Journal of Process Control 24 (2014) 13–26

Contents lists available at ScienceDirect

Journal of Process Control

journal homepage: www.elsevier.com/locate/jprocont

Progressive multi-block modelling for enhanced fault isolation in batch processes

Jeong Jin Hong [a], Jie Zhang [b,*], Julian Morris [b]

[a] CAE Team, Display R&D Centre, Samsung Display Co., Ltd., #95 Samsung 2-ro, Giheung-gu, Yongin-City, Gyeonggi-Do 446-711, Republic of Korea
[b] School of Chemical Engineering and Advance Materials, Newcastle University, Newcastle upon Tyne NE1 7RU, UK

ARTICLE INFO

Article history:
Received 3 April 2013
Received in revised form
18 September 2013
Accepted 17 October 2013
Available online 18 December 2013

Keywords:
Fault diagnosis
Process monitoring
Batch processes
On-line progressive modelling
Multi-block method

ABSTRACT

A multi-block progressive modelling approach is proposed for enhanced fault isolation in batch processes. The unfolding of batch data typically leads to matrices with a large number of columns and this complicates contribution analysis. In order to rapidly focus fault isolation in batch processes, it would be desirable to employ multi-block modelling techniques. Multi-block model such as consensus principal component analysis (CPCA) can produce multiple monitoring charts for sub-blocks and block loadings and block scores can be obtained which can represent unique behaviour of each sub-block. CPCA model uses super score which is the same as score from normal principal component analysis (PCA) model and it does not produce enhanced monitoring performance. Multi-block PCA (MBPCA) model using block score for model calculation can represent sub-blocks' character but block scores are obtained from super loading so it may not be the best way to describe sub-blocks. A new MBPCA model is proposed for better expression of each sub-block. Through progressive modelling and contribution analysis, variables related to or affected by the fault, as well as the associated time information, are gradually identified. This enables a fault propagation path being established. The proposed method is applied to a benchmark simulated penicillin production process, PenSim.

1. Introduction

Statistical process control (SPC) has been widely adopted to improve productivity and product quality without big investment of facilities. The most common SPC techniques such as Schewart chart and CUSUM statistics are generally referred to as 'Univariate Statistical Process Control' (USPC). They produce monitoring charts with control limits for a small number of variables to monitor those variables during the process [1]. However, this method is not very effective for processes with a large number of process variables because the USPC technique considers one variable each time and does not consider correlations among the process variables [1,2,23]. Multivariate statistical process control (MSPC) technique has been developed to overcome the limitation of the USPC. MSPC utilises multivariate statistical techniques, such as principal component analysis (PCA) and partial least square (PLS), to perform dimension reduction for the high dimensional process data with high level of correlations so that the nominal process behaviour can be represented by a smaller dimension latent variables or scores [1,3,18,21,25].

MSPC techniques were first developed for continuous processes and then extended to batch processes. Batch process is the widely used processing type in industries such as biochemical, pharmaceutical, semiconductor and display manufacturing industry [4,19,26,27,29]. A big difference between continuous and batch processes is the process duration. A continuous process typically has a long processing duration once the process is started and the conversion from raw material to product is continuous. However, a batch process has shorter processing duration and the production is intermittent. In addition, batch processes have batch to batch variations meaning that each batch shows slightly different performances with the same recipe.

Batch process monitoring using the multivariate statistical process control (MSPC) method based on multi-way principal component analysis (MPCA) was proposed and developed by Nomikos and MacGregor [4–6]. Since then, there have been many published researches on batch process monitoring using MSPC techniques [19,20,22,26,27,29]. Monitoring charts in terms of the monitoring statistics, squared prediction errors (SPE) and the T^2 statistics, are produced and the monitoring statistics are checked with their control limits. Once a monitoring statistics exceeds its control limit, an abnormal situation is detected. Contribution plots [7] are typically used in fault diagnosis. Contributions from individual process variables to the monitoring statistics are produced to identify the process variables that are most related to the fault. However, this procedure is not able to determine if a variable is just affected by

* Corresponding author. Tel.: +44 191 2087240; fax: +44 191 2085292.
E-mail addresses: jj.hong@samsung.com (J.J. Hong), jie.zhang@newcastle.ac.uk (J. Zhang).

0959-1524/$ – see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jprocont.2013.10.013

the fault or the variable is the cause of the fault. And it does not provide time information of when the abnormalities on the highly contributed variables occurred.

A progressive PCA modelling method was developed to overcome this problem [8]. However, it deals with all the process variables in the process as one data block. This may not be efficient when the monitored process contains a large number of variables and/or data samples. Multi-block method such as consensus PCA (CPCA) model [9] can be applied as alternative method to MPCA as it has multiple sets of scores and prediction can be carried out for the data in each sub-block rather than the entire block of data. Therefore, each sub-block can be monitored individually and this can give the advantage of finding fault location more effectively in that the problem can be localised in one or a small number of data blocks [10,11]. For a CPCA model, super score is used for computing block loadings for sub-blocks. As super score represents overall behaviour during a whole processing time, multiple sets of block loadings are all reflected by the overall behaviour. It means that the loadings for each block tend to exhibit a profile for a whole duration rather than a profile for each sub-block and it may not be ideal to analyse local process behaviours of sub-blocks.

A new multi-block PCA model is proposed in this study which uses block scores for computing a set of block loadings instead of the super score. Thus, a set of block scores computed by this method can address the behaviour existed in sub-blocks better than the one computed by CPCA method. The new multi-block PCA method is used within the progressive modelling framework to establish fault propagation path efficiently for batch processes. The proposed method is applied to the benchmark simulation of a fed-batch penicillin production process, PenSim [12]. Three fault batches were analysed using three different modelling methods, the conventional MPCA, CPCA and the new MBPCA. These modelling methods were applied with the progressive modelling procedure to compare the performances. The proposed multi-block PCA gives better results than other two methods for all cases when it combines with the progressive modelling scheme.

The paper is organised as follows. Section 2 presents the background of this study. Section 3 presents the new multi-block PCA method and the proposed progressive multi-block modelling approach to identify fault propagation path. Application results on a benchmark simulated fed-batch penicillin production process are presented in Section 4. Section 5 draws some concluding remarks.

2. Background

2.1. PCA for batch process monitoring

Batch process operation data are generally stored as three-dimension data arrays, where the three dimensions are batch numbers, variables and sample times. Such data are transformed to two-dimension data structure which can be analysed using PCA. Multi-way PCA method is the most popular way to apply the batch process data into the PCA model. It has a step called data unfolding which transforms the three dimensional batch data into two a dimensional data structure. There are two popular ways for data unfolding: batch-wise unfolding proposed by Nomikos and Macgregor [4–6] and variable-wise unfolding proposed by Wold et al. (1998) [30]. Under batch-wise unfolding, the data from an entire batch is transformed into a long row in the unfolded data matrix. With variable-wise unfolding, each column of the unfolded data matrix represents a variable. In this paper, the batch-wise unfolding method is used to remove non-linearity in the data as PCA algorithm itself is a linear method.

Consider an unfolded data matrix **X** with a size of $r \times c$. Each row of **X** represents a batch and a column represents a data point for each variable measured at each sampling time. A PCA model describes **X** as the sum of outer product of vectors, **t** and **p**. Thus, **X** can be expressed as

$$\mathbf{X} = \mathbf{T}\mathbf{P}^{\mathrm{T}} = \sum_{i=1}^{c} \mathbf{t}_i \mathbf{p}_i^{\mathrm{T}} \tag{1}$$

In Eq. (1), $\mathbf{t}_i$ is the ith score vector and $\mathbf{p}_i$ is the ith loading vector. The reconstruction of **X** using the first n ($n \le c$) principal components can be calculated as

$$\hat{\mathbf{X}} = \mathbf{T}_n \mathbf{P}_n^{\mathrm{T}} = \sum_{i=1}^{n} \mathbf{t}_i \mathbf{p}_i^{\mathrm{T}} \tag{2}$$

As the loading matrix is obtained from the normal data, it describes correlations among the variables under the normal operating condition. When a fault presents, it will change the correlation structure among the process variables and/or change the magnitudes of some process variables. These will be detected by the SPE and/or the T^2 statistics respectively.

2.2. Multi-block methods

Multi-block PCA modelling is an alternative way of the conventional PCA modelling for improved monitoring analysis and efficiency. Many batch chemical processes have several phases and different characteristics and data correlations may present in these phases [13,14]. In the conventional MPCA with batch-wise unfolding, only one score value is calculated to describe one batch and it would be difficult to describe a whole system containing multiple characteristics, because the score is calculated from the one loading set containing different types of correlations. Multi-block methods can be considered to model multi-phase/stage behaviour of the process because multiple scores are calculated for a batch from the sub-groups (phases/stages) [13,14]. It means that a score for a sub-block is calculated using the loading set representing the correlations only for the given sub-block. Thus, it can represent its sub-groups better than the score from a single data block. It basically divides the data into multiple sub-groups for the divided groups to have their own correlation structures. Therefore, each sub-block has its block loadings and scores.

There are several published multi-block methods. The first method called consensus PCA (CPCA) was proposed by Wold et al. [9]. This CPCA algorithm is based on the ordinary PCA algorithm. A matrix of super scores of CPCA is the same as the scores of ordinary PCA, so it does not give more information than ordinary PCA (Westerhuis et al., 1998; AlGhazzawi and Lennox, 2008) [31,32]. The only difference is whether data is arranged with multiple blocks or not. The algorithm of CPCA is shown below.

Algorithm of CPCA (in the case of 2 sub-blocks):

1. Set initial super scores, $\mathbf{t}_3$
2. Calculate block loadings: $\mathbf{p}_1 = \mathbf{X}_1{}^{\mathrm{T}}\mathbf{t}_3/(\mathbf{t}_3{}^{\mathrm{T}}\mathbf{t}_3)$, $\mathbf{p}_2 = \mathbf{X}_2{}^{\mathrm{T}}\mathbf{t}_3/(\mathbf{t}_3{}^{\mathrm{T}}\mathbf{t}_3)$
3. Normalised block loadings, $\mathbf{p}_1$ and $\mathbf{p}_2$, to unit length.
4. Calculate block scores: $\mathbf{t}_1 = \mathbf{X}_1\mathbf{p}_1$, $\mathbf{t}_2 = \mathbf{X}_2\mathbf{p}_2$
5. Combine block scores into one block **T**: $\mathbf{T} = [\mathbf{t}_1\ \mathbf{t}_2]$
6. Calculate super weight $\mathbf{w}_3$: $\mathbf{w}_3 = \mathbf{T}^{\mathrm{T}}\mathbf{t}_3/\mathrm{t}3^{\mathrm{T}}\mathbf{t}_3$
7. Normalise $\mathbf{w}_3$ to unit length.
8. Calculate super score $\mathbf{t}_3$: $\mathbf{t}_3 = \mathbf{T}\mathbf{w}_3$
9. Go back to step 2 until $\mathbf{t}_3$ is converged.
10. Obtain residuals: $\mathbf{E}_1 = \mathbf{X}_1\text{-}\mathbf{t}_3\mathbf{p}_1{}^{\mathrm{T}}$, $\mathbf{E}_2 = \mathbf{X}_2 - \mathbf{t}_3\mathbf{p}_2{}^{\mathrm{T}}$
11. Replace $\mathbf{X}_1$ and $\mathbf{X}_2$ by $\mathbf{E}_1$ and $\mathbf{E}_2$ to compute the next principal component.
12. Check if the required model accuracy or the number of required PC is reached, if yes, stop the calculation, if no, go back to step 1.