# Statistics Pattern Analysis: A New Process Monitoring Framework and its Application to Semiconductor Batch Processes

**Q. Peter He**
Dept. of Chemical Engineering, Tuskegee University, Tuskegee, AL 36088

**Jin Wang**
Dept. of Chemical Engineering, Auburn University, Auburn, AL 36849

*DOI 10.1002/aic.12247*
*Published online April 15, 2010 in Wiley Online Library (wileyonlinelibrary.com).*

*In the semiconductor industry, process monitoring has been recognized as a critical component of the manufacturing system. Multivariate statistical process monitoring (SPM) techniques, such as multiway principal component analysis and multiway partial least squares, have been extend to monitor semiconductor processes. These SPM methods require extensive, often off-line data preprocessing such as data unfolding, trajectory mean shift, and trajectory alignment. This requirement is probably not an issue for the traditional chemical batch processes but it poses a significant challenge for semiconductor batch processes. This is because data preprocessing makes model building and maintenance extremely labor intensive due to the large number of models in a typical semiconductor fab. In addition, semiconductor process data often show more severe nonnormality compared to those of the traditional chemical process under closed-loop control, which results in suboptimal performance in many applications. To address these challenges, several pattern classification based monitoring (PCM) methods have been developed recently, but some limitations remain and trajectory alignment is still required. In this article, we analyze the fundamental reasons for the limitations of the SPM and PCM methods when applied to monitor semiconductor processes. In addition, we propose a new statistics pattern analysis (SPA) framework to address the challenges associated with semiconductor processes. By monitoring batch statistics, the proposed SPA framework not only eliminates all data preprocessing steps but also provides superior fault detection performance. Finally, we use an industrial example to demonstrate the advantages of the proposed SPA framework, and examine the fundamental reasons for the improved performance from SPA.* © 2010 American Institute of Chemical Engineers *AIChE J,* 57: 107–121, 2011
*Keywords: statistics pattern analysis, fault detection, semiconductor manufacturing, statistical process monitoring, pattern recognition, higher-order statistics*

## Introduction

Semiconductor technology lies at the heart of the revolution in computing, communications, consumer electronics, transportation and health care. Over the past few decades,

Correspondence concerning this article should be addressed to Q. P. He at qhe@tuskegee.edu.

the semiconductor industry has maintained an average growth of 15% per year. This steady growth is a result of continuous reduction in the cost-per-function by 25–30% per year, in which advanced process monitoring techniques have played a key role. Specifically, it has been recognized that effective fault detection and classification (FDC) applications help increase overall equipment efficiency by detecting abnormal operation conditions promptly.

Generally speaking, there are two main categories in process monitoring methods: univariate methods and multivariate methods. In univariate methods such as Shewhart, cumulative sum and exponentially weighted moving average charts, each variable (usually product quality) is monitored separately to identify the special causes of variation (i.e., faults) from natural or common causes of variation. In multivariate methods such as the principal component analysis (PCA) and partial least squares (PLS) methods,[1–9] multiple variables are monitored together to detect faults that violate normal process variable correlations. In this work, we focus on the multivariate methods because they offer improved monitoring performance. Compared to the univariate methods, the multivariate methods offer reduced Type I (false alarm) and Type II (missed detection) errors.

Multivariate process monitoring in the semiconductor industry is enabled by the availability of massive amount of trace or machine data, i.e., data generated and recorded by process equipments in modern fabs*. Driven by their importance to the manufacturing productivity, advanced multivariate monitoring algorithms have been developed/adapted and implemented in the semiconductor industry to monitor a variety of processes[10] such as etch,[11–19] overlay,[20] lithography,[21,22] deposition,[23,24] and plasma stripper.[22] Based on the way that normal operation condition is captured for FDC, existing multivariate methods can be further categorized into two groups: statistical process monitoring (SPM) methods and pattern classification based monitoring (PCM) methods. The SPM methods extract a low-dimensional, linear model from training data to represent the normal operation condition; while the PCM methods use the complete training data set as the model of the normal operation condition directly. In this work, we discuss the advantages and limitations of the SPM and PCM methods, and propose a novel fault detection framework—statistics pattern analysis (SPA) to address some of the limitations associated with the existing methods. In the rest of this section, we briefly review the application of the SPM and PCM methods in monitoring semiconductor processes.

### *Statistical process monitoring methods*

The multivariate SPM methods such as multiway PCA (MPCA) and multiway PLS (MPLS) have long been used in monitoring batch processes in the traditional chemical and petrochemical industries.[1–9] The characteristics associated with chemical batch processes, such as unequal batch and/or step length, unsynchronized or misaligned batch trajectory and multimodal batch trajectory distribution, usually result in non-Gaussian distributed data and deteriorate the monitoring performance of MPCA and MPLS. To address these challenges, various data preprocessing steps are usually required for the MPCA and MPLS methods to achieve satisfactory monitoring performance. These preprocessing steps, including trajectory alignment/warping, trajectory mean shift and data unfolding, are often performed off-line and could be difficult to automate.

In the last few decades, the multivariate SPM methods have been adopted to monitoring semiconductor processes. The most commonly used multivariate SPM methods in the semiconductor industry are MPCA[11–16] and MPLS.[15,24] Similar to chemical process monitoring, two steps are involved in SPM for semiconductor processes: (1) correlation information among different variables is extracted by applying dimension reduction techniques to measurements of multiple variables; (2) fault detection is performed by examining whether a test sample follows the same correlation pattern exhibited by the normal training samples.

It is worth noting that the characteristics of semiconductor processes are similar to those of the traditional chemical batch processes. However, the data generated from semiconductor processes often show more severe nonnormality compared to those of the traditional chemical processes, due to frequent tool preventive maintenance (PM) and much higher product mixing in the semiconductor industry. In addition, the number of models in a highmix fab is in the range of several thousands to tens of thousands,[25,26] which makes model building and maintenance much more labor intensive than in a typical traditional chemical plant. As a result, despite the success that the SPM methods have achieved in monitoring the traditional chemical processes, their success in monitoring semiconductor processes has been limited. The impacts of the data nonnormality on process monitoring are discussed in more detail in the next section.

### *Pattern classification based monitoring methods*

To reduce data preprocessing steps mentioned previously, several PCM methods have been developed recently, which make use of the fault detection k-nearest-neighbor rule (FD-kNN)[18,19,27–30] and Mahalanobis distance.[23,31,32] In addition, several commercial process monitoring software packages, such as ModelWare† and flexible golden pattern,[33] also use pattern classification based algorithms. These PCM methods perform fault detection based on the simple idea that the trajectory of a normal sample is similar to those of the normal training samples, while the trajectory of a fault sample exhibits some deviation from those of the normal training samples. In other words, in the PCM methods, the complete training data set is used directly as the model for normal operation conditions. By doing so, the process nonlinearity and nonnormality under normal operation conditions can be captured directly in the PCM methods, therefore trajectory mean shift is not required. Meanwhile, the number of the required training samples can be significantly reduced because batch unfolding is not needed either. Last but not least, these methods seem to offer better fault detection performance compared to the SPM methods in many cases.[18,33] However, the PCM methods still require batch trajectory alignment to make batch trajectories synchronized with equal length, and usually requires larger data storage space and longer computation time compared to the SPM methods.

*Various process equipments are often called "tools" in the semiconductor fabs.

†An FDC product of PDF Solutions.

**Table 1. Data Preprocessing Steps Required by Different Methods for Semiconductor Process Monitoring**

| Category | Method | Trajectory Mean Shift | Trajectory Alignment | Batch Unfolding |
|---|---|---|---|---|
| SPM | MPCA[11,21,22] | Yes | Yes | Yes |
| | MPLS[7,8,15,24] | Yes | Yes | Yes |
| | PARAFAC[11,34,35] | Yes | Yes | No |
| | TLD[11,36] | Yes | Yes | No |
| PCM | Mahalanobis[23,31,32] | Yes | Yes | No |
| | FD-kNN[18,27–29] | No | Yes | No |
| | PC-kNN[19,30] | No | Yes | No |
| SPA (this work) | | No | No | No |

To the authors' best knowledge, all existing fault detection methods require certain levels of data preprocessing for semiconductor process monitoring. The data preprocessing steps required by some widely used SPM and PCM methods are summarized in Table 1, where commercial softwares are not compared because their technical details are not available.

In this work, we propose a novel process monitoring framework, termed SPA, to eliminate the data preprocessing steps mentioned above. The major difference between the traditional MPCA-based and the proposed SPA-based fault detection methods is that in general MPCA monitors the process variables, while SPA monitors various statistics of the process variables. In other words, in MPCA a singular value decomposition (SVD) is usually applied to the measurements of the process variables. The obtained model captures the dominant correlations of process variables under normal operation, and the new measurements of the process variables are projected onto the MPCA model to perform fault detection. In SPA, SVD is applied to the statistics variables calculated from process measurements under normal operation. The obtained model captures the dominant correlations of the statistics variables, and the statistics variables calculated from the new measurements are projected onto the model to perform fault detection. In this way, the statistics that capture different characteristics of the process can be selected to model the normal process operation, and process nonlinearity and nonnormality can be quantified explicitly and used for process monitoring. As elaborated later in this work, the proposed SPA method not only eliminates all data preprocessing steps, but also provides superior fault detection performance. The rest of the article is organized as follows. In the next section, MPCA and FD-kNN are briefly reviewed and their limitations in monitoring semiconductor processes are discussed. Then the SPA framework is presented. The performance of the proposed SPA method is illustrated through a benchmark industrial case study. Conclusions and discussions are given at the end of the article.

### *Review of MPCA and FD-kNN*

In this section, we briefly review the MPCA and FD-kNN methods, which are chosen as the representatives of SPM and PCM methods, respectively. We also discuss the fundamental reasons for their limitations in monitoring semiconductor processes. The general notations used in the rest of the paper are in accordance with the following rules: scalar are denoted by italic lowercase characters ($x$), vectors by bold lowercase characters ($\mathbf{x}$), matrices by bold uppercase characters ($\mathbf{X}$) and 3D arrays by an underlined bold uppercase characters ($\underline{\boldsymbol{X}}$).

### *Multiway principal component analysis*

The raw data collected from a batch process is a 3D array (batch $\times$ variable $\times$ time) with unequal lengths in batch duration, as shown in Figure 1a. PCA cannot be applied directly to the raw data and several steps of data preprocessing are needed as illustrated in Figure 1. First trajectory alignment is applied to make batches synchronized (i.e., all batches are observed from the same evolutionary standpoint); then batch lengthes are made the same; finally trajectory mean shift (e.g., subtracting trajectory mean) is applied to make all trajectories follow a unimodal distribution. The raw data after these preprocessing steps is denoted by $\underline{\boldsymbol{X}}$ and is shown in Figure 1b. To perform PCA, the preprocessed 3D array needs to be further unfolded into a 2D matrix, denoted by $\mathbf{X}$, where each row represents a batch in the 3D array before unfolding. The 2D matrix is shown in Figure 1c, which is ready to be analyzed by PCA. The above procedure is termed multiway PCA or MPCA.

The dimensions of the unfolded data matrix $\mathbf{X}$ are $n \times m$ where $n$ is the number of samples (batches) and $m$ is the number of variables after unfolding. $\mathbf{X}$ is first scaled to zero

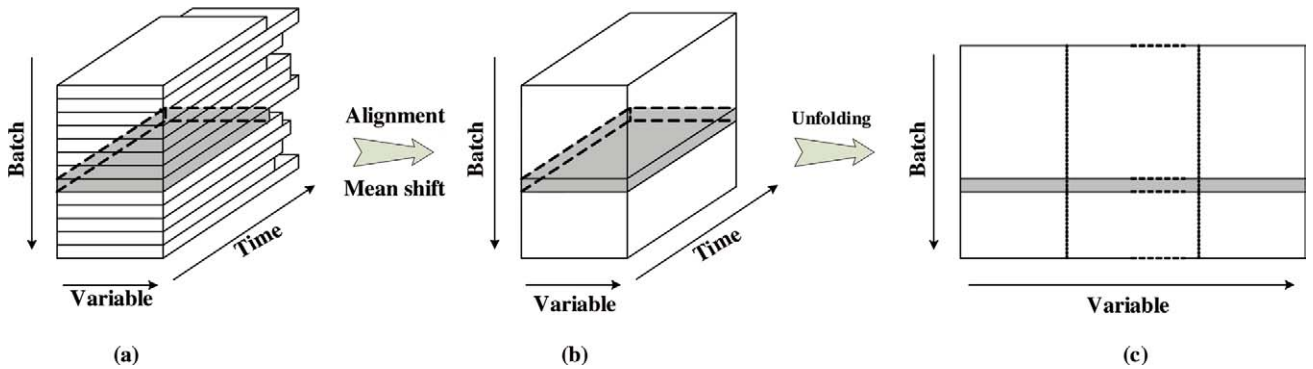



**Figure 1. Illustration of data preprocessing involved in MPCA.**

(a) Original batch records with unequal length; (b) 3D array after trajectory alignment and shifting; and (c) 2D matrix after trajectory unfolding. (A sample batch at the preprocessing stage is highlighted in gray.) [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

mean for the covariance-based MPCA and further to unit variance for the correlation-based MPCA. By performing either the Nonlinear Iterative Partial Least Squares (NIPALS)[37] or SVD, the scaled matrix $\mathbf{X}$ is decomposed as follows,

$$\mathbf{X} = \mathbf{T}\mathbf{P}^T + \tilde{\mathbf{X}} \tag{1}$$

where $\mathbf{T} \in \Re^{n\times l}$ and $\mathbf{P} \in \Re^{m\times l}$ are the score and loading matrices, respectively, $l$ is the number of principal components (usually $l \ll m$), and $\tilde{X}$ is the residual matrix.

The squared prediction error (SPE) and the Hotelling's $T^2$ are the most commonly used indices for fault detection. The SPE index indicates how well each sample conforms to the model, measured by the projection of the sample vector $\mathbf{x}$ on the residual space:

$$\text{SPE} = \| \tilde{\mathbf{x}} \|^2 = \| (\mathbf{I} - \mathbf{P}\mathbf{P}^T)\mathbf{x} \|^2 \tag{2}$$

The process is considered normal if SPE $\leq \delta_\alpha^2$, where $\delta_\alpha^2$ is the confidence limit for SPE with a significance level $\alpha$. The Hotelling's $T^2$ index is a measure of the process variation in the principal component subspace:

$$T^2 = \mathbf{x}^T\mathbf{P}\Lambda^{-1}\mathbf{P}^T\mathbf{x} \tag{3}$$

where $\Lambda$ is the diagonal matrix of the $l$ largest eigenvalues of $\mathbf{X}\mathbf{X}^T$. The $T^2$ statistic forms a hyper-ellipsoid, which represents the joint limits of variations that can be explained by a set of common causes. For a given significance level $\alpha$, the process is considered normal if $T^2 \leq T_\alpha^2$. The confidence limits $\delta_\alpha^2$ and $T_\alpha^2$ can be calculated as in Refs. 38, 39. In addition, the confidence limits of SPE and $T^2$ can also be determined empirically.[11,40]

### *FD-kNN*

In FD-kNN the basic idea for fault detection is implemented through evaluation of the kNN distance, which is defined as the average squared distance between a sample and its kNN in the training data set, i.e.,

$$\mathcal{D}_i^2 = \frac{1}{k}\sum_{j=1}^{k} d_{ij}^2 \tag{4}$$

where $d_{ij}^2$ denotes squared Euclidean distance from sample $i$ to its $j$-th nearest neighbor.

The FD-kNN method assumes that the kNN distances of normal samples follow a noncentral $\chi^2$ distribution while the kNN distance of faulty samples do not follow the same distribution. Thus, the FD-kNN method also consists of two steps: model building and fault detection. In the first step, the confidence limit for the kNN distance is determined based on the normal training samples; in the second step, the new sample's kNN distance is calculated and compared to the predetermined confidence limit to perform fault detection.

Similar to the multivariate SPM methods, the confidence limit of $\mathcal{D}_i^2$ can be determined in two ways. One is based on the assumption that $\mathcal{D}_i^2$ follows a noncentral $\chi^2$ distribution, the other is through the calibration or use of validation data.[11,40] Details on the FD-kNN method, such as the choice of $k$ and determination of $\mathcal{D}_\alpha^2$, can be found in Refs. 18, 19 and 30.

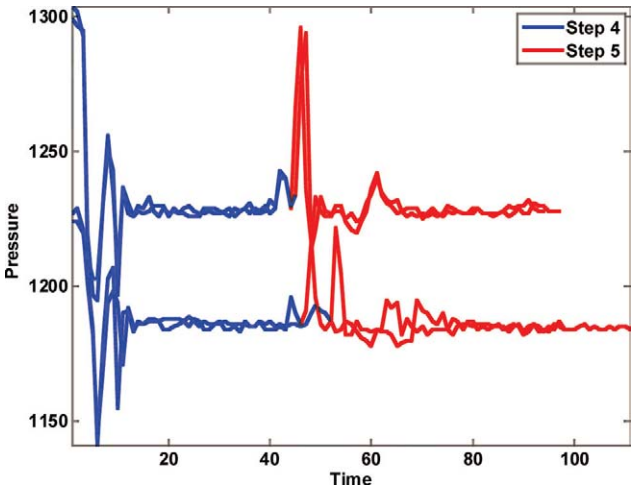



**Figure 2. Illustration of the characteristics of semiconductor processes: the trajectories (Steps 4 and 5) of a single variable from four different batches are shown.**

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

### *Limitations of MPCA and FD-kNN in monitoring semiconductor processes*

As discussed previously, both the MPCA and FD-kNN methods exhibit some limitations and require certain levels of data processing when applied to monitoring semiconductor processes. In this subsection, we first present the characteristics of semiconductor processes and the associated challenges for process monitoring. Then, we discuss the impacts of these challenges on the monitoring performance of MPCA and FD-kNN methods.

### *Characteristics of semiconductor processes*

Here, we use an industrial etch example[11] to illustrate some of the characteristics of semiconductor processes. The etch process is to remove the TiN/A1–0.5%Cu/TiN/oxide stack with an inductively coupled $BCl_3/Cl_2$ plasma. The process consists of six steps: (1) gas flow, (2) pressure stabilization, (3) plasma ignition, (4) etching of the aluminum layer, (5) over-etching of the underlying TiN and oxide layers, and (6) venting the chamber. Among them, Steps 4 and 5 are the actual etch processes and should be monitored closely. Figure 2 shows four batch trajectories of one process variable (chamber pressure) for Steps 4 and 5. It illustrates the following characteristics of semiconductor processes: unequal batch and/or step length; unsynchronized or misaligned batch trajectory; and multimodal batch trajectory distribution.

Unequal batch/step length is mainly caused by closed-loop control. With run-to-run controllers implemented to control the etch depth, the etch times corresponding to Steps 4 and

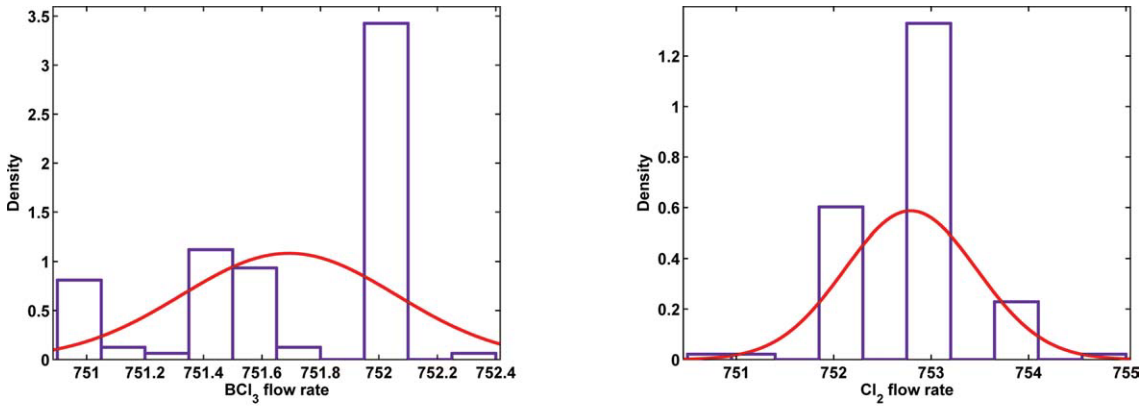



**Figure 3. Histograms of two process variables (measured at a given timestamp) cross all normal batches (a) $BCl_3$ flow rate; (b) $Cl_2$ flow rate**

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

5 are adjusted at each run with the purpose of reducing the final etch depth variation. Consequently, the durations of etch steps and the whole batch vary from batch to batch, which also results in the misalignment of batch trajectories. The multimodal batch trajectory distribution is mainly due to various PM events such as in situ cleaning and part replacement, which all cause shifts in equipment states.

The normal process data of semiconductor processes usually do not follow multivariate Gaussian distributions, which is also the case for some traditional chemical batch processes. However, the data generated from semiconductor processes show more severe nonnormality, as illustrated using the etch example. In Figure 3, the histograms of two variables ($BCl_3$ and $Cl_2$ flow rates) measured at a given timestamp are plotted for all normal batches. The fitted probability density functions (pdf) of Gaussian distributions are also plotted. Clearly the Gaussian distribution describes the actual data distribution poorly for this industrial example. As both the MPCA and FD-kNN based fault detection methods assume that the process data follow multivariate Gaussian distributions (to certain extent), the semiconductor process characteristics pose serious challenges to the performance of these methods. As discussed in detail below, the non-Gaussian distributed data is one of the root causes for the limited success of the MPCA and FD-kNN methods in monitoring semiconductor processes.

### *Limitations of the MPCA based fault detection method*

It should be noted that there is no assumption made about the data distribution when PCA or MPCA is applied to reduce data dimension or to find patterns/clusters in the data. However, when applied for process monitoring, several fundamental assumptions have to be satisfied for the MPCA-based fault detection methods to perform well. Specifically, assumptions are made when determining the confidence limits for the $T^2$ and SPE statistics. The confidence limit of $T^2$ is determined with the assumption that the scores follow a multivariate Gaussian distribution with population mean 0 and estimated covariance matrix $\Lambda$.[2,41] The assumption behind the confidence limit for SPE is that the variables in the unfolded matrix $\mathbf{X}$ has a multivariate Gaussian distribution with population mean 0.[2,38] For most semiconductor processes, the assumption of multivariate Gaussian distribution for the unfolded variables or the scores is violated because of the misaligned batch trajectories and process shifts following tool maintenance events. This assumption is not always justified even after various data preprocessing steps as shown later in the industrial case study section. The non-Gaussian distributed data is one of the major factors that affect the performance of the MPCA-based fault detection methods. The other major factor is that MPCA is a second-order method, which only considers the mean and the variance–covariance of the data. Therefore, MPCA lacks the capability of providing higher-order representations for non-Gaussian data, while the industrial data, especially those from semiconductor processes, are often non-Gaussian.[17,42]

### *Limitations of the FD-kNN method*

Compared to the MPCA method, the fundamental assumption of the FD-kNN method is somewhat relaxed. The FD-kNN method requires that the $k$ nearest neighbors of a given operation point are independent, identically distributed (i.i.d.) random variables that follow a multivariate Gaussian distribution. In other words, FD-kNN does not require all batches within the whole operation region to satisfy these assumptions but just the process data within a small neighborhood. The size of the neighborhood is determined by $k$ (the number of nearest neighbors).

With this relaxation, the FD-kNN method can handle process nonlinearity without requiring data unfolding and mean-centering, as any nonlinear process can be approximated by a piece-wise linear model in a small neighborhood. In addition, the FD-kNN method can handle multimodal distribution without requiring trajectory mean shift, as the process of identifying kNN automatically eliminates the data from other clusters. However, to satisfy the assumption that different samples (i.e., batches) are identically distributed, even within a small neighborhood, synchronized batch trajectories are required, which is why data alignment is still required by the FD-kNN method.

Compared to MPCA, FD-kNN has some advantages. First, the number of samples required by the FD-kNN method to

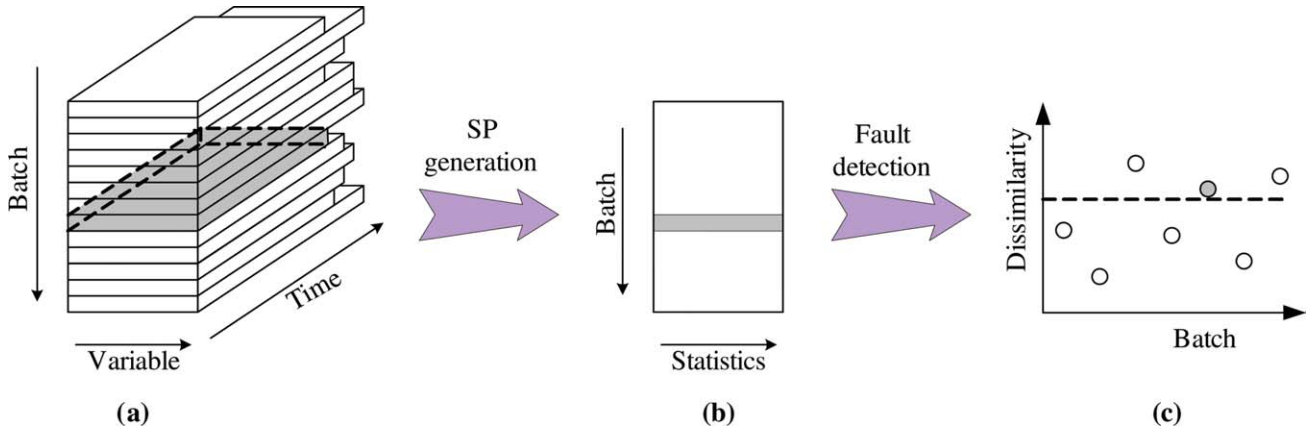



**Figure 4. The schematic plot of the SPA framework for fault detection (a) Original batch records with unequal length; (b) SP generation by calculating various batch statistics; (c) fault detection by dissimilarity quantification.**

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

build a reliable model is much smaller. This is because in FD-kNN, the contributions from different measurements along the batch trajectory are summed together when calculating the kNN distance, instead of being considered separately as different variables as in MPCA. Second, the FD-kNN method is able to eliminate the need of data unfolding and trajectory mean shift, because it uses the complete training data set as the model for normal operation conditions, which captures the process nonlinearity and multimodal distribution. The disadvantage of FD-kNN is that it requires larger memory space to store the training data set and higher on-line computation load compared to the MPCA method. This is because for each test sample, FD-kNN searches through the whole training data set to identify the sample's kNN. A modified version, PC-kNN, was developed recently to address this disadvantage.[19,30]

### *Disadvantages of data preprocessing*

As discussed above, both the MPCA and FD-kNN methods require certain levels of data preprocessing to monitor semiconductor processes. In many cases, these data preprocessing steps improve monitoring performance by making the processed data conform better to a multivariate Gaussian distribution. However, there are some disadvantages associated with these requirements, and the major ones are discussed here. First, the data preprocessing procedures such as trajectory alignment often require human intervention, which makes the automation of process monitoring difficult. Second, the data preprocessing may distort process information and results in deteriorated monitoring performance[43]; More importantly, because of the high-mix production in semiconductor fabs, these required data preprocessing steps make the model building and maintenance extremely labor intensive. In the high-mix production environment, it is common to have dozens or hundreds of different products running on the same piece of equipment. As different products usually have different batch durations, if data unfolding and alignment are needed to perform fault detection, different models are needed for different products. In addition, old products are constantly retired and new products are constantly added, therefore new models need to be developed constantly. For example, by 2006 there were more than 7,000 active FDC models at IBM [25] and over 30,000 models at Intel [26]. As a result, it is highly desirable to minimize the required data preprocessing steps without sacrificing monitoring performance.

### *Statistics Pattern Analysis*

In this section, we present a novel SPM framework, termed SPA, to address the challenges associated with monitoring semiconductor and other batch processes. Unlike existing methods that monitor process variables to perform FDC, the proposed SPA framework monitors batch statistics. As shown in the rest of the article, the new method not only eliminates all data preprocessing steps, but also provides superior monitoring performance.

In the SPA framework we hypothesize that the batch behavior can be characterized more fully by the variance–covariance structure of batch statistics than by the variance–covariance structure of process variables. In this work, a statistics pattern (SP) is a collection of various statistics calculated from a batch trajectory which capture the characteristics of each individual variable (such as the mean and variance), as well as the interactions among different variables (such as the covariance). The basic idea of the SPA framework is that the SPs of normal batches follow a similar pattern (i.e., normal pattern), while the SPs of abnormal or mis-processed batches must show some deviation from the normal pattern. The idea is supported by the fact that different batches processed on the same equipment are governed by the same physical/chemical mechanisms, such as mass transport, kinetics and thermodynamics. It is worth noting that depending on the specific process, a batch trajectory may need to be separated into several steps and the step statistics, instead of the batch statistics, should be computed for process monitoring.

Figure 4 illustrates the two basic steps involved in the SPA-based fault detection method. The first step is SP generation where we extract the characteristics of a batch trajectory by calculating various statistics. The second step is dissimilarity quantification and fault detection, where we quantify the dissimilarities among the SPs of normal batches and

determine the confidence limit (or upper control limit) of the dissimilarity associated with a normal batch. For fault detection, when a new batch record becomes available, its SP is first calculated, and its dissimilarity is then quantified and compared with the confidence limit to detect possible faults. If its dissimilarity index is below the confidence limit, it is classified as a normal batch; otherwise, it is detected as a fault. The details of the two steps are presented in the following subsections.

### *Batch statistics pattern generation*

We use $\chi$ to denote a batch record before unfolding, which corresponds to one layer of the raw data stack highlighted in Figure 4a, i.e.,

$$\mathcal{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_v \end{bmatrix} = \begin{bmatrix} x_1(1) & x_1(2) & \cdots & x_1(d) \\ x_2(1) & x_2(2) & \cdots & x_2(d) \\ \vdots & \vdots & \ddots & \vdots \\ x_v(1) & x_v(2) & \cdots & x_v(d) \end{bmatrix} \in R^{v \times d} \quad (5)$$

where $v$ is the number of variables measured for each batch which is the same for different batch records, and $d$ is the batch duration which takes different values for different batches. Each variable $\mathbf{x}_i$, $i = 1,2,\ldots,v$, is assumed to be a random variable with finite variance, which is a reasonable assumption for a process under closed-loop control.

In the SPA method, we monitor the SPs calculated from different batch trajectories. In this work, four groups of batch statistics are included in an SP, as shown below,

$$\mathbf{S} \equiv [\mu | \Sigma | \gamma | \kappa] \in R^{v \times (v+3)} \quad (6)$$

where $\mu \in R^{v \times 1}$ contains the means of all variables

$$\mu = [E(x_i)] = \left[ \frac{1}{d} \sum_{k=1}^{d} x_i(k) \right] \quad (7)$$

$\Sigma \in R^{v \times v}$ is the covariance matrix whose $(i, j)$ entry is the covariance of $\mathbf{x}_i$ and $\mathbf{x}_j$,

$$\Sigma = \left[ cov(\mathbf{x}_i, \mathbf{x}_j) \right] = \left[ \frac{1}{d-1} \sum_{k=1}^{d} (\mathbf{x}_i(k) - \mu_i)(\mathbf{x}_j(k) - \mu_j) \right] \quad (8)$$

$\gamma \in R^{v \times 1}$ contains the skewnesses of all variables, i.e.,

$$\gamma = [\gamma_i] = \left[ \frac{E[(\mathbf{x}_i - \mu_i)^3]}{E[(\mathbf{x}_i - \mu_i)^2]^{3/2}} \right] \quad (9)$$

$\kappa \in R^{v \times 1}$ contains the kurtoses of all variables, i.e.,

$$\kappa = [\kappa_i] = \left[ \frac{E[(\mathbf{x}_i - \mu_i)^4]}{E[(\mathbf{x}_i - \mu_i)^2]^2} - 3 \right] \quad (10)$$

Note that $\mathbf{S}$ can be defined for an entire batch or for a single step. In addition, the selection of batch statistics can be tailored to capture specific process characteristics. For example, higher-order statistics (HOS) in both time domain (cumulants) and frequency domain (polyspectra) can be included to capture the process dynamics, nonlinearity and nonnormality. Although different batches may vary in step and/or batch durations, they all have the same variables and the same number of steps. Therefore, the SP matrix $\mathbf{S}$ obtained from different batch trajectories always have the same dimensions, and no data preprocessing is needed for SP generation.

For a given batch, the final SP used for model building is generated by vectorizing the matrix $\mathbf{S}$ into a row vector, i.e.,

$$SP = \mathrm{vec}(\mathbf{S}) \in R^{1 \times v(v+3)} \quad (11)$$

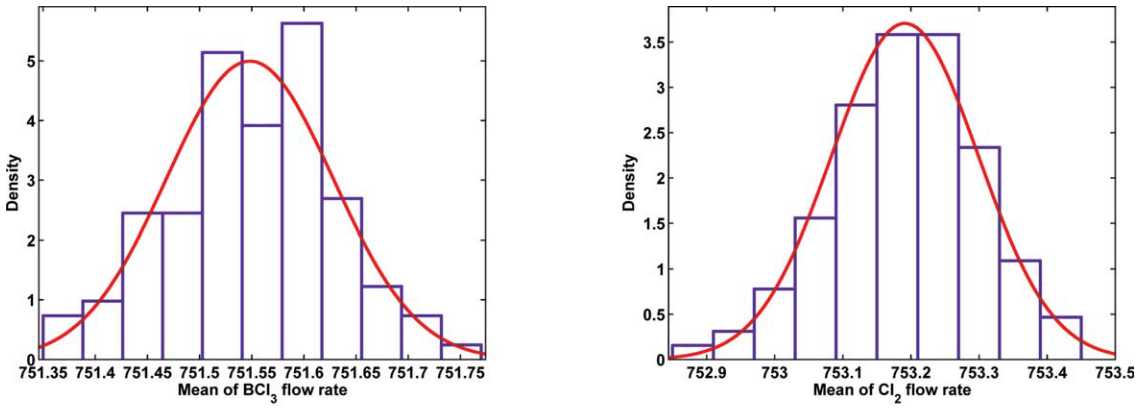
The SPs of normal batches form the rows of the training SP matrix. The length of the SP vector can be reduced because of the symmetry of the covariance matrix $\Sigma$. By including the upper triangular part of $\Sigma$ only, the dimension of the SP vector can be reduced to $v\frac{(v+7)}{2}$. It is worth noting that, as the batch duration $d$ is usually much larger than the number of variables $v$, the size of the final statistic pattern matrix $\left(n \times v\frac{(v+7)}{2}\right)$ is usually smaller compared to the unfolded data matrix used in the MPCA method ($n \times vd$).

### *Dissimilarity quantification and fault detection*

Once the training SP matrix is obtained, the next step is to quantify the dissimilarities among the normal SPs and determine the confidence limit. Distance- or angle- based approaches have been used to quantify the dissimilarities between different objects, which evaluate certain distances or angles between different objects. Most fault detection methods, including PCA, PLS, and FD-kNN, utilize distance- based metrics.

In this work, we use standard PCA to assess the dissimilarity among SPs obtained from different batches. In other words, we perform PCA on the batch statistics matrix obtained from the training samples to determine the dissimilarity confidence limit. To avoid the confusion of the SPA-based fault detection indices with the PCA indices, we use $D_r$ and $D_p$ to denote the SPE and $T^2$ for batch statistics in the SPA method. A batch is considered normal if its dissimilarity indices are below the confidence limits, i.e., $D_r \leq \delta_\alpha^2$ and $D_p \leq T_\alpha^2$, where $\delta_\alpha^2$ and $T_\alpha^2$ denote the confidence limits for dissimilarity index in the residual subspace and the principal component subspace with a confidence level $\alpha$.

Similar to the standard PCA, $\delta_\alpha^2$, and $T_\alpha^2$ can be obtained in two ways. One way is to compute the confidence limits based on the assumed Gaussian distributions, and the analytical expressions for $\delta_\alpha^2$ and $T_\alpha^2$ are available in Refs. 38, 39. The other way, which is more common in industrial applications, is an empirical method based on the calibration or validation data under normal operation conditions.[11,40] For example, a 95% confidence limit can be determined as the $D_r$ or $D_p$ value below which 95% of the calibration samples are located. As the theoretical limits ($\delta_\alpha^2$, $T_\alpha^2$) are influenced by the choice of number of PCs, they may not provide fair comparison among different methods. Therefore, in this work, we use the empirical method to determine the confidence limits so that different methods can be compared using the limits obtained based on the same confidence level.




(a) (b)

**Figure 5. Histograms of two batch statistics across all normal batches (a) Batch mean of $BCl_3$ flow rate; (b) batch mean of $Cl_2$ flow rate.**

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

### *Properties of SPA*

In the SPA framework, PCA is applied to quantify the dissimilarity among SPs from different batches. Therefore, all the fundamental assumptions that the PCA-based method requires apply in the SPA framework. However, one significant distinction between SPA and PCA is that the variables to be monitored in SPA are batch statistics. We claim that the batch statistics satisfy the multivariate Gaussian distribution much better than the process variables in either the original space or in their corresponding principal component subspaces (i.e., the scores), which enables the superior monitoring performance from the SPA method compared to the MPCA-based method. This is verified using the Lilliefors normality test for the industry case study in the next section, and in the following we discuss it in a more general way.

The observation that the distribution of batch statistics is asymptotically Gaussian can be explained through the use of the central limit theorem (CLT). The classical CLT states that (from page 214 of Ref. 44):

"Whatever be the distributions of the independent variables $\xi_v$—subject to certain very general conditions—the sum $\xi = \xi_1 + \cdots + \xi_n$ is asymptotically normal $(m, \sigma)$, where

$$m = m_1 + m_2 + \cdots + m_n \tag{12}$$

$$\sigma^2 = \sigma_1^2 + \sigma_2^2 + \cdots + \sigma_n^2 \tag{13}$$

$m_v$ and $\sigma_v$ are the mean and standard deviation of $\xi_v$."

Many extensions of the classical CLT have been established. The ones relevant to this work are the extensions under weak dependence, which relax the requirement on the independency among different random variables. Specifically, in Ref. 45 it was shown that the CLT applies to sums of bounded random variables generated from stationary dynamic systems; more recently in Ref. 46 the weak dependence condition for stationary dynamic processes was further relaxed to satisfying certain projection criteria.

In this work, different measurements along a batch trajectory are viewed as random variables with weak dependence. Due to closed-loop operation during each batch, batch trajectories are generally bounded, stationary sequences and it is reasonable to assume that the time-series measurements in a batch trajectory satisfy the weak dependence condition. As a batch statistic is simply the mean of a variable (or a function of variables) computed from the measurements contained in the batch record, it follows from the CLT that the distribution of batch statistics is asymptotically Gaussian. This is verified using the industrial benchmark example in the next section.

Second, the statistics calculated from different batches also satisfy the assumption of independent samples. Because of the weak process dynamics of semiconductor processes, it is reasonable to assume that different batches are independent from each other. Note that in the MPCA method, the same assumption is made as well. Then the only question that remains is whether the statistics calculated from different batches follow the same Gaussian distribution, i.e., whether different samples are identically distributed. As different batches have different durations, one may argue that the distributions of statistics calculated from different batches are not identical. However, a closer examination show that the statistics from different batches approximately follow the same Gaussian distribution, which is determined by their similar means and variances (as a Gaussian distribution is fully specified with the mean and variance). More specifically, the averaging effect involved in calculating the statistics assures that the effect of the different batch duration is insignificant, especially for relatively long batch durations.

The above discussion suggests that statistics calculated from different batches are approximately i.i.d. random variables that follow Gaussian distributions. To verify this, in Figure 5 we plot the histograms of two batch statistics (the means of $BCl_3$ and $Cl_2$ flow rates during Step 5) from all normal batches in the industrial etch data set. Similar to Figure 3, the pdfs estimated based on a Gaussian distribution are also plotted. By comparing Figures 3 and 5, we can see that the mean of a variable complies with a Gaussian distribution significantly better than the variable itself. Consequently, when PCA is applied to the batch statistics, the hyperellipsoid that covers the normal operation region will be significantly smaller in volume compared to that generated based on the original process variables. This is verified

**Table 2. Process Variables Used for Monitoring**

| | | | |
|---|---|---|---|
| 1 | $BCl_3$ flow | 11 | RF Pwr |
| 2 | $Cl_2$ flow | 12 | RF impedance |
| 3 | RF Btm Pwr | 13 | TCP tuner |
| 4 | RF Btm Rfl Pwr | 14 | TCP phase err |
| 5 | Endpt Al | 15 | TCP impedance |
| 6 | He press | 16 | TCP top Pwr |
| 7 | Pressure | 17 | TCP Rfl Pwr |
| 8 | RF tuner | 18 | TCP load |
| 9 | RF load | 19 | Vat valve |
| 10 | RF phase err | | |

"Btm", "bottom"; "Pwr", "power"; "Rfl", "reflected"; "Endpt", "endpoint".

in the next section. Therefore, monitoring batch statistics is a more robust and reliable way for fault detection compared with monitoring process variables. This is one of the reasons that we expect the SPA method to perform better than the MPCA method.

Another desirable property of the proposed SPA framework is its flexibility, as different batch statistics can be easily added to capture different process characteristics. In this work, in addition to monitoring the mean and covariance of different variables, the skewness and kurtosis, two of the HOS, are monitored to detect process abnormality. Including these HOS will enhance the monitoring performance as illustrated in the industrial case study. As most batch process variables do not follow Gaussian distributions, nonnormality is a common characteristics of semiconductor processes and should be accounted for in process monitoring. It is worth noting that nonnormality can only be captured by statistics with higher orders, and cannot be captured by the mean or the variance–covariance of the process variables that the MPCA method is based upon. Therefore, the SPA framework provides the capability and flexibility to capture different process characteristics that cannot be captured by MPCA, which is another reason that we expect SPA to perform better. Note that despite the non-Gaussian distribution of the original variables, the batch statistics, including skewness and kurtosis that quantify the nonnormality of the original variables, follow Gaussian distributions due to the CLT. In this work, statistics that capture process dynamics are not included in the SP, mainly because semiconductor processes show weak dynamics. For other processes with strong dynamics, batch statistics such as auto-covariance, cross-covariance, and other higher-order cumulants can be included to capture process dynamics.

## An Industrial Case Study

In this section, we use a benchmark industrial data set to verify the points discussed in the previous sections and to demonstrate the superior performance of the proposed SPA method. Three fault detection methods are compared in this case study: MPCA, FD-kNN, and SPA. To illustrate the effect of data preprocessing on the distribution of process data, as well as on the fault detection performance, four different levels of data preprocessing are compared. They are denoted by (1) original data set; (2) simple cut; (3) dynamic time warping (DTW); and (4) DTW with trajectory mean shift. More detailed information on the data set, data preprocessing, and procedures of fault detection comparison are presented below.

### *Process information*

The data set has been introduced earlier in this article to illustrate the characteristics of semiconductor processes. The data set was collected from an aluminium stack etch process performed on a Lam 9600 plasma etch tool at Texas Instrument.[11,47] The standard process recipe consists of six steps but only two etch steps (4 and 5) are monitored separately in this case study. The whole data set consists of 108 normal batches processed during three experiments which were several weeks apart, and 21 batches with intentionally induced faults processed during the same experiments. Due to the equipment state drifting and maintenance events that reset the equipment states, there are significant differences in variables' means and variances among different groups, as shown in Figure 2.

Similar to Wise et al.,[11] we eliminate two batch records (one normal batch and one fault batch) that have large amount of missing data, and use only 107 normal batches and 20 faulty batches in this study. The batch records were collected by the machine state sensor system at 1-s interval during the etch process. The original data set contains 40 variables including process setpoints, measured variables, and controlled variables such as gas flow rates, chamber pressure and RF power. It has been shown that including irrelevant variables in the analysis degrades the performance of both the MPCA and FD-kNN methods. Therefore, in this work only 19 nonsetpoint process variables are used for fault detection as listed in Table 2, which is the same as in Ref. 11. The information on the faulty runs is listed in Table 3. More detailed description on the faults can be found in Ref. 11.

## Data Preprocessing

Data preprocessing is an important aspect of multivariate statistical analysis and can have significant impacts, for both continuous and batch processes, on the overall sensitivity and robustness of the process monitoring methods.[11] In this work, to gain a comprehensive understanding of the different representative methods and to examine the impact of data preprocessing on the performance of different methods, we perform the following four different levels of data preprocessing. As both Steps 4 and 5 are monitored, the data preprocessing steps listed below are performed for data collected from each step separately.

**Table 3. Induced Faults**

| | | | |
|---|---|---|---|
| 1 | TCP +50 | 11 | $Cl_2$ +5 |
| 2 | RF −12 | 12 | $BCl_3$ −5 |
| 3 | RF +10 | 13 | Pr +2 |
| 4 | Pr +3 | 14 | TCP −20 |
| 5 | TCP +10 | 15 | TCP −15 |
| 6 | $BCl_3$ +5 | 16 | $Cl_2$ −10 |
| 7 | Pr −2 | 17 | RF −12 |
| 8 | $Cl_2$ −5 | 18 | $BCl_3$ +10 |
| 9 | He Chuck | 19 | Pr +1 |
| 10 | TCP +30 | 20 | TCP +20 |

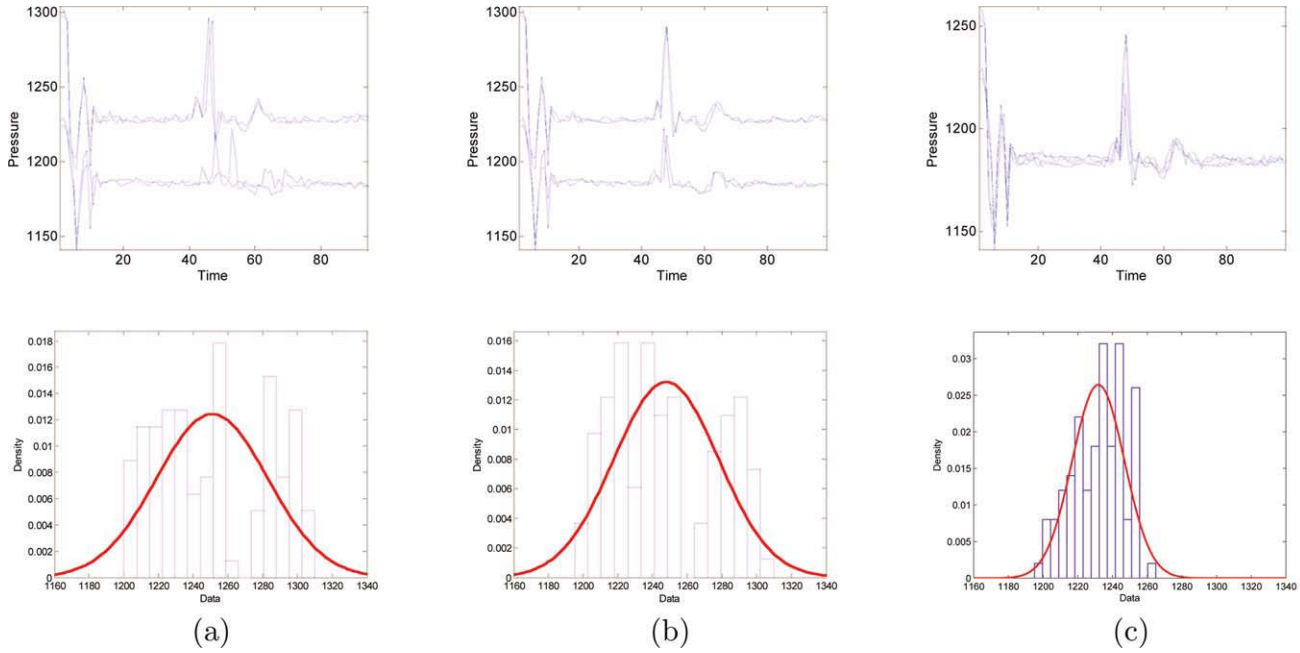



**Figure 6. Illustration of batch trajectories and variable distribution after different levels of data preprocessing (a) after simple cut; (b) after dynamic time warping; (c) after warping and mean shift.**

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

(I) Original data without preprocessing. Different batches have unequal length, unsynchronized trajectories, and follow a multimodal distribution. In this case, only the proposed SPA method is applicable.

(II) Simple cut to make the step trajectories with equal length by removing the last few measurements of a step to match the shortest step trajectory.

(III) (DTW) to synchronize different trajectories by stretching or compressing step trajectories to match a reference step trajectory.

(IV) DTW and trajectory mean shift to make trajectories synchronized with equal length, and follow a unimodal distribution.

One example is given in Figure 6 to illustrate the effect of different data preprocessing steps, where the pressure trajectories of four different batches are plotted after each level of data preprocessing. The histograms of the measured pressure (at sample index 45) after each level of data preprocessing are also given in Figure 6, together with the estimated pdfs based on the assumed Gaussian distributions. It is clear that different levels of data preprocessing progressively improve the normality of the process data.

### *Fault detection performance comparison*

Three fault detection methods (MPCA, FD-kNN, and SPA) are applied to differently preprocessed data sets as described in the previous section. To eliminate the possible bias introduced by any specific training data set, we follow the same procedure as in Refs. 18, 19. For all three methods 200 Monte-Carlo (MC) simulations are performed for confidence limit determination and another 200 MC simulations

**Table 4. Detection Percentage of Three Methods on Differently Preprocessed Data Sets (%)**

| Fault | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MPCA | | | | | | | | | | | | | | | | | | | | |
| I | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| II | 100 | 99 | 0 | 100 | 0 | 0 | 100 | 0 | 0 | 100 | 0 | 100 | 100 | 100 | 100 | 100 | 97 | 100 | 100 | 100 |
| III | 100 | 0 | 0 | 100 | 0 | 0 | 100 | 0 | 0 | 100 | 0 | 100 | 100 | 100 | 77 | 92 | 0 | 100 | 100 | 100 |
| IV | 100 | 0 | 0 | 100 | 0 | 0 | 100 | 0 | 0 | 100 | 0 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| FD-kNN | | | | | | | | | | | | | | | | | | | | |
| I | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| II | 100 | 100 | 0 | 100 | 0 | 0 | 100 | 100 | 0 | 100 | 0 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| III | 100 | 49 | 0 | 100 | 0 | 1 | 100 | 100 | 0 | 100 | 0 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| IV | 100 | 39 | 100 | 100 | 1 | 0 | 100 | 100 | 0 | 100 | 0 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| SPA | | | | | | | | | | | | | | | | | | | | |
| I | 100 | 98 | 100 | 100 | 0 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| II | 100 | 100 | 100 | 100 | 0 | 22 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| III | 100 | 99 | 100 | 100 | 0 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 81 |
| IV | 100 | 100 | 100 | 100 | 51 | 100 | 100 | 100 | 85 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |

"I" denotes raw data; "II" denotes data after simple cut; "III" denotes data after alignment; "IV" denotes data after alignment and mean shift.

**Table 5. Number of Faults Detected by Three Methods on Differently Preprocessed Data Sets**

| | Original | Simple Cut | Warping | Warping and Shift |
|---|---|---|---|---|
| MPCA | – | 14 | 10 | 13 |
| FD-kNN | – | 15 | 14 | 15 |
| SPA | 19 | 18 | 18 | 18 |

are performed for fault detection. Because Steps 4 and 5 are monitored separately, a batch is declared abnormal if a fault is detected in either one of these two steps. During model building, the confidence limit for each step is determined empirically. In each MC simulation, 97 normal batches are randomly selected as training samples, and the rest 10 batches are used as validation samples. Then we pool all the validation results from 200 MC runs together, i.e., 2000 calibration samples (10 calibration per MC), to determine the confidence limit based on a 95% confidence level. Note that the same procedure is followed for all the three fault detection methods. After the confidence limit is determined for each method, additional 200 MC simulations are performed to test the fault detection performance. In each MC test run, the model is built based on 97 randomly selected training samples, then 20 faulty batches are tested for fault detection using the predetermined confidence limits.

The performance of the different methods are reported in Table 4 using the percentage of the fault being detected among the 200 MC test runs. For MPCA and SPA, SPE, and $D_r$ are selected as the fault detection indices because they show better performance compared to $T^2$ and $D_p$ in cross-validation. By using 95% as the cut-off rate of successful detection, the number of faults detected by different methods on differently preprocessed data sets are listed in Table 5. Both Tables 4 and 5 show that among the three methods, the SPA method performs the best for all levels of data preprocessing. In addition, by comparing the performance of SPA with itself when applied to differently preprocessed data sets, we observe that SPA performs the best on the original data set without any preprocessing, where only one of 20 faults was missed. The number of missed fault increases to two when SPA is applied to preprocessed data sets. Another observation is that for this case study, data preprocessing steps also deteriorate fault detection performance for the MPCA and FD-kNN methods, as the MPCA and FD-kNN methods based on the simple-cut data set perform at least as good as on other differently preprocessed data sets. This observation suggests that there could be information loss or distortion caused by data manipulation during preprocessing. This effect has also been observed by other researchers.[43,48]

### *Further comparison on MPCA and SPA*

In the previous section, we discussed the fundamental reasons for the improved monitoring performance provided by the SPA method. In this subsection, we use the industrial case study to verify them. Specifically, we first compare the normality of batch statistics with that of the process variables, then we examine to what extent the HOS contribute to the fault detection performance.

### *Normality test of batch statistics and process variables*

To quantity the effects of different data preprocessing steps, we evaluate the normality of the variables obtained after different levels of preprocessing, and compare the normality of process variables with that of batch statistics. Note that, for MPCA, the variables are not the 19 original process variables but the unfolded variables [i.e., 19 $\times$ (step duration) variables]. To evaluate the normality of different variables, we compare the empirical distribution of a variable to the Gaussian distribution with the same mean and variance as the empirical distribution, and use the Lilliefors test[49] as a goodness-of-fit index to measure the departure of the empirical distribution from Gaussian distribution. The Lilliefors test is similar to the Kolmogorov-Smirnov test but it adjusts for the fact that the parameters of the Gaussian distribution are estimated from the data set rather than specified in advance. The percentages of variables that pass the Lilliefors test at 1% significant level are listed in Table 6 for both unfolded process variables (data used in MPCA) and batch statistics (data used in SPA). For unfolded process variables, Table 6 indicates that various preprocessing steps do improve the data normality progressively, as the percentage of the process variables that pass the Lilliefors test keeps increasing with additional preprocessing steps. However, even with both trajectory alignment and mean shift, the process variables do not satisfy the assumption of Gaussian distribution well, as only $<50\%$ of the variables pass the Lilliefors test. On the other hand for batch statistics, about 80% of statistics pass the Lilliefors test without any preprocessing, indicating that batch statistics comply to Gaussian distributions significantly better. Further data preprocessing does not improve the normality of the batch statistics substantially.

It is worth noting that how well the data satisfy the fundamental assumptions of the MPCA-based fault detection method has significant impact on the fault detection performance. This is because when MPCA is applied to evaluate the dissimilarity among different batches, it determines the confidence limit of SPE based on the assumption that the unfolded variables of the normal batches follow a multivariate Gaussian distribution and it determines the confidence limit of $T^2$ based on the assumption that the scores of the normal batches follow a multivariate Gaussian distribution. Here, we use the $T^2$ statistic to illustrate this point. If the scores are not Gaussian distributed, for a given confidence level, the ''normal'' hyperellipsoid determined by MPCA will be enlarged and cover part of abnormal operation region which will result in increased Type II error (missed detection). On the other hand, if the ''normal'' hyperellipsoid is shrunk to reduced missed detection, it will miss part of the normal operation region and results in elevated false alarms. This point is illustrated in Figure 7, which compares the

**Table 6. Percentage of Variables that Pass the Lilliefors Normality Test**

| | Original/ Simple Cut | | Warping | | Warping and Shift | |
|---|---|---|---|---|---|---|
| | Step 4 | Step 5 | Step 4 | Step 5 | Step 4 | Step 5 |
| Unfolded variables | 38.3 | 32.1 | 40.3 | 36.8 | 45.6 | 45.0 |
| Batch statistics | 83.7 | 78.5 | 82.0 | 76.3 | 86.0 | 79.0 |

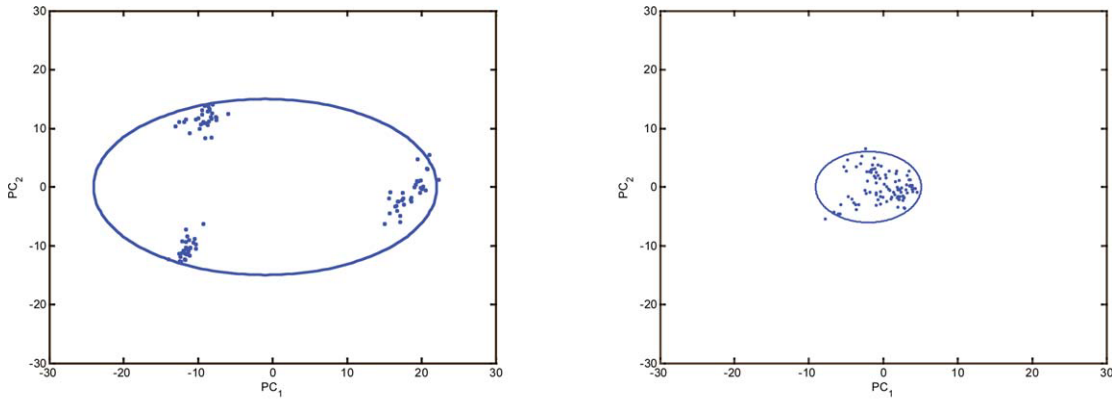



**Figure 7. Comparison of score plots: (a) MPCA; (b) SPA. The ellipsoid represents the confidence limit of $T^2$ in MPCA and $D_p^2$ in SPA.**

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

score plots of the normal batches in the subspace spanned by the first two PCs from MPCA and SPA. The result from MPCA is based on the simple-cut data and that from SPA is based on the raw data. The scales in two score plots are comparable because the data matrices are autoscaled before MPCA is applied in both methods. Figure 7 indicates that for the MPCA - based method, the better the scores of normal batches follow a multivariate Gaussian distribution, the better the fault detection performance of $T^2$, which is also true for the unfolded variables when SPE is applied for fault detection. This is one of the reasons that SPA performs better than MPCA.

### *Contributions of different batch statistics*

In this subsection, we examine the contributions of each batch statistics to the detection of different faults, with emphasis on the faults that are not detected by MPCA and FD-kNN. To evaluate the contributions of each statistic, we break down the fault detection index $D_r$ into components that correspond to each batch statistic, in a similar way that the contribution plots are generated in the traditional MPCA method,[50,51] i.e.,

$$D_r = ||\widetilde{\mathrm{SP}}||^2 = \sum_{k=1}^{v\frac{(v+7)}{2}} D_{r,k}^2 = \sum_{k=1}^{v\frac{(v+7)}{2}} \left(\mathrm{SP}_k - \widehat{\mathrm{SP}}_k\right)^2 \qquad (14)$$

where $\mathrm{SP}_k$ denotes the $k$-th element in a SP.

We categorize all batch statistics used in this work into three groups: mean (first-order statistics), Covariance (second-order statistics), and HOS (skewness and kurtosis). Figure 8 shows the contributions from different batch statistics groups to $D_r$ for both Steps 4 and Step 5. From Figure 8, we observe that in general HOS contribute substantially to the fault detection index, which also indicates that the data are not Gaussian distributed. Because of its significance, nonnormality should be accounted for in semiconductor process monitoring. In addition, Figure 8 shows that the faults that are easily detected by the MPCA and FD-kNN methods usually have large mean shifts (in either one of the steps or both steps), such as Faults 1, 4, 7, 12, and 13, and we term them Group A faults. On the other hand, the faults with small mean shifts are usually difficult to be detected by the MPCA and FD-kNN methods, including Faults 2, 3, 6, and 9, which we term as Group B faults. For Group B faults, the SPA method was able to detect all of them except Fault 5. In Figure 9, we compare the contribution from variable means between two groups of faults, as well as the ratio of the contributions from the HOS to those from the mean. Figure 9 indicates that the HOS contribute significantly for the SPA method to detect group B faults, which is another reason that SPA outperforms the MPCA.

## Conclusions and Discussions

One major challenge in semiconductor process monitoring is the extensive and often off-line data preprocessing required by the traditional SPM and PCM methods, which

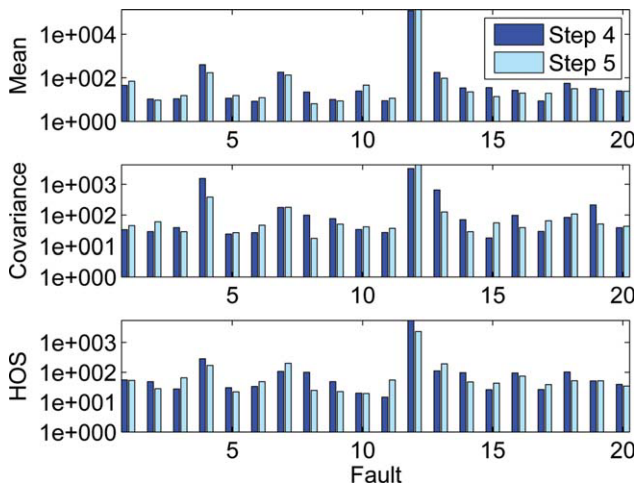



**Figure 8. Contributions from different statistics groups (Steps 4 and 5).**

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

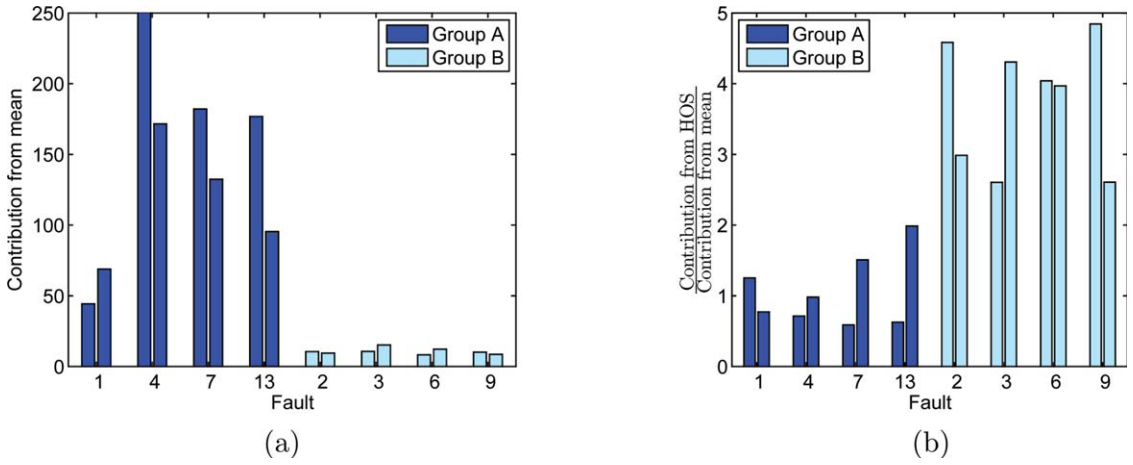



**Figure 9. Comparison of different faults in groups A and B.**

(a) Contribution from the variable means; (b) ratio of contributions from the HOS to those from the variable means. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

makes model building and maintenance extremely labor intensive due to the large number of models in a typical semiconductor fab. Another major challenge is that semiconductor process data often show more severe nonnormality compared to those of the traditional chemical processes under closed-loop control, which results in suboptimal performance of the traditional SPM and PCM methods in many applications. To address these challenges, we propose a new process monitoring framework termed SPA. SPA monitors batch statistics to perform fault detection, which eliminates all data preprocessing steps. In addition, SPA provides superior fault detection performance compared to the MPCA and FD-kNN methods.

It is shown that one of the fundamental reasons for the limited success of the MPCA and FD-kNN methods in monitoring semiconductor processes is that the process data violate the basic assumptions required by these methods. Due to semiconductor process characteristics, the unfolded process variables usually are not identically distributed random variables, and do not follow Gaussian distributions even with various data preprocessing steps. To address this limitation, in the proposed SPA framework, various batch statistics are computed from batch trajectories to characterize process behavior, and PCA is applied to quantify the dissimilarity among batch statistics computed from different batches. It follows from the CLT that the distribution of batch statistics is asymptotically Gaussian, which explains why SPA performs better than MPCA and FD-kNN.

In addition, MPCA is a second-order method, which lacks the capability of providing higher-order representations for non-Gaussian data. To address this limitation, SPA employs various batch statistics, including HOS, to capture process characteristics, which provides flexibility that MPCA and FD-kNN do not offer. For example, in this work, HOS such as skewness and kurtosis are included in the SP to quantify process nonnormality. As the process nonnormality cannot be captured by variable means or covariances, MPCA and FD-kNN are not able to make use of this important process information. In contrast, SPA can make use of this additional information which reveals important process properties, and is able to detect the faults that MPCA and FD-kNN cannot detect.

It is worth noting that the idea of using HOS for fault detection is not new. In fact, it was a popular research topic in the 1990s for fault detection in mechanical systems via spectrum analysis/signal processing.[52,53] In addition, the fault detection methods that implement artificial neural network[54,55] or independent component analysis[17,56] all make use of HOS implicitly. However, the idea of monitoring the batch statistics, in which HOS only play a partial role and are monitored explicitly, is new and was never presented before.

It is also worth noting that although the SPA framework is proposed to address the challenges associated with semiconductor process monitoring, its application is not limited to semiconductor batch processes. Instead, it can be applied to various batch processes, as well as continuous processes. Specifically, in Ref. 57 it is shown that the SPA method outperforms the PCA and Dynamic principal component analysis (DPCA) methods significantly in monitoring continuous chemical processes when the process data and the scores in the principal component subspace do not follow multivariate Gaussian distributions, and several cases studies including the benchmark Tennessee Eastman process are given to illustrate the performance of the proposed SPA method.

Finally in the following, we discuss some practical issues and possible limitations associated with SPA. First, if batch trajectories follow multimodal distributions, most likely their variable means will follow multimodal distributions as well. This is part of the reason that about 20% of the batch statistics in the industrial example do not pass the normality test. However, second-order statistics and HOS do not suffer from the multimodal distributions, because variable means are subtracted in calculating them. In addition, even for variable means, they comply to Gaussian distribution much better than the original variables due to the averaging effect, which is illustrated in Figure 3 and 5.

Second, in general, the number of measurements collected during a step in a semiconductor process is large enough to obtain good statistics that capture the step characteristics. However, there are cases that a step is so short that only few

measurements are collected. In this case, the statistics for the step may not be reliable, which could lead to increased Type I and Type II errors.

## Acknowledgments

Financial support from NSF is gratefully acknowledged by QPH under grant CBET-0853748 and JW under grant CBET-0853983. In addition, the authors thank Dr. Dmitry Glotov for helpful discussions.

## Literature Cited

1. Nomikos P, MacGregor JF. Monitoring of batch processes using multiway principal component. *AIChE J*. 1994;40:1361–1375.
2. Nomikos P, MacGregor JF. Multivariate SPC charts for monitoring batch processes. *Technometrics*. 1995;37:41–59.
3. MacGregor JF, Kourti T. Statistical process control of multivariate processes. *Contr Eng Pract*. 1995;3:403–414.
4. Kourti T, MacGregor JF. Multivariate SPC methods for process and product monitoring. *J Qual Tech*. 1996;28:409–428.
5. Westerhuis JA, Kourti T, MacGregor JF. Comparing alternative approaches for multivariate statistical analysis of batch process data. *J Chemometr*. 1999;13:397–413.
6. Kourti T. Multivariate dynamic data modeling for analysis and statistical process control of batch processes, start-ups and grade transitions. *J Chemometr*. 2003;17:93–109.
7. Kourti T, Nomikos P, MacGregor JF. Analysis, monitoring, and fault diagnosis of batch processes using multi-block and multi-way PLS. *J Proc Cont*. 1995;5:277–284.
8. Nomikos P, MacGregor JF. Multi-way partial least squares in monitoring batch processes. *Chemometr Intell Lab Syst*. 1995;30:97–108.
9. Qin SJ, Valle-Cervantes S, Piovoso M. On unifying multi-block analysis with applications to decentralized process monitoring. *J Chemometr*. 2001;15:715–742.
10. Qin SJ, Cherry G, Good R, Wang J, Harrison CA. Semiconductor manufacturing process control and monitoring: a fab-wide framework. *J Proc Cont*. 2006;16:179–191.
11. Wise BM, Gallagher NB, Butler SW, White D, Barna GG. A comparison of principal component analysis, multiway principal component analysis, trilinear decomposition and parallel factor analysis for fault detection in a semiconductor etch process. *J Chemometr*. 1999;13:379–396.
12. Wise BM, Gallagher NB, Martin EB. Application of PARAFAC2 to fault detection and diagnosis in semiconductor etch. *J Chemometr*. 2001;15:285–298.
13. Yue H, Qin SJ, Wiseman J, Toprac A. Plasma etching endpoint detection using multiple wavelengths for small open area wafers. *J Vac Sci Technol*. 2001;19:66–75.
14. Yue H, Qin SJ, Markle R, Nauert C, Gatto M. Fault detection of plasma etchers using optical emission spectra. *IEEE Trans Semicond Manuf*. 2000;13:374–385.
15. Doh HH, Wang P, Xu N, Yadav P, Magni E, McMillin B. Application of PCA/PLS to optical emission spectra for plasma etch process monitoring. In: *AEC/APC Symposium XV*. Colorado Springs, CO, 2003.
16. Yue HH, Tomoyasu M. Weighted principal component analysis and its applications to improve FDC performance. In: *43rd IEEE Conference on Decision and Control*. Institute of Electrical and Electronics Engineers, Atlantis, Paradise Island, Bahamas, 2004;4262–4267.
17. Lee J, Qin SJ, Lee IB. Fault detection and diagnosis based on modified independent component analysis. *AIChE J*. 2006;52:3501–3514.
18. He QP, Wang J. Fault detection using k-nearest-neighbor rule for semiconductor manufacturing processes. *IEEE Trans Semicond Manuf*. 2007;20:345–354.
19. He QP, Wang J. Large-scale semiconductor process monitoring using a fast pattern recognition based method. *IEEE Trans Semicond Manuf*. 2010; in press.
20. Yu J, Qin SJ. Variance component analysis based fault diagnosis of multi-layer overlay lithography processes. *IIE Trans*. 2009;41:764–775.
21. Cherry G, Good R, Qin SJ. Semiconductor process monitoring and fault detection with recursive multiway PCA based on a combined index. In: *AEC/APC Symposium XIV*. Salt Lake City, UT, 2002.
22. Cherry G, Qin SJ. Multiblock principal component analysis based on a combined index for semiconductor fault detection and diagnosis. *IEEE Trans Semicond Manuf*. 2006;19:159–171.
23. Samata S, Ushiku Y, Ishii K, Tanaka M, Furuhata T, Nakao T, Yamamoto A. Fault detection method for dry vacuum pump of LPCVD system. In: *AEC/APC Symposium XIV*. Snowbird, UT, 2002.
24. Wong J. Batch PLS analysis and FDC process control of within lot SiON gate oxide thickness variation in sub-nanometer range. In: *AEC/APC Symposium XVIII*. Westminster, CO, 2006.
25. Adamson T, Moore G, Passow M, Wong J, Xu Y. Strategies for successfully implementing fab-wide FDC methodologies in semiconductor manufacturing. In: *AEC/APC Symposium XVIII*. Westminster, CO, 2006.
26. Moore T, Harner B, Kestner G, Baab C, Stanchfield J. Intel's FDC proliferation in 300 mm HVM: progress and lessons learned. In: *AEC/APC Symposium XVIII*. Westminster, CO, 2006.
27. He QP, Wang J. A multivariate fault detection method using k-nearest-neighbor rule. In: *Proceedings of AEC/APC Symposium XVIII*. Westminster, CO, 2006.
28. Chamness K. Multivariate fault detection and visualization in the semiconductor industry. PhD Thesis, The University of Texas at Austin, Austin, Tx, 2006.
29. Chamness K, Edgar TF. Local-kNN fault detection algorithm description and examples. In: *TWMCC Spring 2006 Conference*, Austin, Tx, 2007.
30. He QP, Wang J. Principal component based k-nearest-neighbor rule for semiconductor process fault detection. In: *Proceedings of the 2008 American Control Conference*. Institute of Electrical and Electronics Engineers, Seattle, WA, 2008;1606–1611.
31. He QP. Novel multivariate fault detection methods using Mahalanobis distance. In: *AEC/APC Symposium XVII*. Indian Wells, CA, 2005.
32. Stutts T. Method for calculating variable influences on the Mahalanobis distance. In: *AEC/APC Symposium XX*. Salt Lake City, UT, 2008.
33. Lacaille J, Zagrebnov M. An unsupervised diagnosis for process tool fault detection: the flexible golden pattern. *IEEE Trans Semicond Manuf*. 2007;20:355–363.
34. Smilde AK, Doornbos DA. Three way methods for the calibration of chromatographic systems: comparing PARAFAC and three-way PLS. *J Chemometr*. 1991;5:345–360.
35. Smilde AK, Wang Y, Kowalski BR. Theory of medium-rank second-order calibration with restricted-Tucker models. *J Chemometr*. 1994;8:21–36.
36. Sanchez E, Kowalski BR. Tensorial resolution: a direct trilinear decomposition. *J Chemometr*. 1990;4:29–45.
37. Wold S, Esbensen K, Geladi P. Principal component analysis. *Chemometr Intell Lab Syst*. 1987;2:37–52.
38. Jackson JE, Mudholkar G. Control procedures for residuals associated with principal component analysis. *Technometrics*. 1979;21:341–349.
39. Qin SJ. Statistical process monitoring: basics and beyond. *J Chemometr*. 2003;17:480–502.
40. Russell EL, Chiang LH, Braatz RD. Fault detection in industrial processes using canonical variate analysis and dynamic principal component analysis. *Chemometr Intell Lab Syst*. 2000;51:81–93.
41. Tracy N, Young J, Mason R. Multivariate control charts for individual observations. *J Qual Tech*. 1992;24:88–95.
42. Kermit M, Tomic O. Independent component analysis applied on gas sensor array measurement data. *IEEE Sensors J*. 2003;3:218–228.
43. Zhang Y, Edgar TF. A robust dynamic time warping algorithm for batch trajectory synchronization. In: *Proceedings of the 2008 American Control Conference*. Institute of Electrical and Electronics Engineers, Seattle, WA, 2008;2864–2869.
44. Cramer H. *Mathematical Methods of Statistics*. Princeton, NJ: Princeton University Press, 1946.
45. Pene F. Rate of convergence in the multidimensional central limit theorem for stationary processes. application to the Knudsen gas and to the Sinai billiard. *Ann Appl Probab*. 2005;15:2331–2392.

46. Dedecker J, Rio E. On mean central limit theorems for stationary sequences. *Ann Inst H Poincar Probab Statist*. 2008;44:693–726.
47. Wise BM. Metal etch data for fault detection evaluation. Available at http://software.eigenvector.com/Data/Etch/index.html, 1999.
48. Castelo T, Moffitt J, New-Courtney T. Fault detection benefits for metrology equipment. In: *Proceedings of the AEC/APC Symposium XX*. Snowbird, UT, 2008.
49. Conover W. *Practical Nonparametric Statistics*. New York: Wiley, 1980.
50. Kourti T, MacGregor JF. Multivariate SPC methods for monitoring and diagnosing of process performance. In: *Proceedings of PSE*. Elsevier, 1994; 739–746.
51. Miller P, Swanson R, Heckler C. Contribution plots: a missing link in multivariable quality control. *Appl Math Comp Sci*. 1998;8:775–792.
52. Isermann R. Process fault detection based on modeling and estimation methods—a survey. *Automatica*. 1984;20:387–404.
53. Benbouzid M. A review of induction motor signature analysis as a medium for fault detection. *IEEE Trans Ind Electron*. 2000;47:984–993.
54. Watanabe K, Matsuura I, Abe M, Kubota M, Himmelblau DM. Incipient fault diagnosis of chemical processes via artificial neural networks. *AIChE J*. 1989;35:1803–1812.
55. Hoskins JC, Kaliyur KM, Himmelblau DM. Fault diagnosis in complex chemical plants using artificial neural networks. *AIChE J*. 1991;37:137–141.
56. Kano M, Tanaka S, Hasebe S, Hashimoto I, Ohno H. Monitoring independent components for fault detection. *AIChE J*. 2003;49:969–976.
57. Wang J, He QP. Multivariate statistical process monitoring based on statistics pattern analysis. *Inc Eng Chem Res*. In press.

*Manuscript received July 30, 2009, and revision received Feb. 19, 2010.*