P_{DV}$, raise the P_{IV} to 1 and put this independent variable into suspected set S then go to Step5, else stop and output suspected set S.

Step5. Merge all dependent variables not belong in the suspected set into the nearest node that belong to any chords or streams in S to generate a subgraph then go to Step2.

4. USE CANDIDATES IN THE MILP METHOD

Gross error candidates generated in the last section will be introduced into equation (4) to improve its calculation efficiency. This step is accomplished by removing the binary variables in equation (4) that represent the streams not in the candidates. And the problem can be represented as:

$$\begin{aligned} & \sum_{i \in S} \frac{1}{\sigma_i} \left| \hat{F}_i - (\tilde{F}_i - \delta_i) \right| + W_i B_i + \sum_{j \notin S} \frac{1}{\sigma_j} \left| \hat{F}_j - \tilde{F}_j \right| \\ s.t \quad & \mathbf{A} \cdot \hat{F} = 0 \\ & \zeta U_i B_i \le |\delta_i| \le U_i B_i \\ & B_i \in binary \end{aligned} \quad (9)$$

When there are equivalent sets in the detected biases, the updated weighting factors can be farther used to output suspect biases with higher probabilities. This strategy can be regard as the utilization of both spatial and temporal redundancy of the measurement network. In order to understand this method easily, a flow chart represent the improved method is show in Figure 2.

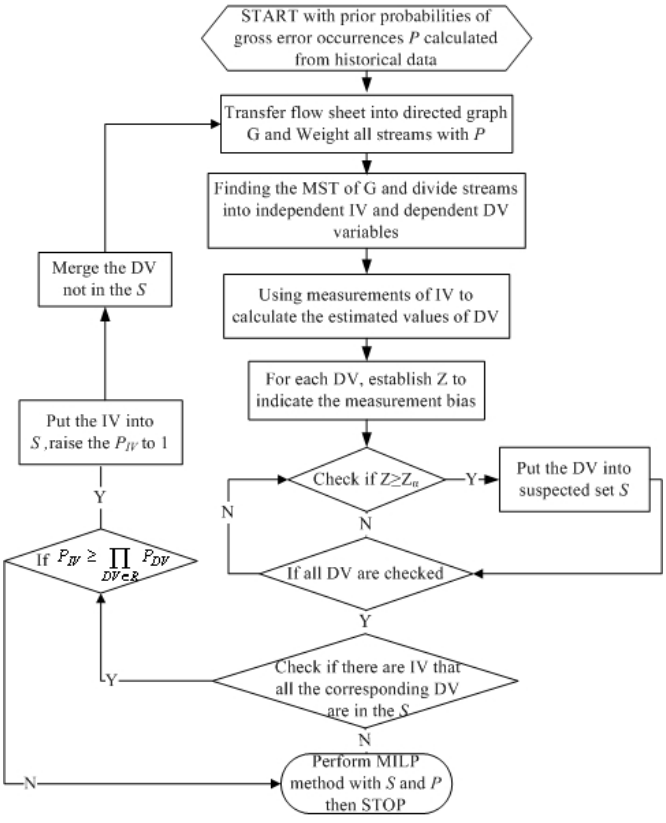

Figure 2 Flowchart of the improved method

5. SIMULATION STUDY

The system chosen for study was an industrial steam metering process (Serth and Heenan, 1986). And this system was also used for comparison in many other gross error detection methods (Rollins et al., 1996). The system used for simulation is illustrated in Figure 3 and the true flow rates can be found in Table 1.

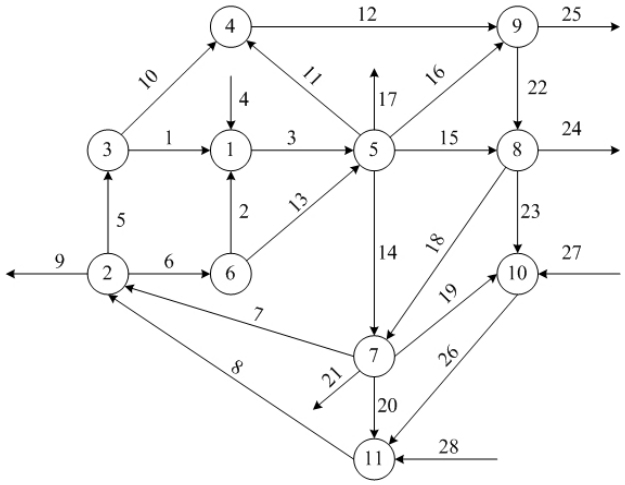

Figure 3 System of Steam Metering Process

Table 1 True Values for The flow rates in Steam Metering Process

No	True Flow Rate	No	True Flow Rate
1	0.86	15	1.5
2	1	16	0.591
3	111.82	17	0.81825
4	109.95	18	0.40575
5	53.27	19	0.19875
6	112.27	20	0.2625
7	2.32	21	2.1818
8	164.05	22	0.13625
9	0.86	23	0.06475
10	52.41	24	1.166
11	14.86	25	2.1363
12	67.27	26	2.033
13	111.27	27	1.7693
14	91.86	28	1.8058

Monte Carlo simulations are conducted in this section to evaluate the performance of the proposed method. In order to compare the simulation done by Soderstrom (2001), the number of biased variables was fixed at 3, 5 or 7 and the sign of the bias was randomly assigned with equal probability. Because there are prior probabilities of gross error occurrence for each variable, so the location of the biased variables was chosen according to the prior probabilities. As same as the simulation study made by Soderstrom (2001), the magnitude of the bias was chosen to be between 12.5% and 62.5% of true value of the measured variable in Table 1 and the standard deviation of the measurements are chosen to be $\sigma_i = 0.025F_i$ to enable a comparison to the simulation of original MILP method. All simulation in the paper was performed using MATLAB and LINGO 9 optimization software.

Average number of type I error (AVTI) and overall power (OP) (Narasimhan and Jordache, 2000) were applied to evaluate the performance of the improved method. The criteria are defined as follows:

$$\text{OP} = \frac{\text{Number of gross errors correctly identified}}{\text{Number of gross errors simulated}} \quad (10)$$

$$\text{AVTI} = \frac{\text{Number of gross errors wrongly identified}}{\text{Number of simulation trials made}} \quad (11)$$

Both OP and AVTI are calculated after 100 simulation runs in different conditions.

Table 2. Results for the improved method ($\alpha = 0.05$)

No. High Weight Biases	Percentage (Approximately)	No. Biased Stream	OP	AVTI
0	0	3	0.701	0.211
1	33.3%		0.763	0.271
2	66.7%		0.803	0.235
3	100%		0.905	0.255
0	0	5	0.626	0.532
1	33.3%		0.721	0.633
3	66.7%		0.750	0.668
5	100%		0.840	0.771
0	0	7	0.613	0.947
2	33.3%		0.678	0.961
4	66.7%		0.726	1.030
7	100%		0.780	1.160

Several interesting comparison can be made by checking the simulation results presented below. One of these is the influence of choosing more reliable independent variables. In

the simulation trials, the number of higher weight bias is set as 0, 1/3, 2/3, 1 times of number of the random selected biases. Higher the weight (prior probability of gross error occurrence) the stream has, less probability that this stream will be chosen as independent variable. Nearly all the OP is increased together with the No. Higher Weight Bias while the AVTI remains at the same level. When all biased streams are weighted with higher value, the OP increases significantly. But it is rarely that all the biased streams have higher weights, however, the result seems acceptable when few biased streams are given higher weights. Figure 4 and Figure 5 shows that good results can be obtained by fewer high weight biased streams.

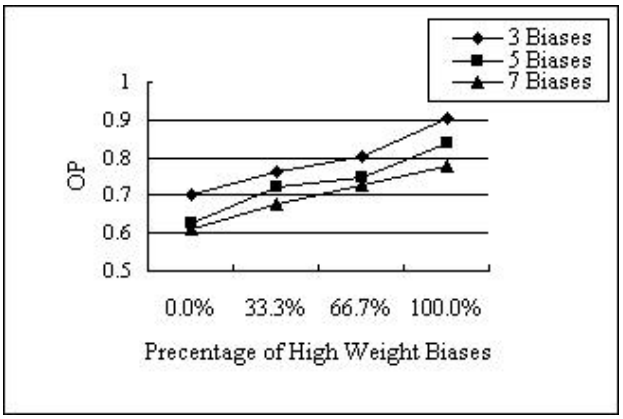

Figure 4. The Influence of high weight biases percentage on Overall Power

The number of the dependent variables is equal to the number of branches in the flowsheet, so the relationship between the number of dependent variables, independent variables and the total number of the measurements are:

$$n_{DV} = n - 1 \qquad (12)$$

$$n_{IV} = s - n + 1 \qquad (13)$$

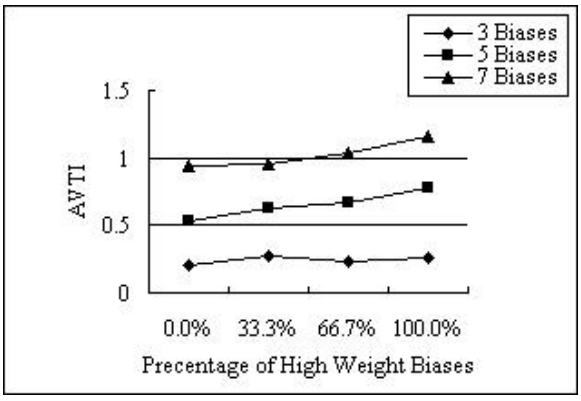

Figure 5. The Influence of high weight biases percentage on Average Type I Error

When the independent variable has bigger σ, the bias in the streams with small true value might be ignored in the candidate generation. This problem may be improved by adjusting the weights of streams with small σ (increase a little to raise the probability of being independent variable). Once the equivalent set exists because the candidate streams form a loop in the graph representing the flowsheet (Jiang and Bagajewicz, 1999), a solution can be used by updating weights to give the gross error candidates with larger probabilities.

Simulation times for different conditions are presented in Table 3. The results show that the solution times for gross error candidate generation (Average CPU Times 1) are rather limited. The solution times are increased due to the increase in the number of biased streams. And the trend is opposite when the percentage of high weight biases is increased. All simulations are performed in the Dell Inspiron 600m laptop.

Table 3. Average CPU Times for 3 biased streams test

Percentage of High Weight Bias	Average CPU Time 1(S)	Average CPU Time 2 (S)
0%	0.183	1.033
33.3%	0.170	0.982
66.7%	0.162	1.114
100%	0.142	1.042

6. CONCLUSIONS

This paper proposes an improved MILP method for simultaneous gross error detection and data reconciliation. The data reconciliation and gross error detection based on the framework of MILP has a great advantage of process system integration in process industry. Effective gross error candidate generation method is applied before the solution of mixed integer linear programming. The simulation results show that choosing reliable independent variables can improve the accuracy of measurements. This method was found to have better performance over the original MILP method. Reduction of binary variables in the objective function leads to a superior improvement of the calculation efficiency. This improvement will make the MILP method more suitable for application in the large-scale process industries.

ACKNOWLEDGMENTS

This work was supported by The National High Technology R&D program of China (2007AA04Z191) and The National Natural Science Foundation of China (60421002).

NOMENCLATURE

A	constraint matrix
a_{ki}	correlative parameter
B	binary variable
DV	dependent variable
F	true flow rate
$\widetilde{F}$	measured value
IV	independent variable

l	number of previous failure
m	sum of previous lifetimes
n	number of process nodes
R	set of independent variables used to estimate the dependent variable
s	number of streams
W	weighting factor
δ_i	deterministic bias
ε_i	random error
$\Gamma(\cdot)$	gamma function
σ_i	standard deviation

REFERENCES

Crowe, C. M. (1986) Reconciliation of Process Flow Rates by Matrix Projection. Part II: Nonlinear Case *AIChE Journal*, **32**, 616-623.

Crowe, C. M. (1996) Data reconciliation -- Progress and challenges. *Journal of Process Control*, **6**, 89-98.

Crowe, C. M., Campos, Y. A. G. and Hrymak, A. (1983) Reconciliation of Process Flow Rates by Matrix Projection. Part I: Linear Case. *AIChE Journal*, **29**, 881-888.

Jiang, Q. and Bagajewicz, M. J. (1999) On a Strategy of Serial Identification with Collective Compensation for Multiple Gross Error Estimation in Linear Steady-State Reconciliation. *Ind. Eng. Chem. Res*., **38**, 2119-2128.

Kuehn, D. R. and Davidson, H. (1961) Computer control. II Mathematics of control. *Chemical Engineering Progress*, **57**, 44-47.

Mei, C., Su, H. and Chu, J. (2006) An NT-MT Combined Method for Gross Error Detection and Data Reconciliation. *Chinese Journal of Chemical Engineering*, **14**, 592-596.

Narasimhan, S. and Jordache, C. (2000) Data Reconciliation and Gross Error Detection, Houston, Gulf Publishing Company.

Rollins, D. K., Cheng, Y. and Devanathan, S. (1996) Intelligent selection of hypothesis tests to enhance gross error identification. *Computers & Chemical Engineering*, **20**, 517-530.

Serth, R. W. and Heenan, W. A. (1986) Gross error detection and data reconciliation in steam-metering systems. *AIChE Journal*, **32**, 733-742.

Soderstrom, T. A., Himmelblau, D. M. and Edgar, T. F. (2001) A mixed integer optimization approach for simultaneous data reconciliation and identification of measurement bias. *Control Engineering Practice*, **9**, 869-876.

Tamhane, A. C., Iordache, C. and Mah, R. S. H. (1988a) A Bayesian approach to gross error detection in chemical process data : Part I : Model development. *Chemometrics and Intelligent Laboratory Systems*, **4**, 33-45.

Tamhane, A. C., Iordache, C. and Mah, R. S. H. (1988b) A Bayesian approach to gross error detection in chemical process data : Part II: Simulation results. *Chemometrics and Intelligent Laboratory Systems*, **4**, 131-146.

Tong, H. and Crowe, C. M. (1995) Detection of gross errors in data reconciliation by principal component analysis. *AIChE Journal*, **41**, 1712-1722

Wang, F., Jia, X. P., Zheng, S. Q. and Yue, J. C. (2004) An improved MT-NT method for gross error detection and data reconciliation. *Computers and Chemical Engineering*, **28**, 2189-2192.

Yang, Y., Ten, R. and Jao, L. (1995) Study of gross error detection and data reconciliation in process industries. *Computers and Chemical Engineering*, **19**, S217-S222.

Zhang, P., Rong, G. and Wang, Y. (2001) A new method of redundancy analysis in data reconciliation and its application. *Computers & Chemical Engineering*, **25**, 941-949.