JOURNAL OF CHEMOMETRICS
J. Chemometrics **13,** 379–396 (1999)

A COMPARISON OF PRINCIPAL COMPONENT ANALYSIS, MULTIWAY PRINCIPAL COMPONENT ANALYSIS, TRILINEAR DECOMPOSITION AND PARALLEL FACTOR ANALYSIS FOR FAULT DETECTION IN A SEMICONDUCTOR ETCH PROCESS

BARRY M. WISE,[1]* NEAL B. GALLAGHER,[1] STEPHANIE WATTS BUTLER,[2] DANIEL D. WHITE JR[2] AND GABRIEL G. BARNA[2]
[1]*Eigenvector Research, Inc., 830 Wapato Lake Road, Manson, WA 98831, USA*
[2]*Texas Instruments, 13536 North Central Expressway, MS 944, Dallas, TX 75265, USA*

SUMMARY

Multivariate statistical process control (MSPC) tools have been developed for monitoring a Lam 9600 TCP metal etcher at Texas Instruments. These tools are used to determine if the etch process is operating normally or if a system fault has occurred. Application of these methods is complicated because the etch process data exhibit a large amount of normal systematic variation. Variations due to faults of process concern can be relatively minor in comparison. The Lam 9600 used in this study is equipped with several sensor systems including engineering variables (e.g. pressure, gas flow rates and power), spatially resolved optical emission spectroscopy (OES) of the plasma and a radio-frequency monitoring (RFM) system to monitor the power and phase relationships of the plasma generator. A variety of analysis methods and data preprocessing techniques have been tested for their sensitivity to specific system faults. These methods have been applied to data from each of the sensor systems separately and in combination. The performance of the methods on a set of benchmark fault detection problems is presented and the strengths and weaknesses of the methods are discussed, along with the relative advantages of each of the sensor systems. Copyright © 1999 John Wiley & Sons, Ltd.

KEY WORDS: fault detection; measurement selection; multivariate statistical process control; principal component analysis; multiway principal component analysis; trilinear decomposition; parallel factor analysis

1. INTRODUCTION

Semiconductor processes, like many chemical processes, are becoming more measurement-rich all the time. A wide variety of sensors and sensor systems are available. The goal of adding sensors, of course, is to reduce costs and/or improve the final product quality through improved process control or fault detection. Often, however, it is not apparent what sensors will be useful in meeting these goals. For sensors to be useful, they must be sensitive to variations in the process and be stable enough to provide information over extended time periods. In addition, methods used to treat process data must be specified, as they will also impact sensitivity and robustness performance.

Recently, 'chemometric techniques' have been applied to process (as opposed to analytical

* Correspondence to: B. M. Wise, Eigenvector Research, Inc., 830 Wapato Lake Road, Manson, WA 98831, USA.
E-mail: bmw@eigenvector.com

CCC 0886–9383/99/040379–18 $17.50
Received 22 September 1998
Accepted 17 March 1999

chemistry) problems. These applications can be roughly divided between those directed at maintenance of process instruments, e.g. calibration, and those that are concerned with maintenance of the process itself, e.g. statistical process control and fault detection. Our focus will be on the latter area. In this paper we describe a study performed on a Lam 9600 metal etcher to determine which of three sensor systems, alone and in combination, and what data treatment method are the most sensitive to a series of known faults. One of the most often used chemometric techniques, principal component analysis (PCA), will be reviewed, along with an adaptation of PCA for use with three-dimensional arrays, multiway PCA (MPCA). These methods were originally used on these data in Wise *et al*.[1] Trilinear decomposition (TLD) and parallel factor analysis (PARAFAC) will also be considered here. The issue of robustness of the sensors and methods over long periods is discussed in Gallagher *et al*.[2]

2. THE METAL ETCH PROCESS

There are many steps in the manufacture of semiconductors. This project was focused on an Al stack etch process performed on the commercially available Lam 9600 plasma etch tool.[3] The goal of this process is to etch the TiN/A1–0·5% Cu/TiN/oxide stack with an inductively coupled BCl_3/Cl_2 plasma. The key parameters of interest are the linewidth of the etched A1 line (specifically the linewidth reduction in relation to the incoming resist linewidth), uniformity across the wafer and the oxide loss.

The standard recipe for the process consists of a series of six steps. The first two are for gas flow and pressure stabilization. Step 3 is a brief plasma ignition step. Step 4 is the main etch of the A1 layer terminating at the A1 endpoint, with step 5 acting as the over-etch for the underlying TiN and oxide layers. Note that this is a single-chemistry etch process, i.e. the process chemistry is identical during steps 3–5. Step 6 vents the chamber. The process ‘profile’ as indicated by the endpoint A signal (the plasma emission intensity measured by a filter spectrometer) is shown in Figure 1. The stabilization step is followed by the three etch regions: Al, TiN and oxide etch.

3. PROCESS SENSORS

Sensor selection is a primary consideration when planning a fault detection and classification (FDC) system. In the etch process it would be ideal to have sensors which directly reflected the state of the wafers in the process. However, with a few exceptions, wafer state sensors are typically unavailable in original equipment manufacturer (OEM) processing tools. Thus the alternative is to select more commonly available process state sensors, with the understanding that wafer state information will have to be inferred.

The metal etcher used for this study was equipped with three sensor systems: machine state, radio-frequency monitors (RFMs), and optical emission spectroscopy (OES). The machine state sensors, built into the processing tool, collect machine data during wafer processing. The machine data consist of 40 process setpoints and measured and controlled variables sampled at 1 s intervals during the etch. These are engineering variables such as gas flow rates, chamber pressure and RF power. In this work, non-setpoint process variables with some normal variation were used for monitoring, as shown in Table 1. Also, the physics of the problem suggests that these variables should be relevant to process and final product state.

The RFM sensors measure the voltage, current and phase relationships at the fundamental frequency of 13·56 MHz and the next four harmonics at four locations in the RF control system. The resulting 70 values are sampled every 3 s. The presence of each chemical species affects the plasma power and phase relationships in unique ways; thus the RFM sensors provide a surprising amount of chemical information.

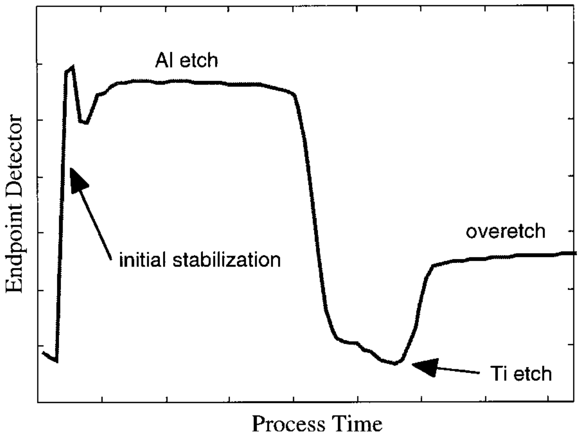


Figure 1. Endpoint trace

The OES is used to monitor the plasma in the range from 245 to 800 nm in three locations above the wafer using fiber optics. The original data consist of 2042 channels per location; however, in this work the data were preprocessed by integrating a much smaller number of peaks (40) in each of the three spectra which correspond to process gases and species evolving from the wafer owing to the etch.

A major objective of this work was to determine which sensors, or combinations of sensors, are most useful for detecting process faults. Data from the three sensors systems were used to develop models of the process in a variety of ways, and the ability of the models to detect faults was tested.

4. PROCESS SHIFTS AND DRIFT

Ideally, under normal conditions a process would be stationary, i.e. retain the same mean and covariance structure over time. Unfortunately, measurements from the etch process are clearly non-stationary. Changes in the data are primarily due to three sources: aging of the etcher over a clean cycle (the period of time between routine maintenance of the machine) as residue accumulates on the inside of the chamber; differences in the incoming materials due to changes in upstream processes; and drift in the process-monitoring sensors themselves. In addition, process maintenance can result in sudden shifts in the mean. The result is that it is normal for the process data to show considerable variation over time. The shift in process mean and covariance is shown graphically in Figure 2. The process mean drifts during a clean cycle, then shifts suddenly during maintenance. The process

Table 1. Machine state variables used for process monitoring

1	BCl_3 flow	8	RF tuner	15	TCP impedance
2	Cl_2 flow	9	RF load	16	TCP top power
3	RF bottom power	10	Phase error	17	TCP reflected power
4	RFB reflected power	11	RF power	18	TCP load
5	Endpoint A detector	12	RF impedance	19	Vat valve
6	Helium pressure	13	TCP tuner		
7	Chamber pressure	14	TCP phase error		

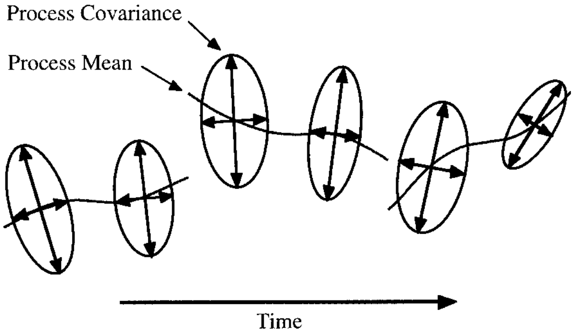


Figure 2. Moving mean and covariance

covariance also drifts, but typically much more slowly. These variations are often much larger than changes due to actual process faults.

5. DATA TREATMENT

Chemical and manufacturing processes are becoming more heavily instrumented and the data are recorded more frequently. This is creating a data overload, and the result is that a good deal of the data are 'wasted', i.e. no useful information is obtained from them. The problem is one of both compression and extraction. Generally, there is a great deal of correlated or redundant information provided by process sensors. This information must be compressed in a manner that retains the essential information and is more easily displayed than each of the process variables individually. Also, often essential information lies not in any individual process variable but in how the variables change with respect to one another, i.e. how they covary. In this case the information must be extracted from the data. Furthermore, in the presence of large amounts of noise it would be desirable to take advantage of some sort of signal averaging.

5.1. Principal component analysis

Principal component analysis (PCA) is a favorite tool of chemometricians for data compression and information extraction.[4–7] PCA finds combinations of variables or *factors* that describe major trends in a data set. Mathematically, PCA relies on an eigenvector decomposition of the covariance or correlation matrix of the process variables. In this work we will use the convention that rows of a data matrix **X** correspond to samples while columns correspond to variables. For a given data matrix **X** with *m* rows and *n* columns the covariance matrix of **X** is defined as

$$\mathrm{cov}(\mathbf{X}) = \frac{\mathbf{X}^{\mathrm{T}}\mathbf{X}}{m-1} \tag{1}$$

This assumes that the columns of **X** have been 'mean centered', i.e. adjusted to have zero mean by subtracting the mean of each column. If the columns of **X** have been 'autoscaled', i.e. adjusted to zero mean and unit variance by dividing each column by its standard deviation, equation (1) gives the correlation matrix of X. (Unless otherwise noted, it is assumed that data are either mean centered or autoscaled prior to analysis.) PCA decomposes the data matrix **X** as the sum of the outer product of vectors $\mathbf{t}_i$ and $\mathbf{p}_i$ plus a residual matrix **E**:

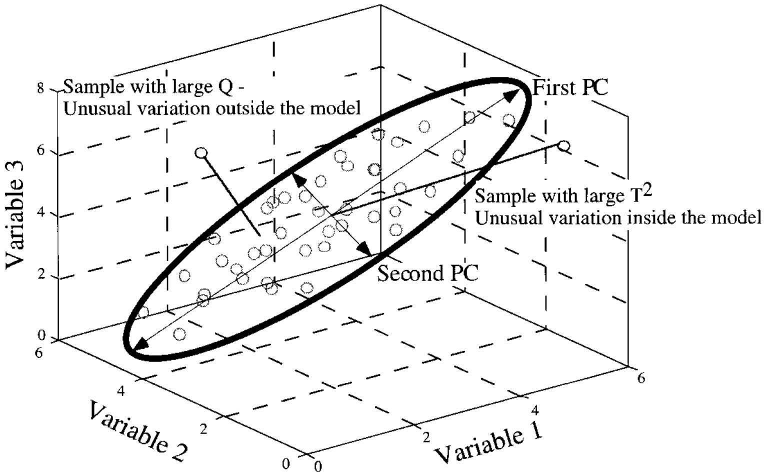


Figure 3. Principal component model of three-dimensional data set lying primarily in a single plane showing Q and T^2 outliers

$$\mathbf{X} = \mathbf{t}_1\mathbf{p}_1^{\mathrm{T}} + \mathbf{t}_2\mathbf{p}_2^{\mathrm{T}} + \ldots + \mathbf{t}_k\mathbf{p}_k^{\mathrm{T}} + \mathbf{E} = \mathbf{T}\mathbf{P}^{\mathrm{T}} + \mathbf{E} \tag{2}$$

Here k must be less than or equal to the smaller dimension of **X**, i.e. $k \leq \min\,[m,n]$. The t_i vectors are known as *scores* and contain information on how the *samples* relate to each other. The $\mathbf{p}_i$ vectors are *eigenvectors* of the covariance matrix, i.e. for each $\mathbf{p}_i$

$$\mathrm{cov}(\mathbf{X})\mathbf{p}_i = \lambda_i\mathbf{p}_i \tag{3}$$

where λ_i is the *eigenvalue* associated with the eigenvector $\mathbf{p}_i$. In PCA the $\mathbf{p}_i$ are known as *loadings* and contain information on how *variables* relate to each other. The $\mathbf{t}_i$ form an orthogonal set ($\mathrm{t}_i^{\mathrm{T}}\mathrm{t}_j = 0$ for $i \neq j$), while the $\mathbf{p}_i$ are orthonormal ($\mathbf{p}_i^{\mathrm{T}}\mathbf{p}_j = 0$ for $i \neq j, \mathbf{p}_i^{\mathrm{T}}\mathbf{p}_j = 1$ for $i = j$). Note that for **X** and any $\mathbf{t}_i$,$\mathbf{p}_i$ pair

$$\mathbf{X}\mathbf{p}_i = \mathbf{t}_i \tag{4}$$

This is because the score vector $\mathbf{t}_i$ is the linear combination of the original **X** data defined by $\mathbf{p}_i$. The $\mathbf{t}_i$, $\mathbf{p}_i$ pairs are arranged in descending order according to the associated λ_i. The λ_i are a measure of the amount of *variance* described by the $\mathbf{t}_i$, $\mathbf{p}_i$ pair. In this context we can think of variance as *information*. Because the $\mathbf{t}_i$, $\mathbf{p}_i$ pairs are in descending order of λ_i, the first pair capture:the largest amount of information of any pair in the decomposition. In fact, it can be shown that the $\mathbf{t}_1$, $\mathbf{p}_1$ pair captures the greatest amount of variation in the data that it is possible to capture with a linear factor. Subsequent pairs capture the greatest possible variance remaining at that step.

The concept of principal components (PCs) is shown graphically in Figure 3. The figure shows a three-dimensional data set where the data lie primarily in a plane; thus the data are well described by a two-PC model. The first eigenvector or PC aligns with the greatest variation in the data, while the second PC aligns with the greatest amount of variation that is orthogonal to the first PC. Generally it is found that the data can be adequately described using far fewer principal components than original variables, i.e. $k \ll n$.

It is also possible to calculate a lack of model fit statistic, Q, for each sample. Q is simply the sum of squares of each row (sample) of $\mathbf{E}$ (from equation (2)); for example, for the jth sample in $\mathbf{X}$, $\mathbf{x}_j$,

$$Q_j = \mathbf{e}_j\mathbf{e}_j^{\mathrm{T}} = \mathbf{x}_j(\mathbf{I} - \mathbf{P}_k\mathbf{P}_k^{\mathrm{T}})\mathbf{x}_j^{\mathrm{T}} \tag{5}$$

where $\mathbf{e}_j$ is the jth row of $\mathbf{E}$, $\mathbf{P}_k$ is the matrix of the first k loading vectors retained in the PCA model (where each vector is a column of $\mathbf{P}_k$) and $\mathbf{I}$ is the identity matrix of appropriate size ($n \times n$). The Q statistic indicates how well each sample conforms to the PCA model. It is a measure of the amount of variation in each sample *not* captured by the k principal components retained in the model.

A measure of the variation *within* the PCA model is given by Hotelling's T^2 statistic. T^2 is the sum of normalized squared scores defined as

$$T_j^2 = \mathbf{t}_j\boldsymbol{\lambda}^{-1}\mathbf{t}_j^{\mathrm{T}} = \mathbf{x}_j\mathbf{P}\boldsymbol{\lambda}^{-1}\mathbf{P}^{\mathrm{T}}\mathbf{x}_j^{\mathrm{T}} \tag{6}$$

where $\mathbf{t}_j$ refers to the jth row of $\mathbf{T}_k$, the matrix of k score vectors from the PCA model. The matrix $\boldsymbol{\lambda}^{-1}$ is a diagonal matrix containing the inverse eigenvalues associated with the k eigenvectors (principal components) retained in the model. Statistical limits can be developed for Q and T^2 (along with limits on the scores and individual residuals).[4,6]

5.2. Applying an existing PCA model: MSPC

Once a PCA model has been developed (including mean and variance scaling vectors, eigenvalues, loadings, statistical limits on the scores, Q and T^2), it can be used with new process data to detect changes in the system generating the data. Scores for new data, $\mathbf{t}_{i,\mathrm{new}}$, can be obtained for new data $\mathbf{X}_{\mathrm{new}}$ with equation (4) using the original loading vectors $\mathbf{p}_i$. In a similar fashion, new Q and T^2 can be obtained with equations (5) and (6) by substituting $\mathbf{x}_{i,\mathrm{new}}$ for $\mathbf{x}_i$. When one monitors these values as the process proceeds, the result is multivariate statistical process control (MSPC).[8–13]

In this work we will use primarily Q and T^2 for detecting system faults when using PCA and MPCA (see below). Some discussion of the geometric interpretation of Q and T^2 is perhaps in order. As noted above, Q is a measure of the variation in the data outside the plane of the PCA model. Refer again to the case of three variables where the data are restricted to lie on a plane as shown in Figure 3. Such a system would be well described by a two-PC model. Q is a measure of the distance off the plane formed by the first two PCs. In fact, $\sqrt{Q}$ is the Euclidean distance of the operating point from the plane formed by the two-PC model. A point with an unusually large Q value is depicted in Figure 3. The Q limit defines a distance off the plane that is considered unusual for normal operating conditions. T^2, on the other hand, is a measure of the distance from the multivariate mean to the projection of the operating point onto the plane defined by the two PCs. The T^2 limit defines an ellipse on the plane within which the operating point normally projects. Again, Figure 3 shows a point with a high T^2 value.

In practice, violations of the Q and T^2 limits generally occur for different reasons. Assuming a normal value of Q, a T^2 fault indicates that the process has gone outside the usual range of operation but in a direction of variation common to the process. In some sense there is too much or too little of something normally present in the process. A Q fault indicates that the process has gone in an entirely new direction— something entirely new has happened. It is our experience that most process faults show up in Q. Very few faults are detected by T^2 alone.

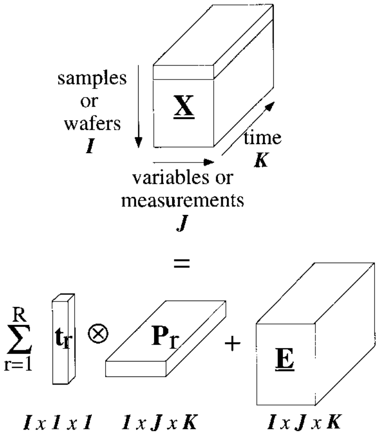


Figure 4. Three-dimensional data array and multiway PCA decomposition

5.3. Multiway PCA

The PCA method outlined above takes no explicit account of the ordered nature of a data set, i.e. the fact that the data were collected in a sequential manner. Reordering the samples in PCA would produce identical results. There are methods that explicitly consider that the data are ordered. These are referred to as multiway methods because the data are usually organized into time-ordered blocks that are each representative of a single sample or process run. The blocks are then arranged into multiway matrices. Multiway methods are particularly useful for the analysis of batch process data.

Consider the three-dimensional data array shown in Figure 4. A data matrix of this type would be typical of a series of runs of a batch process such as our example of semiconductor processing where each ‘batch’ is a wafer. Here there are $j = 1, 2, \ldots, J$ variables measured at times $k = 1, 2, \ldots, K$ throughout the batch (not to be confused with k principal components). Similar data will exist on $i = 1, 2, \ldots, I$ runs of the batch process. The data can be summarized in the three-dimensional ($I \times J \times K$) array $\underline{\mathrm{X}}$. Different batch runs (samples) are arranged along the vertical side, different process measurements (variables) along the horizontal side, and time recedes into the figure. Each horizontal slice through the array is a $J \times K$ matrix representing the time history for all variables of a particular batch or sample. Each vertical slice made parallel to the front face of the cube is an $I \times J$ matrix representing the values of all the variables in all the batches taken at a common time. A vertical slice made parallel to the side of the cube (the time axis) would represent an $I \times K$ matrix of all the time histories of a single variable for all the batches.

There are several methods for decomposing the array $\underline{\mathbf{X}}$.[14] These methods include trilinear decomposition (TLD)[15] and parallel factor analysis (PARAFAC).[16,17] Initially we will consider one of the more straightforward approaches, that of multiway PCA (MPCA).[18] Each of the decomposition methods places different constraints on the resulting matrices and vectors.

MPCA is statistically and algorithmically consistent with PCA and has the same goals and benefits.[19,20] In MPCA the array $\underline{\mathbf{X}}$ is decomposed as the summation of the product of score vectors (**t**) and loading matrices (**P**) plus a residual array $\underline{\mathbf{E}}$ that is minimized in a least squares sense:

$$\underline{\mathbf{X}} = \sum_{r=1}^{R} \mathbf{t}_r \otimes \mathbf{P}_r + \mathbf{E} \tag{7}$$

This decomposition is shown graphically in Figure 4. It is done in accordance with the principles of PCA and separates the data into two parts. The noise or residual part $\underline{\mathbf{E}}$ is as small as possible and is associated with non-deterministic variation in the data. The systematic part, the sum of the $\mathbf{t}_r \otimes \mathbf{P}_r$, expresses the deterministic variation as one fraction ($\mathbf{t}$) related only to batches and a second fraction ($\mathbf{P}$) related to variables and their time variation.

MPCA is equivalent to performing PCA on a large two-dimensional matrix formed by unfolding the three-way array $\underline{\mathbf{X}}$ in one of six possible ways, only three of which are mathematically unique. For example, one might unfold $\underline{\mathbf{X}}$ in such a way as to put each of its vertical slices ($I \times J$) side by side to the right, starting with the slice corresponding to the first time interval. The resulting two-dimensional matrix has dimensions $I \times JK$. This particular unfolding allows one to analyze variability among the batches in $\underline{\mathbf{X}}$ by summarizing information in the data with respect to variables and their time variation. A mathematically equivalent unfolding would be to take slices off the side of $\underline{\mathbf{X}}$ and place them down the time axis, which also forms a matrix with dimensions $I \times JK$. (The latter unfolding orders the matrix with the history of each variable kept together, while the former orders the matrix with all the measurements taken at the same time kept together.) One might also be interested in unfolding $\underline{\mathbf{X}}$ in other ways; however, the unfolding discussed above (or its mathematical equivalent) is the only way that keeps batch (sample)-specific information separate from time and variable information.

The MPCA algorithm proceeds as follows. First the matrix is unfolded in one of the two equivalent ways described above. Each column of the resulting matrix is then mean centered and, if appropriate, scaled to unit variance (autoscaled). An eigenvector decomposition as described in equations (1–3) is then applied to the unfolded $\underline{\mathbf{X}}$. Each of the $\mathbf{p}$, however, is really an unfolded version of the loading matrix $\mathbf{P}_r$. After the $\mathbf{p}$ are obtained, the $\mathbf{P}_r$ can be obtained by reversing the unfolding procedure. In a similar manner the three-way array $\underline{\mathbf{E}}$ can be formed by folding the PCA residual matrix $\underline{\mathbf{E}}$. The Q and T^2 statistics can be calculated using the unfolded solution as shown in equations (5) and (6).

This version of MPCA explains variation in measured variables about their average trajectories. Subtracting the average trajectory from each variable (accomplished by mean centering the columns of the unfolded matrix $\mathbf{X}$) removes large amounts of normal variation from the process data. The ith elements of the $\mathbf{t}$ score vectors correspond to the ith batch (sample) and summarize the overall variation in this batch with respect to the other batches in the database over the entire history of the batch. The $\mathbf{P}$ loading matrices summarize the time variation in the measured variables about their average trajectories. The elements of $\mathbf{P}$ are the weights, which when applied to each variable at each time interval within a batch, give the $\mathbf{t}$ scores for that batch. Additional examples of MPCA for MSPC can be found in References 21–23.

5.4. Trilinear decomposition and PARAFAC

MPCA is in some sense a ‘poor man’s’ multiway technique as it relies upon a two-way method (PCA) and rearrangement of the original data. An additional problem is that the loading matrices are very difficult to interpret as they convolute time and variable information. TLD and PARAFAC are true multiway methods in that they decompose the original array into factors in each of the original dimensions. The TLD and PARAFAC model is

$$d_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr} + e_{ijk} \tag{8}$$

where R is chosen such that $\underline{\mathbf{E}}$ with elements e_{ijk} has small norm. This is the trilinear model. If $\mathbf{A}$ is defined such that its rth column is $\mathbf{a}_r$, and likewise for $\mathbf{B}$ and $\mathbf{C}$, then the outer product of $\mathbf{a}_r$, $\mathbf{b}_r$ and $\mathbf{c}_r$ is the rth triad of the PARAFAC model. This model is shown graphically in Figure 5. In words, the

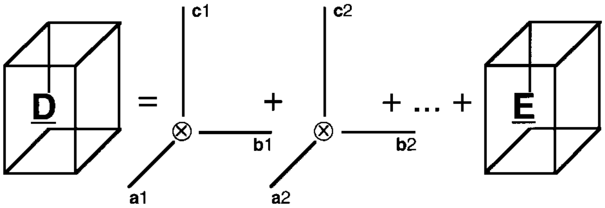


Figure 5. TLD and PARAFAC model

PARAFAC and TLD model is a summation over the outer product of vector triads. Note that, in contrast with PCA, in multiway models the factors are referred to as loadings in all dimensions.

One very important property of the PARAFAC decomposition is that, under very mild assumptions, it is unique.[24,25] The result is that **A**, **B** and **C** give pure component responses and relative concentration estimates directly. This is a very powerful advantage of the method when interpretation of the models is important.

The major difference between TLD and PARAFAC is in how the models are identified. In PARAFAC the model is identified using an alternating least squares (ALS) procedure or by direct optimization of the parameters, or a combination of the two approaches. In the ALS approach the factors in a particular dimension are estimated as a least squares fit of the factors in the remaining dimensions to the data. This process is repeated on each dimension sequentially until the solution converges. The result is a set of factors which fit the data in a least squares sense, i.e. minimize the sum of squared residuals. This can also be accomplished by direct optimization of the factors in all dimensions. Note that, unlike PCA, the number of factors must be decided before applying the algorithm, and all factors are determined simultaneously.

The TLD model is identified by application of the generalized rank annihilation method (GRAM).[26,27] GRAM uses the QZ algorithm[28] to solve for the joint invariant subspace of a pair of matrices. In TLD this pair of matrices is made up of linear combinations of the original sample matrices chosen in such a way as to contain variation from all the factors in the data. The factors obtained from the GRAM step are then fit to the original data in a single least squares step. The major advantage of TLD is that it is computationally much faster than PARAFAC. When used with relatively noisy data, however, depending upon the algorithm used, application of GRAM can lead to imaginary solutions which have no physical interpretation.[29] These factors can be rotated back to the nearest real solution,[30,31] but in our experience the results are often not satisfactory.

Preprocessing is an important aspect of applying multiway models.[25] In general, one would like to apply centering and scaling of the data in such a way that the trilinearity of the model would be maintained. However, when there is no theory of the system being investigated that suggests how this be done (such as in spectral data where the trilinear model is expected to hold on the raw data), preprocessing becomes more of a trial-and-error procedure. In this work, several different procedures were tested as discussed below.

5.5. Applying TLD and PARAFAC to new data

Once a TLD or PARAFAC model has been identified, it is a simple matter to fit the model to new data. Usually, the loading sample order is fit to the other two orders using a single least squares step. Fit residuals, similar to Q in PCA and MPCA models, can also be calculated. The question then becomes whether the new loadings and residuals are significantly different from those of the original data on which the model was developed. Unfortunately, statistical tests for the loadings and residuals in these models are not well established, in part because it is difficult to determine the proper number of degrees of freedom to use in the calculations. In this work, limits were developed for loadings by assuming that the loadings were normally distributed. Limits were then calculated based on the standard normal variate. Limits on the residuals were determined directly from the fitted data, e.g. a 90% confidence limit was estimated as residual for which 90% of the calibration samples were lower. Clearly there is room for improvement here. However, in use this approximation makes little difference in the following study as very few samples were borderline.

5.6. Data preprocessing

Before applying PCA or MPCA, several options are available for preprocessing the data. In PCA one would often simply determine a single mean and variance for scaling the data and apply this scaling to all additional data. In our current example, however, it is known that process drift occurs and that the process mean may shift. Thus one might consider mean centering the data from each wafer to eliminate the effect of drift. It is also possible to continually rebuild PCA models so that they are based only on recent data.[2,32]

An additional complication involves stretching of the time axis in the data record. In the etch process, time line stretching causes blocks of data from each wafer to have different numbers of samples. This is due to differing lengths of the etch because of changes in layer thickness. One way to approach this is to simply average the data from each wafer over all available samples and work with only a mean. Another approach would be to select a specified number of samples where some point in the selected record corresponds to some particular process event. This approach was used for the MPCA, TLD and PARAFAC models. The process event was the ‘peak’ of the titanium etch (point at which the Ti concentration was highest in the plasma, which is evident from the endpoint and RFM data). In related work we have also used speech recognition methods such as dynamic time warping to map the process response back onto a reference trace.[3,33]

As will be seen in the following sections, the data pretreatment method can have a significant impact on the overall sensitivity and robustness of the method.

6. INDUCED FAULT EXPERIMENTS

A series of three experiments (29, 31 and 33) were performed where faults were intentionally induced by changing the TCP power, RF power, pressure, Cl_2 or BCl_3 flow rate and He chuck pressure. These three experiments consisted of a total of 129 wafers with 21 faults.

To make the test more representative of an actual sensor failure, the analysis was done with ‘reset’ values: values for the controlled variable which was intentionally moved off its setpoint were reset to have the same mean as its normal baseline value, i.e. the controlled variable which was changed was reset to look normal in the data file. For example, if the induced fault was a change in the TCP power from 350 to 400 W, the data file value of the TCP power was reset from a mean of 400 back to 350 by addition of a constant bias. The resulting data look as if the controller was seeing a biased sensor for TCP power and adjusting accordingly: TCP power would appear normal, but it would not be. The effect of a TCP power offset, however, should be evident (we hope) in the remaining process variables, because the apparent relationship between the TCP power and the remaining variables should be different.

The three induced fault experiments were run at widely spaced intervals (in February, March and April 1996 respectively). Process drift is apparent in the data: each experiment has a significantly different multivariate mean. This is evident in Figure 6, which shows the scores on the first two PCs of the machine state data for all three experiments. The data clearly split into three groups, one for each of the experiments. This suggests that models based on all the data will define a much larger region of

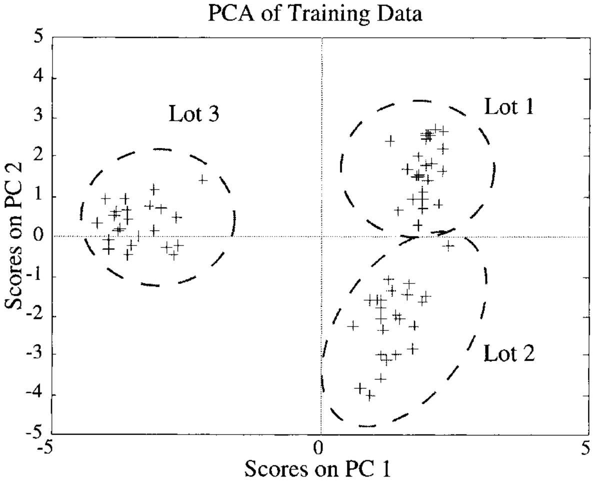


Figure 6. Scores on first two PCs from analysis of experiment 29, 31 and 33 induced fault data

the multivariate space as normal variation than would a model of a single experiment. We will refer to a model of all the data as a global model, and a model of each of the single experiments as a local model.

7. RESULTS

Data from experiments 29, 31 and 33 were used to test the sensitivity of PCA and MPCA for detecting the induced faults. Machine state, RFM and OES data were available for each of these experiments. As described above, these experiments included some wafers where the setpoints for some variables were offset from the normal recipe. Prior to analysis, data from the sensors that measure each of these parameters (and are used for feedback control) were ‘reset’ to their means from previous runs. All subsequent analysis was performed using the PLS_Toolbox 2·0 software[34] running under MATLAB 5.[35]

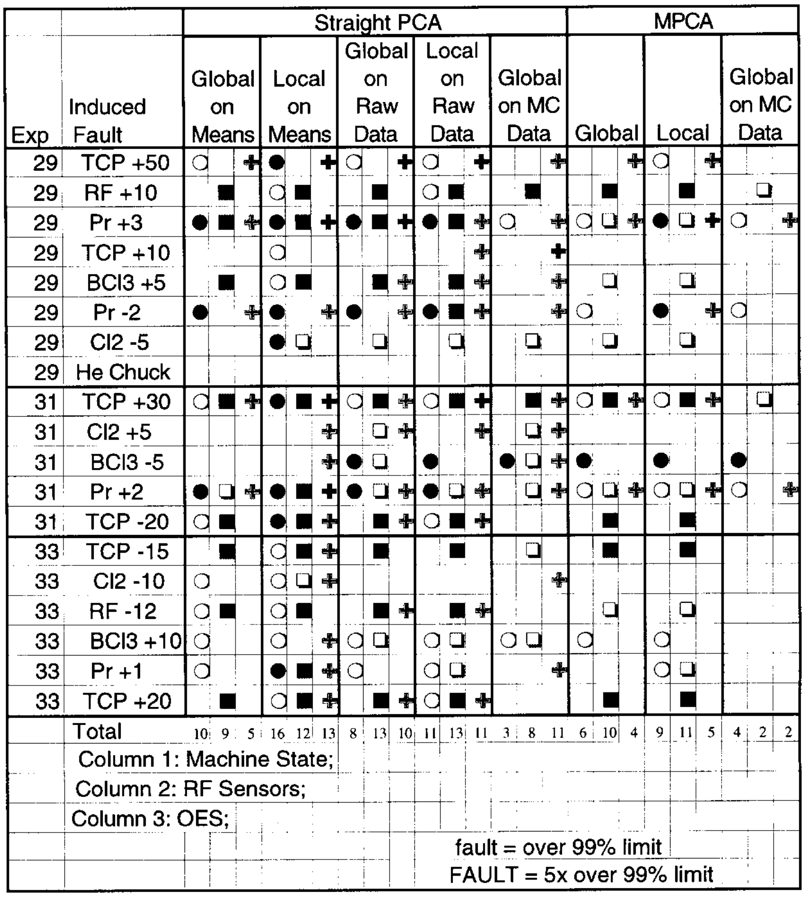
Several different approaches were employed in the development of the fault detection models used in this test. To get a direct comparison of the sensitivity of the process sensors, only data from these experiments were used in model development (very few additional data exist where all three sensor systems are available). Models were developed that were intended to mimic the local and global behavior of the process. Local models were built using only data from a particular experiment, i.e. a model was built using data from the normal wafers from an experiment and was used to test the remaining wafers. The local models were intended to represent the upper limit of what might be achievable with models that update themselves continuously and thus are always local. Global models were developed using the normal wafers from all the experiments simultaneously and then tested on the fault wafers. This represents the case where models span a large amount of long-term process variation or drift, i.e. include lot-to-lot and over-a-maintenance-cycle effects. These models included a larger amount of variation as normal than the local models.

The data were also preprocessed in a number of different ways prior to analysis. For some tests, data from each wafer were reduced to a single vector of means of the variables over the entire wafer. In other cases, raw data were used for model development. Analysis was also performed using raw data where data from each of the wafers were centered to their own mean. When this method is used, only faults which change the covariance matrix of the wafers can be detected.

Multiway analysis was also performed. In these instances each sample in the analysis includes the time history of the process sensors. As described above, the same number of samples was used for each wafer during model development and testing. For machine state data, 70 samples were used, including the last 25 data points from step 4 and the first 45 data points for step 5. Similarly, 25 and 28 data samples were used from the RFM and OES respectively. RFM and OES variables that mirrored the process endpoint trace were found and a consistent number of samples were selected on either side of the peak of the TiN etch.

Data for the MPCA models were 'group scaled'. Here the array is unfolded to a two-dimensional matrix that is batches by (time steps * variables). Group scaling is applied so that, after subtracting the mean trajectory, each variable 'block' is adjusted to have the same variance. For TLD and PARAFAC the data were 'autoscaled'. The array is unfolded to a two-dimensional matrix that is (batches * time steps) by original variables. The result was autoscaled, then refolded. The MPCA-style scaling was also applied to the machine data prior to TLD and PARAFAC analysis. The results were similar to the

Table 2. Results of sensitivity tests for single sensor systems



Exp	Induced Fault	Straight PCA															MPCA								
		Global on Means			Local on Means			Global on Raw Data			Local on Raw Data			Global on MC Data			Global			Local			Global on MC Data		
29	TCP +50	○		✚	●		✚	○		✚	○		✚			✛			✛	○		✛			
29	RF +10		■		○	■			■		○	■			■			■			■			❑	
29	Pr +3	●	■	✛	●	■	✚	●	■	✚	●	■	✛	○		✛	○	❑	✛	●	❑	✚	○		✛
29	TCP +10				○								✛			✚									
29	BCl3 +5		■		○	■			■	✛		■	✛			✛		❑			❑				
29	Pr -2	●		✛	●		✛	●		✛	●	■	✛			✛	○			●		✛	○		
29	Cl2 -5				●	❑			❑			❑			❑			❑			❑				
29	He Chuck																								
31	TCP +30	○	■	✛	●	■	✚	○	■	✛	○	■	✚		■	✛	○	■	✛	○	■	✛		❑	
31	Cl2 +5						✛		❑	✛			✛		❑	✛									
31	BCl3 -5						✛	●	❑		●			●	❑	✛	●			●			●		
31	Pr +2	●	❑	✛	●	■	✚	●	❑	✛	●	❑	✛		❑	✛	○	❑	✛	○	❑	✛	○		✛
31	TCP -20	○	■		●	■	✛		■	✛	○	■	✛					■			■				
33	TCP -15		■		○	■	✛		■			■			❑			■			■				
33	Cl2 -10	○			○	❑	✛									✛									
33	RF -12	○	■		○	■			■	✛		■	✛					❑			❑				
33	BCl3 +10	○			○		✛	○	❑		○	❑		○	❑		○			○					
33	Pr +1	○			●	■	✛	○			○	❑				✛				○	❑				
33	TCP +20		■		○	■	✛		■	✛	○	■	✛					■			■				
	Total	10	9	5	16	12	13	8	13	10	11	13	11	3	8	11	6	10	4	9	11	5	4	2	2
	Column 1: Machine State;																								
	Column 2: RF Sensors;																								
	Column 3: OES;																								
											fault = over 99% limit														
											FAULT = 5x over 99% limit														

'autoscaling' procedure above, but on average not quite as good. No scaling was applied to OES data as they have a natural zero and should fit the trilinear model best with no scaling.

Sensitivity results for the machine state, RFM and OES sensors used individually are shown in Table 2. The results for the sensors in combination are shown in Table 3. Results for the TLD and PARAFAC models on individual sensor systems are shown in Table 4. The faults are listed in the second column of each table. Note that only faults where all data were available are considered in the table; thus there are 19 faults listed rather than the original 21. In Tables 2 and 3 the results for straight PCA models are shown on the left for five different data pretreatment approaches. MPCA model results are shown on the right for three different data pretreatment approaches. Six different combinations of sensors are considered for each method/preprocessing combination: machine state, RFM, OES (Table 2), machine state + RFM + OES, machine state + RFM and machine state + OES (Table 3). A symbol in the body of the table indicates that the particular combination of data analysis method, pretreatment and sensors caught the particular fault. An open symbol indicates that the fault exceeded the 99% confidence limit, while a full symbol indicates that the fault exceeded the 99%

Table 3. Results of sensitivity tests for combinations of sensor systems

		Straight PCA									MPCA								
Exp	Induced Fault	Global on Means			Local on Means			Global on Raw Data	Local on Raw Data	Global on MC Data	Global			Local			Global on MC Data		
29	TCP +50	○	□	✙	●	■	✚				○		✙	○		✙			
29	RF +10	●	■		●	■					●	■		●	■		○	■	
29	Pr +3	○	■	✙	●	■	✚				○	□	✙	○	□	✙	○	□	✙
29	TCP +10				○	□	✙												
29	BCl3 +5	○	□		○	■						□		○	□				
29	Pr -2	○	□	✙	●	■	✚				○	□	✙	○	□	✙			
29	Cl2 -5		□		○	■	✙								□				
29	He Chuck																		
31	TCP +30	●	■	✙	●	■	✚				○	□	✙	○	■	✙	○	□	
31	Cl2 +5						✙												
31	BCl3 -5				○		✙				●	■	✚	●	■	✚	●	■	✚
31	Pr +2	○	□	✙	●	■	✚				○	□	✙	○	□	✙	○	□	✙
31	TCP -20	●	■		●	■	✙				○	□		○	□				
33	TCP -15	●	■		●	■	✙				○	□		○	□				
33	Cl2 -10		□		○	□	✚												
33	RF -12	○	□		○	■	✙								□				
33	BCl3 +10		□		○	□	✙					□			□				
33	Pr +1		□		○	■	✙								□				
33	TCP +20	●	■		●	■	✙				○	□		○	□				
	Total	11	15	5	17	16	16				10	11	6	11	14	6	5	5	3
	Column 1: Machine State + RF +OES;																		
	Column 2: Machine State + RFM;																		
	Column 3: Machine State + OES;																		
											fault = over 99% limit								
											FAULT = 5x over 99% limit								

Table 4. Results of sensitivity tests for single sensor systems with TLD and PARAFAC

		TLD						PARAFAC					
Exp	Induced Fault	Global			Local			Global			Local		
29	TCP +50	●		✚	●		✚	●		✚	●		✚
29	RF +10		■		●	■			■		●	■	
29	Pr +3	●		✚	●	■	✚	●		✚	●	■	✚
29	TCP +10			✚	●		✚				●		✚
29	BCl3 +5					■					●	■	
29	Pr -2	●		✚	●		✚	●		✚	●		✚
29	Cl2 -5	●			●	■		●			●	■	
29	He Chuck												
31	TCP +30	●	■	✚	●	■	✚	●	■	✚	●	■	✚
31	Cl2 +5	●		✚				●			●	■	
31	BCl3 -5			✚						✚		■	
31	Pr +2	●		✚	●	■	✚	●			●	■	✚
31	TCP -20	●	■		●	■	✚	●	■		●	■	✚
33	TCP -15		■		●	■	✚		■		●	■	✚
33	Cl2 -10	●			●	■	✚	●			●	■	✚
33	RF -12		■			■			■		●	■	
33	BCl3 +10	●	■		●		✚	●			●		✚
33	Pr +1	●			●	■		●			●	■	
33	TCP +20		■	✚	●	■	✚	●	■	✚	●	■	✚
	Total	11	7	9	14	12	12	12	6	6	17	14	11
	Column 1: Machine State							●					
	Column 2: RFM;							■					
	Column 3: OES;							✚					

limit by a factor of five or more. Note, however, that for analysis of the raw data an open symbol indicates that more than 15% of the time samples exceeded the 95% confidence limit, while a full symbol indicates that over 30% of the samples exceeded the 95% confidence limit. Also, sensors were not considered in combination using the raw data, since the data acquisition times are not synchronized between the sensors. Combinations of sensors were not considered for TLD and PARAFAC models for the same reason. Note also that no distinction is made between 99% and 5 × 99% in the TLD and PARAFAC results.

The number of faults caught with each method and single sensor system is shown in Table 5. The upper portion of the table gives the results for global models (models based on all three experiments), while the lower portion of the table gives the results for local models (separate models for each of the three experiments).

Table 5. Faults detected for each combination of sensor, method and timescale

		TLD	PARAFAC	MPCA	PCA/mean
Global	Machine	11	12	10	10
	RFM	7	6	11	9
	OES	9	6	6	5
Local	Machine	14	17	11	16
	RFM	12	14	14	12
	OES	12	11	6	13
Mean		10·8	11·0	9·7	10·8

8. DISCUSSION

Several trends are evident upon examination of the results. It is clear that local models outperform global models. In Tables 2–4 all local models performed better than all similarly configured global models. This is particularly apparent in Table 5. This is expected because global models include a larger amount of variation as normal. Thus faults are smaller relative to the normal process variation included in global models and are therefore more difficult to detect. PCA on the wafer means is somewhat more sensitive than PCA on raw data for machine state data but not for RFM and OES. The increased sensitivity with machine data, which tend to have a larger proportion of unmodeled variance, is probably signal averaging, i.e. it is easier to see a shift in the mean when signals are averaged over many samples. With OES and RFM data there is generally very little unmodeled variance in the raw data, and changes are more easily detected in the raw data.

In this analysis, MPCA does not perform better than PCA of the raw data. However, in previous analyses, with different arrangements of the data, MPCA did perform better on the machine state and OES sensors but not the RFM. It is expected that MPCA will be more sensitive to some types of faults than PCA because the time-ordered nature of the data is considered explicitly. Faults which change the shape of the process trajectory, but not the overall mean and covariance, would be detectable with MPCA but not with PCA. Changes in shape can include stretching due to lengthening or shortening of some periods of the etch. In previous analysis, data were arranged for MPCA by including a specified number of samples starting from the beginning of the run, as opposed to including data centered on a particular feature near the middle of the run. MPCA models are more sensitive to stretching when the starting points are fixed in the data record, rather than a point near the center. It is not clear why MPCA does not lead to increased sensitivity when used with the RFM data. It may be that the shapes of the RFM trajectories are inherently more variable, making changes to them harder to detect.

PARAFAC performed slightly better than TLD, which was equivalent to PCA on the means, as shown in Table 5. The differences between TLD and PARAFAC may be due to problems resulting from imaginary solutions, which, when rotated back to real solutions, still tend not to be as good as the PARAFAC solutions.

Methods based on wafers centered to their own mean are less sensitive than those based on raw data, as might be expected. However, there are several instances where the analysis on mean-centered wafers detects faults whereas analysis of the means does not. This suggests that these techniques could be used simultaneously.

The overall performance of the different sensors is similar; however, the OES sensor appears to degrade the most as the models are changed from local to global. This is no doubt due to the vary large amount of drift in the OES signals over the course of a clean cycle due to residue build-up on the

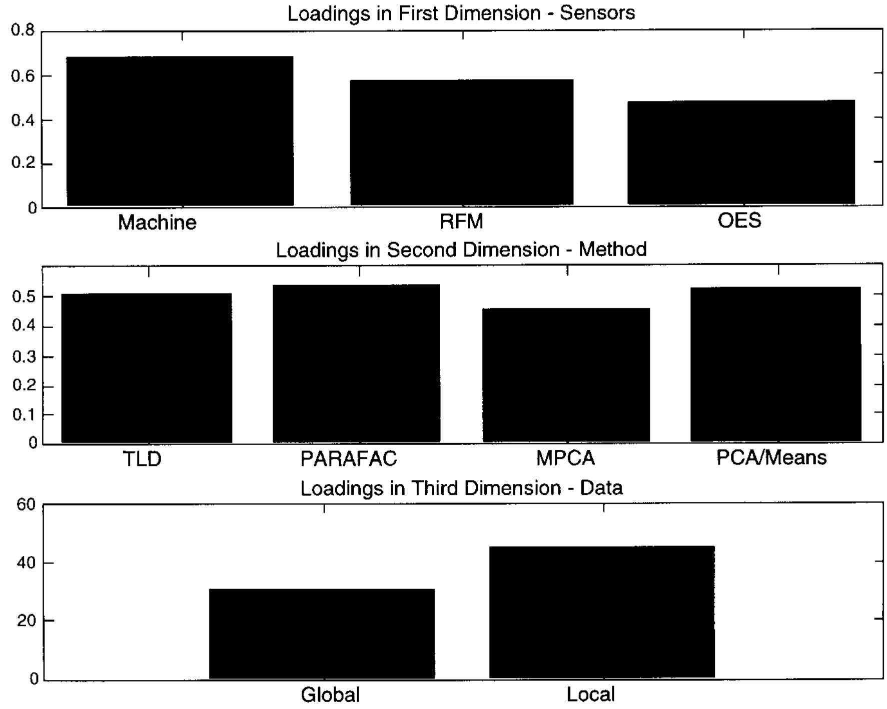


Figure 7. PARAFAC loadings of sensitivity results

chamber window. The sensitivity of RFM models, on the other hand, changed little when they were local or global. This suggests that the RFM sensors are the most stable and/or least sensitive to lot-to-lot changes that do not affect processing.

The best combination of sensors is machine state plus RFM. This combination is more successful for detecting faults than any of the sensors alone or in combination when global models are considered. When OES data are combined with machine state and RFM data, the ability to detect faults generally decreases in the global models. This is not necessarily true for local models where long-term variation in the OES is not important.

The data from Table 5 (single sensor group results) were arranged in a three-way matrix and modeled by PARAFAC. A one-factor model fit the data reasonably well, with a root-mean-square fit error of 1·75. The PARAFAC loadings are shown in Figure 7. Here the trends are clear: machine data tend to catch more faults than RFM, which is better than OES. PARAFAC models are slightly more sensitive to faults than TLD and PCA on the means, which are better than MPCA. Local models catch more faults than global models. This example illustrates one of the outstanding features of PARAFAC, which is the ease of interpretation of the model. Note, however, that two-way analysis of variance shows that the difference in methods is not significant. The difference in sensors is significant at 99%, as is the difference between global and local models.

9. CONCLUSIONS

This study has shown how one can systematically step through the options for sensor systems and data treatment for fault detection systems in order to select the best measurements and analysis method for the particular job. For this particular application, PARAFAC appeared to work best, followed closely by PCA on the means and TLD. Given the simplicity of PCA of the means, this might be a reasonable choice in practice. The major unresolved issue in this paper concerns how one might deal with process and sensor drift. It is apparent that this had the single largest effect on the ability to detect sensor faults in this study. This issue is the subject of our companion study.[2]

REFERENCES

1. B. M. Wise, N. B. Gallagher, S. W. Butler, D. White and G. G. Barna, 'Development and benchmarking of multivariate statistical process control tools for a semiconductor etch process: impact of measurement selection and data treatment on sensitivity', *IFAC SAFEPROCESS'97*, Kingston upon Hull, 1997, pp. 35–42.
2. N. B. Gallagher, B. M. Wise, S. W. Butler, D. D. White Jr and G. G. Barna, 'Development and benchmarking of multivariate statistical process control tools for a semiconductor etch process: improving robustness through model updating', *presented at ADCHEM 1997*, Banff, 1997.
3. G. G. Barna, 'Procedures for implementing sensor-based fault detection and classification (FDC) for advanced process control (APC)', *SEMATECH Technical Transfer Document 97013235A-XFR* (1997).
4. J. E. Jackson, *A User's Guide to Principal Components*, Wiley, New York (1991).
5. B. M. Wise and B. R. Kowalski, 'Process chemometrics', in *Process Analytical Chemistry*, pp. 259–312, Blackie, London (1995).
6. B. M. Wise and N. B. Gallagher, 'The process chemometrics approach to chemical process fault detection and supervision', *J. Process Control*, **6**, 329–348 (1996).
7. S. Wold, K. Esbensen and P. Geladi, 'Principal components analysis', *Chemometrics Intell. Lab. Syst.* **2**, 37–52 (1987).
8. B. M. Wise and A. H. McMakin, 'A statistical technique for analyzing data from liquid-fed ceramic melters', *PNL-SA-15267*, Pacific Northwest Laboratory, Richland, WA (1987).
9. B. M. Wise, D. J. Veltkamn, B. Davis, N. L. Ricker and B. R. Kowalski, 'Principal components analysis for monitoring the West Valley liquid fed ceramic melter', *presented at Waste, Management '88*, Tucson, AZ, 1988.
10. B. M. Wise, and N. L. Ricker, 'Feedback strategies in multiple sensor systems', *AIChE Symp. Ser.* **85**, 19–23 (1989).
11. B. M. Wise, N. L. Ricker, D. J. Veltkamp and B. R. Kowalski, 'A theoretical basis for the use of principal components models for monitoring multivariate processes', *Process Control Qual.* **1**, 41–51 (1990).
12. B. M. Wise, D. J. Veltkamp, N. L. Ricker, B. R. Kowalski, S. M. Barnes and V. Arakali, 'Application of multivariate statistical process control (MSPC) to the West Valley slurry-fed ceramic melter process', *presented at Waste Management '91*, Tucson, AZ, 1991.
13. J. V. Kresta, J. F. MacGregor and T. E. Marlin, 'Multivariate statistical monitoring of process operating performance', *Can. J. Chem. Engng.* **69**, 35–47 (1991).
14. P. Geladi, 'Analysis of multi-way (multi-mode) data', *Chemometrics Intell. Lab. Syst.* **7**, 11–30 (1989).
15. E. Sanchez and B. R. Kowalski, 'Tensorial resolution: a direct trilinear decomposition', *J. Chemometrics,* **4**, 29–45 (1990).
16. A. K. Smilde and D. A. Doornbos, 'Three way methods for the calibration of chromatographic systems: comparing PARAFAC and three-way PLS', *J. Chemometrics.* **5**, 345–360 (1991).
17. A. K. Smilde, Y. Wang and B. R. Kowalski, 'Theory of medium-rank second-order calibration with restricted-Tucker models', *J. Chemometrics.* **8**, 21–36 (1994).
18. S. Wold, P. Geladi, K. Esbensen and J. Ohman, 'Multi-way principal components and PLS analysis', *J. Chemometrics*, **1**, 41–56 (1987).
19. P. Nomikos and J. F. MacGregor, 'Monitoring batch processes using multiway principal component analysis', *AIChE J.* **40**, 1361–1375 (1994).
20. P. Nomikos and J. F. MacGregor, 'Multivariate SPC charts for monitoring batch processes', *Technometrics.* **37**, 97–108 (1995).
21. K. A. Kosanovich, M. J. Piovoso, K. S. Dahl, J. F. MacGregor and P. Nomikos, 'Multi-way PCA applied to an industrial batch process', *presented at Am. Control Conf.* 1994.
22. N. B. Gallagher, B. M. Wise and C. W. Stewart, 'Application of multi-way principal components analysis to nuclear waste storage tank monitoring', *Comput. Chem. Engng.* **20**, S739–S744 (1996).

23. B. M. Wise and N. B. Gallagher, ‘Multi-way analysis in process monitoring and modeling’, *AIChE Symp. Ser.* **316**, 271–274 (1997).
24. R. Bro, ‘PARAFAC. Tutorial and applications’, *Chemometrics Intell. Lab. Syst.* **38**, 149–171 (1997).
25. R. Bro, ‘Multi-way analysis in the food industry—models, algorithms and applications’, *Doctoral Thesis*, University of Amsterdam (1998).
26. E. Sanchez and B. R. Kowalski, ‘Tensorial calibration: II. Second-order calibration’, *J. Chemometrics.* **2**, 265–280 (1988).
27. B. E. Wilson, E. Sanchez and B. R. Kowalski, ‘An improved algorithm for the generalized rank annihilation method’, *J. Chemometrics.* **3**, 493–498 (1989).
28. C. B. Moler and G. W. Stewart, *SIAM J. Numer. Anal.* **10**, 241 (1973).
29. K. Faber, ‘On solving generalized eigenvalue problems using MATLAB’, *J. Chemometrics*, **11**, 87–91 (1997).
30. S. Li, J. C. Hamilton and P. J. Gemperline, ‘Generalized rank annihilation method using similarity transformations’, *Anal. Chem.* **64**, 599–607 (1992).
31. S. Li and P. J. Gemperline, ‘Eliminating complex eigenvectors and eigenvalues in multiway analyses using the direct trilinear decomposition method’, *J. Chemometrics.* **7**, 77–78 (1993).
32. S. Wold, ‘Exponentially weighted principal components analysis’, *Chemometrics Intell. Lab. Syst.* **23**, 149–161 (1994).
33. D. White, G. G. Barna, S. W. Butler, B. M. Wise and N. B. Gallagher, ‘Methodology for robust and sensitive fault detection’, *Electrochem. Soc. Proc.* **97–9**, 55–79, (1997).
34. B. M. Wise and N. B. Gallagher, PLS _Toolbox for Use with MATLAB, Version 2·0, Eigenvector Research, Manson, WA (1998).
35. *MATLAB 5 User’s Guide*, The MathWorks, Natick, MA (1998).