Chemometric Methods for Process Monitoring and High-Performance Controller Design

Michael H. Kaspar and W. Harmon Ray
Dept. of Chemical Engineering, University of Wisconsin, Madison, WI 53706

A huge amount of data is collected by computer monitoring systems in the chemical process industry. Such tools as principal component analysis and partial least squares have been shown to be very effective in compressing this large volume of noisy correlated data into a subspace of much lower dimension than the original data set. Because most of what is eliminated is the collinearity of the original variables and the noise, the bulk of the information contained in the original data set is retained. The resulting low dimensional representation of the data set has been shown to be of great utility for process analysis and monitoring, as well as in selecting variables for control. These types of models can also be used directly in control system design. One way of approaching this is to use the loading matrices as compensators on the plant. Some advantages of using this approach as part of the overall control system design include automatic decoupling and efficient loop pairing, as well as natural handling of nonsquare systems and poorly conditioned systems.

Introduction

With the introduction of computer data acquisition and control to the process industries, it became possible to obtain huge volumes of data. Typically, hundreds of variables may be monitored in a single operating unit, with values being recorded thousands of times per day. Such data sets are overwhelming. However, rarely are all the variables that may be measured on a process independent. Often, many of the variables are highly correlated, and the process is, in reality, restricted to move in a space of much lower dimension than the number of possible measurements would indicate. In this case, much of the overwhelming volume of information is redundant. Analysis can be simplified greatly if the useful information can be extracted and the redundant information can be eliminated.

A number of such data reduction methods are available. Two of these are principal component analysis (PCA) and partial least squares (PLS). Principal component analysis is used to analyze a single block of data which contains significant redundancies by compressing it into a lower dimensional space which contains most of the variance of the original matrix. Partial least squares is used to model the relationship between two blocks of data while simultaneously compressing them.

Even when the number of measured variables is relatively small (as in the case of an identification experiment in which an important subset of the variables has been selected in advance), a large portion of the information in the data set may be redundant. Such redundancy in the input variables is often the result of poor experimental design. In some cases, good design may be impossible because of economic, operability, or safety constraints. Redundancy in the output variables may result directly from redundancy in the input variables, or it may result from the nature of the system itself. Poorly conditioned systems exhibit correlation among output variables in spite of good experimental design. Thus, the analysis of multivariable systems can benefit from application of these techniques—even systems which are not extremely large.

A number of applications of PCA and PLS have been reported in the literature. PCA and PLS have been used for a wide range of purposes including dimensionality reduction, measurement classification, outlier detection and process monitoring, selection of variables for control, selection of inference variables, and others. MacGregor gives an overview of the ways these methods have been used specifically in process analysis and control (MacGregor et al., 1991). These include preliminary analysis of data (taking advantage of the dimensionality reduction provided by these methods), multivariate statistical process control, development of predictive models for inferential control, and identification of dynamic models. The literature contains a number of examples of the use of

Correspondence concerning this article should be addressed to W. H. Ray.

PCA and PLS in dimensionality reduction, including one in which PCA was used to select variables to control in a polybutadiene reactor system (Roffel et al., 1989). Process monitoring using PCA and PLS has been demonstrated by Kresta who used these techniques to monitor simulations of a fluidized-bed reactor and an extractive distillation column, by Wise and Ricker who used these methods to monitor a glass melting process, and by Wise et al. who used PCA and PLS to detect process upsets, especially those related to sensor failure (Kresta et al., 1989; Wise and Ricker, 1989; Wise et al., 1989). In the area of dynamic modeling and control, canonical correlation analysis (somewhat similar to PLS) was used by Jutan to find the appropriate dimension and redefine the variable space for the stochastic part of the model of a butane hydrogenolysis reactor and by Wright to select predictor variables for inferential control of a tubular packed-bed reactor (Jutan et al., 1977; Wright et al., 1977). Partial least squares was used by Ricker in the development of a finite impulse response (FIR) model for an anaerobic wastewater treatment process (Ricker, 1988).

In the current work, PLS is used to model a process dynamically, and the resulting model is used to design a control system for the process. In doing this, the model is not transformed back to the original variable space as is often the case in reduced dimension modeling, but rather the variable transformations, as a fundamental part of the PLS modeling process, are retained and utilized for the advantages they can confer in controlling the process. Established techniques for process analysis and monitoring are also illustrated, and a grayscale representation of the monitoring diagrams showing the distributional nature of the process is introduced.

Definitions

This section contains an overview of the theoretical background of PCA and PLS. A thorough background treatment of PCA and related topics is presented in a number of multivariate analysis textbooks (Cooley and Lohnes, 1971; Mardia et al., 1979; Press, 1972). The development of PLS was more recent. This method is treated in detail by Wold and colleagues (Wold et al., 1984a, 1982; Wold et al., 1984b). An excellent tutorial on PLS is also given by Geladi and Kowalski (1986).

PCA

In principal component analysis (PCA), the goal is to model a single block of data using orthogonal components. The components are arranged in the order of decreasing importance, which is determined by how much of the variance of the data set is explained by each.

The mathematical and statistical basis for PCA is well developed. Let Γ be the covariance matrix of a p dimensional random process, x. The spectral decomposition of Γ is given by:

$$\Gamma = PLP' \tag{1}$$

where L is a diagonal matrix containing the ordered eigenvalues of Γ, and P is unitary matrix whose columns are the normalized eigenvectors of Γ. The principal component transformation is given by:

$$t = P'x \tag{2}$$

where t is the vector of transformed variables. The first element of t, t_1, is the first principal component of x. Remaining principal components two through p are similarly defined. If Γ is rank-deficient, there will be only $r<p$ principal components and r nonzero eigenvalues, where r is the rank of the covariance matrix. The covariance matrix of t is L. Thus, the elements of t are uncorrelated, and the variance of t_i is l_i.

In practice, the covariance matrix for the process is not known, but must be estimated from an $n \times p$ data matrix, X.

$$\hat{\Gamma} = n^{-1}X'X. \tag{3}$$

This estimate may be decomposed using the spectral decomposition. The principal component transformation for the matrix X is then given by:

$$T = XP. \tag{4}$$

Sometimes thinking in terms of a decomposition, rather than a transformation, is convenient. Thus, X can be decomposed as:

$$X = TP' \tag{5}$$

The columns of T are called the score vectors, and the columns of P are called the loading vectors.

A number of properties of the principal components of x are of importance.

1. $E(t_i) = 0$
2. $V(t_i) = l_i$
3. $C(t_i, t_j) = 0,\ i \neq j$
4. $V(t_1) > V(t_2) > \cdots > V(t_p)$
5. $\Sigma_{i=1}^{p} V(t_i) = \mathbf{tr}\ \Gamma$
6. $\Pi_{i=1}^{p} V(t_i) = |\Gamma|$
7. No standardized linear combination of x has a greater variance than t_1.
8. The ratio $\Sigma_{i=1}^{k} l_i / \Sigma_{i=1}^{p} l_i$ represents the proportion of the total variation explained by the first k principal components.
9. The principal components vary depending on the scaling of the data.

This last property is of a great deal of practical importance. Incorrect scaling of the data can greatly affect the apparent relationships among the variables. If nothing is known about the relative importance of the variables, common practice is to scale them all to unit variance. However, more insight into the underlying behavior of the process can be gained if any prior knowledge of the process available is used in the scaling of the data. Quality control measurements can be scaled using production specifications. Control input measurements can be scaled according to the controller range. Variables with the same physical meaning should be given the same scaling. For example, temperature measurements from each tray of a 20-tray distillation column or from a number of points along the length of a tubular reactor should all be scaled by the same factor.

In practice, the principal components are computed by an algorithmic method which does not require Γ to be computed explicitly. This iterative method (Wold et al., 1982) is more

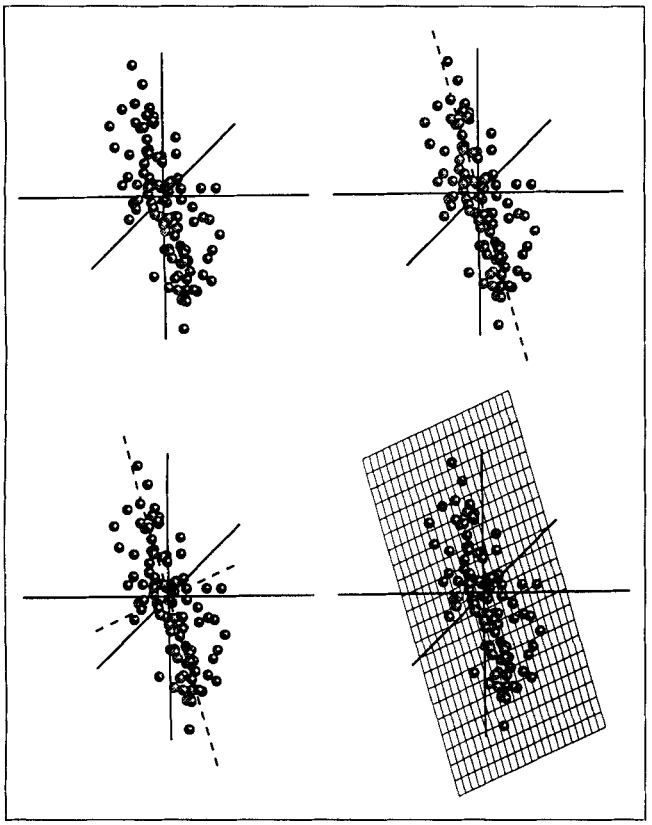

Figure 1. Geometric explanation of PCA.

At upper left, a swarm of data points in $\Re^3$ is shown. Each axis represents a measured variable of the process, and each point represents a measurement (that is, a point in time or an experiment). The loading vector of the first principal component points in the direction of greatest variability in the data, and the orthogonal projection of the data onto this loading vector is the first score vector. Similarly, the second loading vector points in the direction of greatest variability subject to the constraint that it be orthogonal to the first loading vector. These first two component loading vectors define the plane which is shown along with the data at lower right, and the projection of the data into this plane is the score plot for the first two components. Such a score plot represents a low dimensional window through which most of the behavior of interest can be observed.

efficient than computation and spectral decomposition of the covariance matrix because it is usually unnecessary to compute all of the components. Normally, only the first few components will be required to explain the majority of the variance in the data. Use of the iterative algorithm can thus result in significant savings when the number of variables is much larger than the number of significant principal components.

The principal component method for decomposing a data matrix is related to the singular value decomposition (SVD). The SVD of X is given by:

$$X = U\Sigma V' \tag{6}$$

where U has the normalized eigenvectors of XX' as its columns, V is similarly composed of the normalized eigenvectors of $X'X$, and Σ is a diagonal matrix having the positive square roots of the ordered eigenvalues of $X'X$ as its diagonal elements. Based on the previous definition of the principal component transformation and the principal component

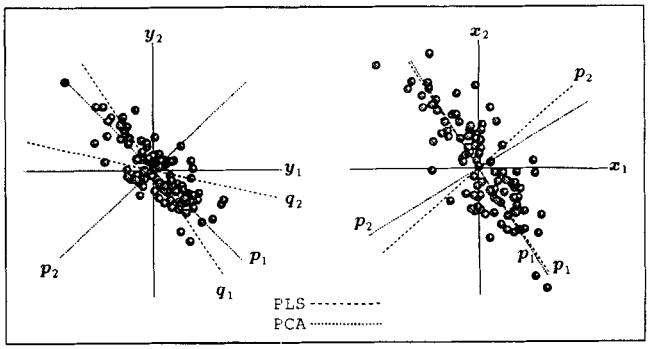

Figure 2. Rotation of PLS loading vectors.

The loading vectors in PCA are orthogonal. In PLS, these vectors are rotated, and the orthogonality is lost. Absence of orthogonality means the individual data matrices are not modeled in the most efficient possible way. However, the rotation allows a better model for the relation between the two data matrices (the inner relation) which minimizes $\|F\|$. In this way PLS is a compromise between PCA and CCA. PCA models the individual data matrices, and CCA models the relationship between them. PLS does both.

decomposition, it can be seen that the two methods are related by $P = V$ and $T = U\Sigma$.

The ideas of PCA lend themselves to graphical explanation. In the upper left portion of Figure 1, a data set in $\Re^3$ is shown. Coordinate axes represent the physical variables corresponding to the columns of the data matrix. Each point in the plot represents a row of a data matrix, X. The upper right portion shows the same data set along with a dashed line pointing in the direction of greatest variation, the slope of which is defined by the first loading vector. Projection of the data points onto this line gives the first score vector. The sum of the distances of all points from this line is a minimum in the least-squares sense. In addition to what was shown by the upper right portion, the lower left portion of the figure shows a dashed line whose slope is defined by the second loading vector. This line is orthogonal to the first and represents the direction of the second greatest variability in the data set. These two lines define a plane which is a manifold of the original data space and is represented by the grid in the lower right portion of the figure. Projecting the data points into this plane gives the score plot for the first two principal components. Such a score plot represents a low dimensional "window" through which most of the activity of interest in the system may be observed.

PLS

Partial least squares (PLS) is also known as projection to latent structures. It has a conceptual similarity to canonical correlation analysis (CCA) in that the relationship between two blocks of data are modeled to find combinations of variables which are highly correlated. At the same time, however, it selects these linear combinations in a way that eliminates redundancies in the data blocks and defines a new set of variables in each block which are highly independent. In this way it is an extension of PCA with the input components rotated to line up with the output components (see Figure 2). The rotation results in a relationship between the two sets of scores (the newly defined set of variables) which is stronger than that which would be obtained by the simplistic approach of applying

Table 1. PLS Algorithm

1. Start: set u equal to a column of Y
2. $w' = u'X/u'u$ (regress columns of X on u)
3. Normalize w to unit length
4. $t = Xw/w'w$ (calculate the scores)
5. $q' = t'Y/t't$ (regress columns of Y on t)
6. Normalize q to unit length
7. $u = Yq/q'q$ (calculate new u vector)
8. Check convergence: if YES to 9, if NO to 2
9. X loadings: $p = X't/t't$
10. Regression: $b = u't/t't$
11. Calculate residual matrices: $E = X - tp'$ and $F = Y - btq'$
12. To calculate the next set of latent vectors replace X and Y by E and F and repeat.

PCA to both data sets and regressing the scores of the output block on the scores of the input block.

PLS is described by the following equations, although the elements of the matrices are ordinarily computed using an iterative algorithm (Wold et al., 1982). The independent block, X, is decomposed into a score matrix, T, and a loading matrix, P. Similarly, the dependent block, Y, is decomposed into a score matrix, U, and a loading matrix, Q.

$$X = TP' + E \tag{7}$$

$$Y = UQ' + F^* \tag{8}$$

Thus, X and Y are represented as the sum of a series of rank one matrices, each given by the product of a score vector and the transpose of the corresponding loading vector ($t_i p_i'$ and $u_i q_i'$). The error matrices, E and F^*, can be made zero by including all the components. Usually, however, only the significant components are retained so that only meaningful information is included in the model. Equations 7 and 8 are the outer relations. An inner relation relates the two blocks. This inner relation is given by:

$$Y = TBQ' + F \tag{9}$$

The diagonal matrix, B, is selected so $\|F\|_2$ is minimized. Product TB in the inner relation is an estimator of U. F is the prediction error. The algorithm for computing the PLS model from a set of data is given in Table 1. This version of the algorithm is taken from MacGregor et al. (1991).

The PLS algorithm is explained graphically by Figure 3. The first set of loading vectors, p_1 and q_1, is determined using cross regression between X and Y. Projection of the data onto these vectors gives the first score vectors, t_1 and u_1. Matrices X and Y are related through their scores by the inner relation, $\hat{u}_1 = t_1 b_1$. The variance explained by the first component is removed from the data by subtracting $t_1 p_1'$ from X and $t_1 b_1 q_1'$ from Y. This process is repeated until all necessary components have been calculated. The decision to stop may be based on reduction of the residual variance to some predetermined fraction of the original variance, or more sophisticated approaches such as cross validation may be used.

Hoskuldsson gives a mathematical and statistical treatment of PLS in which the important vectors computed by the PLS algorithm are given in terms of eigenvectors of appropriate matrices, and some orthogonality relations are given among these vectors (Höskuldsson, 1988).

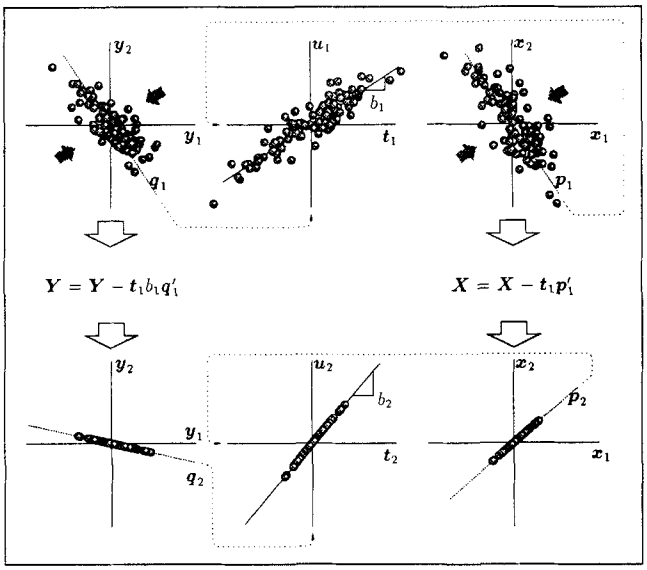

Figure 3. Geometric explanation of PLS.

The first component loading vectors, p_1 and q_1, are found for the input and output data. Projection of the data onto these vectors (grey arrows) gives the score vectors, t_1 and u_1. Relating the score vectors is a regression line with a slope of b_1. The variance represented by the first component is subtracted from both the X and Y matrices. An estimate of u_1 given by $t_1 b_1$ is used for the output so the variance explained by the prediction is what is removed from the data. This procedure is repeated for the second component. For this data set, two components explain 100% of the variance of both the input and the output.

Applications

This section discusses some methods of applying PCA and PLS. These methods include graphical techniques for data analysis and process monitoring, as well as techniques for dimensionality reduction. The techniques in the first two sections have been presented in some detail in the relevant literature. In the final section an extension of the basic ideas of one of the monitoring techniques to the identification of dynamic models and the use of the resulting models directly in the design of feedback controllers is introduced.

Graphical methods

PCA and PLS lend themselves to graphical interpretation due to the property they share of concentrating the variance of a data set in the first few dimensions of the new space they define. Two methods of data interpretation specifically tailored to these techniques are the loading plot and the score plot. These are especially useful when the first two components contain over 90% of the variance of the original data set.

A loading plot is obtained by plotting the second loading vector against the first. In PLS, plots are made of both the input loadings and the output loadings. The loading plot is used to find relationships between the original variables. Points that fall close together represent highly correlated pairs or groups of variables. Analysis using loading plots can guide the selection of manipulated and controlled variables.

A score plot is obtained in a manner similar to the loading plot. The second score vector is plotted against the first. In Figure 1, the projection of the data points onto the plane in

the bottom right portion represents the score plot for the first two components. The score plot is used to find relationships between the data points. Points that fall close together represent similar data points. Thus, the score plot is useful for classifying measurements. An additional axis is provided by the squared prediction error. In PLS this is calculated from the rows of the matrix, F. In PCA, it is calculated from the rows of the product of the eliminated score and loading vectors. The score plot has an interesting application in process monitoring. In the normal operation of the process, the points in the score plot will fill a region (or regions) consistently. When the points begin to fall outside this region, it is an indication the process is statistically "out of control." In this application, the score plot serves a similar function to the charts used in statistical quality control. It has the advantage of being able to show the entire process at a glance (rather than requiring the operators to look at hundreds of Shewart charts or CUSUM charts, for example). Another application of the score plot is in the classification of data points. This can be used to detect outliers in a data set or to identify unknowns characterized by a set of measurements (Musumarra et al., 1983).

Score and loading plots can be constructed when more than two components are required to explain the data. However, when more than three dimensions are required, interpretation of the plots becomes extremely difficult. In situations where four or more components are required, it is often better to partition the data according to operating units or some other criterion and perform analysis on each subset of the data matrix separately in an effort to exchange a single problem with four or more significant dimensions for two or more problems with two or three dimensions. Alternatively, graphical techniques can be abandoned in favor of mathematical techniques when the dimension is too large, or the graphical techniques can be augmented by mathematical techniques.

These interpretations and monitoring methods have been explored in some detail by MacGregor and coworkers (Jutan et al., 1977; Kresta et al., 1989; Roffel et al., 1989; Wright et al., 1977).

Dimensionality reduction

The techniques of PCA and PLS also provide means of dimensionality reduction. This reduction is accomplished in one of two fundamental ways, although there are a number of variations on each. Both of these approaches have been discussed in detail in the literature (Cooley and Lohnes, 1971; Geladi and Kowalski, 1986; Jutan et al., 1977; Kresta et al., 1989; Liao, 1989; MacGregor et al., 1991; Mardia et al., 1979; Musumarra et al., 1983; Press, 1972; Ricker, 1988; Roffel et al., 1989; Wise and Ricker, 1989; Wise et al., 1989; Wold et al., 1982, 1984a,b; Wright et al., 1977). The first approach is to rely on the first few components to define a subspace containing most of the variance of the original data. In this approach, the scores become a set of composite variables with desirable statistical properties. The second approach is to use PCA or PLS in any of a number of ways to select a subset of the original physical variables which are relatively uncorrelated and representative of the overall process.

When using the first approach, the question of how many components to retain must be answered. A number of approaches have been presented in the literature ranging from heuristic techniques such as retaining enough components to explain 90% of the variance to statistically founded techniques such as cross validation. When using the rigorous statistical methods it must be remembered that a component may be statistically significant while being irrelevant from a practical standpoint. This situation sometimes presents itself when a large data set reveals minor effects which, although they are statistically "real," have relatively little influence on the process as a whole. Wold et al. (1984a) discusses this point.

When using the second approach, the selection of variables is guided in a variety of ways using the loadings. One technique is to examine the loading plot for the first two or three components to detect the presence of clusters of variables. The variables belonging to each cluster are highly correlated, and little information is lost by selecting a single variable from each cluster to act as a representative for all the variables in that cluster. The other variables are removed from further consideration. An example of this approach is given by Musumarra in which the number of solvent systems required for drug identification by thin-layer chromatography is reduced from eight to three (Musumarra et al., 1983). Another technique is to select the variable whose weight in the first loading vector is the greatest in absolute value. The selected variable is then removed from the data set, and the PCA or PLS analysis is repeated on the remaining variables. A second variable is then selected using the same criterion as the first. This process is repeated until enough variables to adequately represent the process have been selected.

There are advantages and disadvantages to each of these basic approaches. The component selection approach is statistically preferable because of the desirable properties of the composite variables in the redefined space. However, some people have a strong preference for working with physical variables as opposed to composite variables with little or no physical meaning, and for these people the variable selection approaches are more appropriate.

Controlling the scores

As useful as PCA and PLS are for data analysis, dimensionality reduction, and process monitoring, another application presents itself in the design of multivariable feedback controllers. A natural extension of the PLS monitoring chart is to perform the computation of the control action based on the score variables. However, such an approach would likely raise objections from those who dislike the idea of controlling an artificial set of variables which, in general, have no physical significance.

Control design based on the scores can be implemented in a transparent way by using the loading matrices are pre- and postcompensators on the plant. Thus, the operator still "sees" the physical variables, while the controller "sees" the error signals and control signals in terms of new bases defined by the columns of the respective input and output loading matrices. This approach results in the top block diagram shown in Figure 4. In this diagram, W_y and W_x are the diagonal scaling matrices used before applying the PLS algorithm to the data. Q is the loading matrix for the output data block, and $Q^\dagger$ is the appropriate inverse of Q. If the number of retained components is equal to the number of output variables, Q is square and $Q^\dagger = Q^{-1}$. If there are more outputs than retained components, the appropriate inverse is the left inverse,

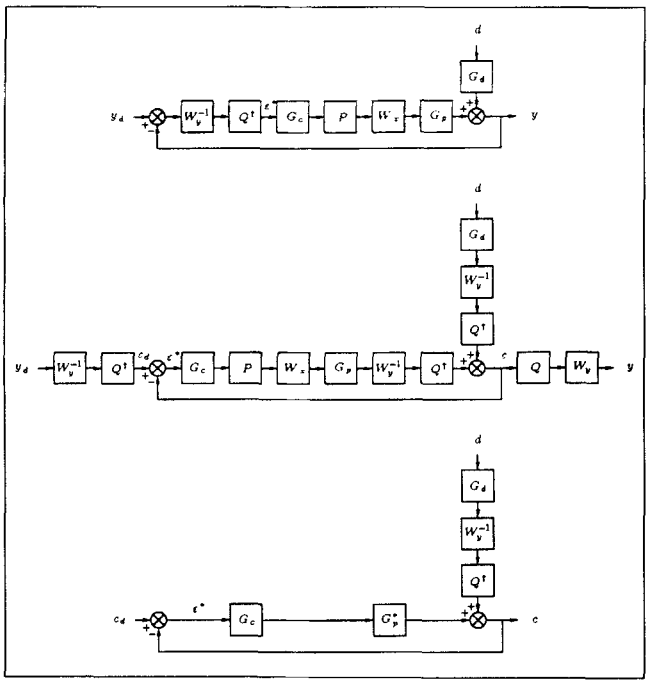

Figure 4. Block diagrams for PLS compensation.

$Q^- = (Q'Q)^{-1}Q'$. If the opposite condition exists and there are more retained components than outputs, the appropriate inverse is the right inverse, $Q^+ = Q'(QQ')^{-1}$. P is the loading matrix for the input data block. As in common process control nomenclature, G_c, G_p, and G_d are the transfer function matrices for the controller, plant, and disturbances, respectively. The block diagram can be rearranged as shown in the middle diagram of Figure 4, and after cutting the loop in the appropriate places and replacing $Q^+ W_y^{-1} G_p W_x P$ with G_p^* it reduces to the bottom diagram in Figure 4. This is the desired configuration. The controller is taking action to eliminate the error in terms of the basis defined by the columns of Q, and the output of the controller is given in terms of the basis for the physical inputs defined by the columns of P. If the PLS model is perfect (that is, $\|F\|$ is zero), the compensated plant, G_p^*, will be exactly equal to the inner relation of the PLS model. However, if the model is imperfect, the compensated plant will differ from the inner relation due to the modeling error.

Any of the commonly applied control design techniques can be applied within this compensation framework. Thus, the block represented generically by G_c might, in practice, be a controller designed using frequency response techniques, a direct synthesis controller, an internal model controller, a dynamic matrix controller, or anything else familiar to a particular control engineer and appropriate for the specific problem at hand.

Such a compensation scheme offers a number of positive features. The process is decoupled to a certain extent because of the orthogonality of the input scores and the rotation of the input scores to be highly correlated with the output scores. Because of the similarity between PLS and CCA, the scores represent a way of pairing the inputs and outputs which is in some sense optimal. In CCA, the first canonical variable from the input block can be thought of as the "best predictor," and the first canonical variable from the output block as the "most easily predicted" variable (Mardia et al., 1979). In PLS, then, with the data partitioned such that manipulated variables are in the X block and controlled variables are in the Y block, the first component of the input represents the set of variables which most strongly influences the output, and the first component of the output represents the set of variables which is the most easily controllable. Although this analogy between CCA and PLS is imperfect even for the first component and breaks down further with increasing component number (due to the additional objective PLS has of modeling the individual data blocks effectively), this property presents an advantage to be gained by pairing each of the input scores with the corresponding output score. In situations in which the true (or practical) dimension of the process is less than that of the measurement space, the possible dimensionality reduction offers at least two benefits. It simplifies the design of the controller (because fewer controller elements need be specified), and it prevents infeasible setpoints from reaching the controller. The process is free to move in any direction in the component space. Any physically impossible directions are eliminated when the dimension of the problem is reduced. Thus, if an infeasible setpoint is specified in terms of the physical variables, when it is projected into the component space only the feasible part is retained. The problem of controllers in a multiloop scheme "fighting" each other in a vain attempt to reach an impossible setpoint is eliminated. Nonsquare systems (both overdetermined and underdetermined) can be handled in a natural way by this compensation technique using the same concept of projecting the physical setpoints into the space defined using the output loading vectors as a basis.

When the variable space is redefined as the input and output scores by using the loading matrices as compensators, the inner relation of the PLS model becomes the appropriate model for use in control system design. In standard PLS, this model is a diagonal matrix of constants. For control system design, dynamic transfer functions would be more useful. These can be obtained in several different ways. The first approach is to replace the step in the PLS algorithm in which u_i is regressed on t_i (step 10 in Table 1) by a step in which a transfer function is found relating the output, u_i, to the input, t_i. (Note that step 11 is also altered by this change, because b is no longer a constant, but a dynamic transfer function.) This approach produces a diagonal transfer function matrix relating the output scores to the input scores. The second approach is to wait until the loading matrices have been computed and apply them as compensators on the input and output data sets (along with the appropriate scaling matrices) to give a set of transformed data corresponding to the plant, G_p^*, in Figure 4. A full square transfer function matrix relating the output scores to the input scores may be obtained by this approach. In applying the second method it can be advantageous to use the first method to obtain a preliminary model. The choice of inner relation will affect the second and higher-numbered loading vectors, and thus, the type of inner relation chosen during the PLS stage of the modeling will have a bearing on the final model even when the second approach is used.

A problem with these two approaches is that the appropriate structure for the transfer function elements comprising the inner relation may be quite complicated. The PLS algorithm attempts to find an outer relation which will produce a good

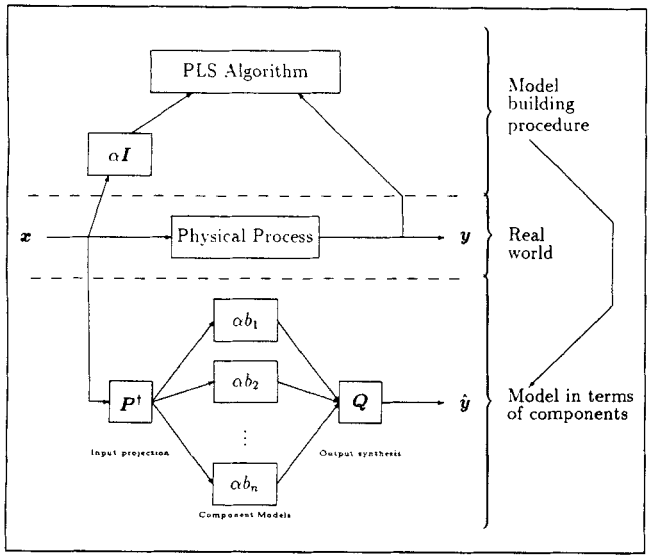

Figure 5. Modeling procedure and modal representation of model.

linear algebraic inner relation. When the true relationship between the input and output variables is dynamic, the attempt of the PLS algorithm to force the input and output scores to be colinear can actually be counterproductive. In some cases, the attempt may be successful with the result that a model with very fast dynamics in the most significant components is obtained, and any slow dynamics are modeled by the less significant components. In other cases, the modeling may be unsuccessful in that a bad choice of outer relations is found for which the inner relations are poor. This results in a poor model accounting for only a small fraction of the variance in the output data. What is needed to solve this problem is a way of presenting the PLS algorithm with data for which the appropriate inner relation is a constant diagonal matrix.

It is not unreasonable to assume that some knowledge of the approximate process dynamics will be available due to operating experience for existing processes or design parameters for new processes. At minimum, the relevant time scale should be available. If some idea of the dynamics of the process is available *a priori*, these dynamics can be used to filter the input data prior to modeling using standard PLS. This represents the third approach to incorporating dynamics in the PLS model. (See Figure 5.) The dynamic transformation represented by this filter has the effect of presenting the PLS algorithm with data from which the dynamics have been approximately removed. The result is a set of data which contains steady-state information and can be modeled effectively using PLS. The dynamics used to filter the inputs become part of the final model. In general, detailed knowledge of the process dynamics will be unknown at the time the filtering dynamics are selected. For this reason and because of the ease of manipulation and simplicity of the resulting model, a filter of the form αI is suggested, where α is a scalar transfer function containing what is considered the "average" dynamics of the process, and I is the identity matrix of appropriate dimension. (Optimization of the dynamics with the objective of minimizing $\|F\|$ in Eq. 9 would actually be more desirable than selecting them *a priori*. This approach will be investigated in future work.) If more detailed information about the dynamics is known, appropriate diagonal or square transfer function matrices may be applied to the input data. However, the complexity of the combination of the inner relation with the filter dynamics increases dramatically when general diagonal or square filters are used. Much of the benefit of the PLS compensation method of controller design lies in the simplicity of the inner relation as a plant model. Because this benefit is lost if the inner relation is made unnecessarily complicated, it is recommended that dynamics incorporated into the model by filtering the inputs be limited to the simple form, αI.

Even the simple αI form of the input transformation filter should result in a much better PLS fit of dynamic data than using the raw data. If the *a priori* dynamics are insufficient to adequately describe the data, however, additional dynamics may be incorporated by using a diagonal transfer function for the inner relation within the PLS algorithm or by identifying a square transfer function matrix model for the scores after the PLS loading matrices have been computed, as discussed previously.

In summary, the performance of a control scheme using the compensation tactic in Figure 4 is expected to have the following characteristics. It should be decoupled to an extent dependent on the fidelity of the PLS model. The performance (here loosely defined as speed of error elimination combined with minimization of control action, that is, a nonrigorous LQG objective) of the first component controller will be the best and the performance will degrade with increasing component number. This means the component of the setpoint change (disturbance) in the direction of the first component will be tracked (rejected) most easily, and the ease of tracking (rejection) will decrease with increasing component number. Although at first glance this seems like a serious problem, on additional thought it can actually be seen as an advantage. The first component represents the direction in the variable space in which the plant most naturally moves. Thus, disturbances and setpoint changes occurring most often are those controlled best, performance is sacrificed for those occurring infrequently, and control is not even attempted for those representing physically impossible combinations of process variables.

Example: Shell standard control problem

This problem was developed by Shell personnel for the first Shell Process Control Workshop (Prett and Morari, 1987). The physical system is a heavy oil fractionator. A nominal linear model with three controllable inputs, two disturbance inputs, and seven measurable outputs was provided. In addition, the control objectives, control constraints, model uncertainty description, and a set of prototype test cases were specified. A detailed description of the problem is given in the proceedings of the first Shell workshop (Prett and Morari, 1987).

The process is represented by a matrix of transfer functions of the following form:

$$\frac{Ke^{-\tau_d}}{\tau s+1}$$

The nominal parameters are given in Table 2. Two disturbance inputs, d_1 and d_2, are the intermediate reflux duty and the

Table 2. Model for Shell Problem

	Top Draw (TD)	Side Draw (SD)	Bottoms Reflux Duty (BRD)	Inter. Reflux Duty (IRD)	Upper Reflux Duty (URD)
Top End Point (TEP)	$K=4.05$ $\tau=50$ $\tau_d=27$	$K=1.77$ $\tau=60$ $\tau_d=28$	$K=5.88$ $\tau=50$ $\tau_d=27$	$K=1.20$ $\tau=45$ $\tau_d=27$	$K=1.44$ $\tau=40$ $\tau_d=27$
Side End Point (SEP)	$K=5.39$ $\tau=50$ $\tau_d=18$	$K=5.72$ $\tau=60$ $\tau_d=14$	$K=6.90$ $\tau=40$ $\tau_d=15$	$K=1.52$ $\tau=25$ $\tau_d=15$	$K=1.83$ $\tau=20$ $\tau_d=15$
Top Temp. (TT)	$K=3.66$ $\tau=9$ $\tau_d=2$	$K=1.65$ $\tau=30$ $\tau_d=20$	$K=5.53$ $\tau=40$ $\tau_d=2$	$K=1.16$ $\tau=11$ $\tau_d=0$	$K=1.27$ $\tau=6$ $\tau_d=0$
Upper Reflux Temp. (URT)	$K=5.92$ $\tau=12$ $\tau_d=11$	$K=2.54$ $\tau=27$ $\tau_d=12$	$K=8.10$ $\tau=20$ $\tau_d=2$	$K=1.73$ $\tau=5$ $\tau_d=0$	$K=1.79$ $\tau=19$ $\tau_d=0$
Side Draw Temp. (SDT)	$K=4.13$ $\tau=8$ $\tau_d=5$	$K=2.38$ $\tau=19$ $\tau_d=7$	$K=6.23$ $\tau=10$ $\tau_d=2$	$K=1.31$ $\tau=2$ $\tau_d=0$	$K=1.26$ $\tau=22$ $\tau_d=0$
Inter. Reflux Temp. (IRT)	$K=4.06$ $\tau=13$ $\tau_d=8$	$K=4.18$ $\tau=33$ $\tau_d=4$	$K=6.53$ $\tau=9$ $\tau_d=1$	$K=1.19$ $\tau=19$ $\tau_d=0$	$K=1.17$ $\tau=24$ $\tau_d=0$
Bottoms Reflux Temp. (BRT)	$K=4.38$ $\tau=33$ $\tau_d=20$	$K=4.42$ $\tau=44$ $\tau_d=22$	$K=7.20$ $\tau=19$ $\tau_d=0$	$K=1.14$ $\tau=27$ $\tau_d=0$	$K=1.26$ $\tau=32$ $\tau_d=0$

upper reflux duty, respectively. Three manipulated inputs are the top draw, side draw and bottoms reflux duty.

Production specifications are given only for the TEP and SEP. They are not allowed to deviate by more than ± 0.005 at steady state. In addition, the TEP is never to go outside the range ± 0.5 even during a transient response. No bottom product specification is given except for a constraint on the BRT, which is not allowed to go below -0.5. All the controllable inputs are constrained between -0.5 and 0.5. In addition, they are allowed to move at a rate of only 0.05 unit per minute. In order to maximize the steam make in the bottom of the column, the BRD is minimized. The fastest allowable sampling interval is 1 minute, and this is the sampling rate used for all data analysis and simulation.

Preliminary analysis and monitoring

The dimensionality reduction provided by chemometric methods is helpful in obtaining preliminary understanding of the relationships among the variables of a process. A data set representing "normal open loop operation" was generated using the CONSYD simulator SMXPO by forcing all three manipulated variables and the two disturbance variables with white noise of mean zero and standard deviation 0.01. (CONSYD is a computer-aided CONtrol SYstem Design software package. It was originally developed at the University of Wisconsin—Madison under the direction of Professors Morari and Ray. Contact Professor W. H. Ray for details.) This was assumed to represent the "process noise" inherent in the heavy oil fractionator. (Note that its relation to reality is unknown because neither the magnitude nor the covariance structure of the noise of the actual process was provided in the problem statement.) Such noise is assumed to have an unmeasurable origin and to arise spontaneously within the process. This is equivalent to assuming, in addition to the disturbance transfer function matrix specified in the problem statement, another disturbance transfer function matrix is present which is the combination (stacked side by side) of the specified plant and disturbance matrices. (See Figure 6.) The aggregate disturbance transfer matrix is constantly forced with a white noise vector of dimension five with each element having mean zero and standard deviation 0.01. Thus, there are no "inputs" to measure (the supplementary white noise disturbances are unmeasurable), and only the outputs are available for analysis. Under these circumstances (having only one block of variables available for analysis), PCA is the appropriate technique for analysis of the data.

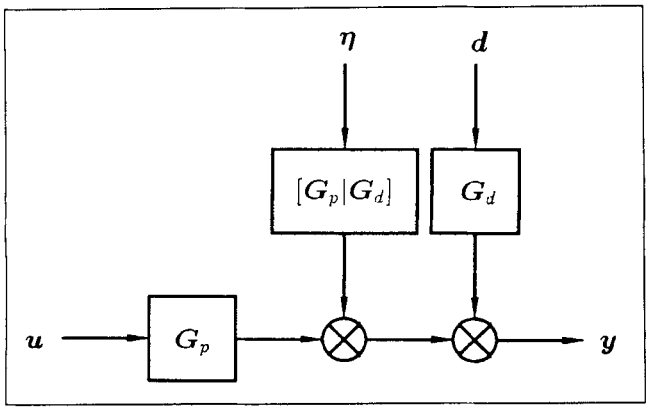

Figure 6. Input and disturbance block diagram.

A PCA model was constructed for the seven output variables by using CONSYD program CHEMOMETRIC to analyze the noise data. (This is a new program not included in the 1989 release. It will be included with the next release of CONSYD.) Figure 7 shows the variance explained by each of the seven components. The first three components explain about 87%

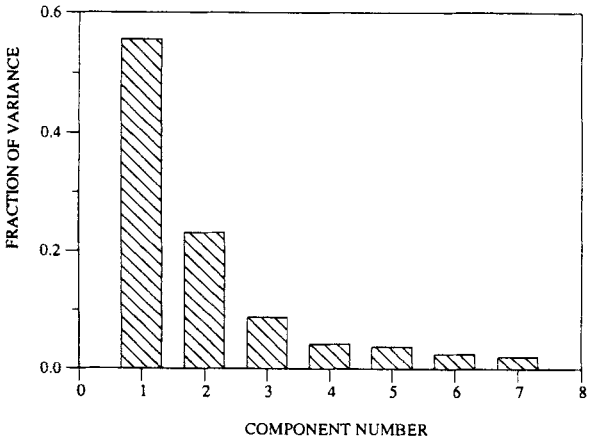

Figure 7. Variance explained by each component in PCA model.

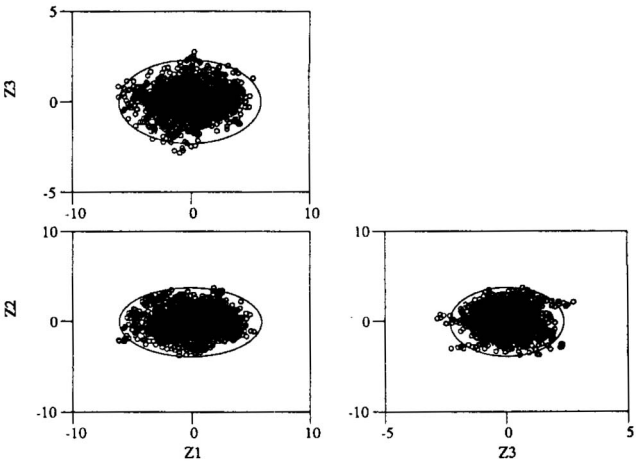

Figure 9. PCA score plot for noise data.

of the total variance of the data. Because the remaining four components together account for only about 13% of the variance, they are considered insignificant and eliminated from further analysis. A loading plot is constructed for the first three components in Figure 8. Two groups of points, each containable within a cylinder in the space defined by the first three components, are visible in the plot. According to the theoretical background of PCA, points falling in clusters on the loading plot should represent groups of related and therefore highly correlated variables. In this case, one group contains the two end point measurements, and the other contains the five temperatures. This result supports the idea that the PCA loading plot can be used for detecting groups of related variables because the technique has grouped the related variables together.

In the same way as the loading plot can be used to detect relationships between the variables, a score plot can be used to detect relationships between measurements. Application of the PCA score plot to process monitoring is a special case of this detection of relationships between measurements. Figure 9 shows the score plot of the first three components in the PCA model for the noise data. Most of the points fall in a well defined region on the plot (approximate 99% confidence region), which, using SPC terminology, indicates the process is statistically "in control." When the process is statistically "out of control," points should stray outside these well defined regions. Figure 9 shows the score plot in a scatter diagram format, which is how these are ordinarily presented in the literature. This representation can fail to convey the distributional nature of the score plot when the data points are sufficiently numerous so that significant numbers of them overlap in the center of the regions. More information can be conveyed if they are presented in a way that shows the density of the data, such as the grayscale score plot shown in Figure 10. This figure contains the same information as Figure 9, but it clearly shows that the density of the data is higher in the center of the control regions than at the edges. Figure 11 shows the score plot for a data set in which, in addition to the process noise, a mean shift of $+0.01$ was imposed on each of the two disturbances at $t = 1{,}000$. In the same, the disturbance is clearly visible. Measurements taken for $t > 1{,}000$ fall outside the previously defined control regions.

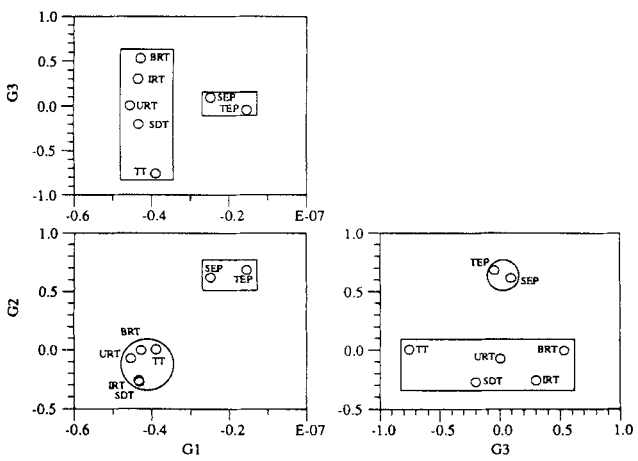

Figure 8. Loading plot for PCA model.

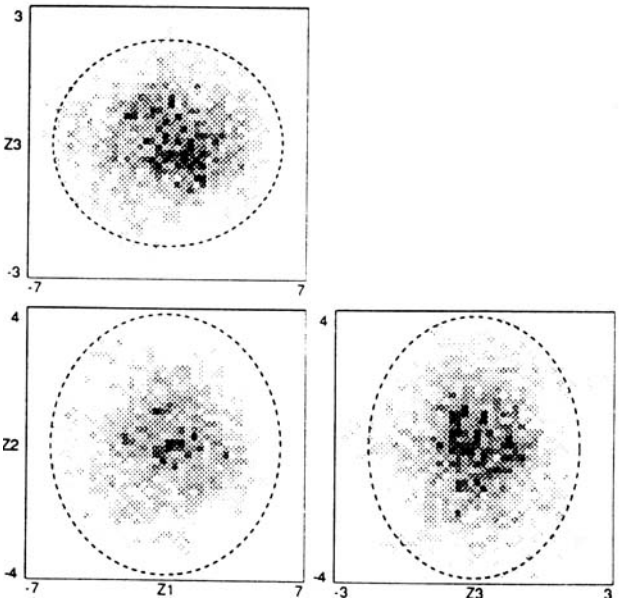

Figure 10. PCA score plot for noise data—Grayscale representation.

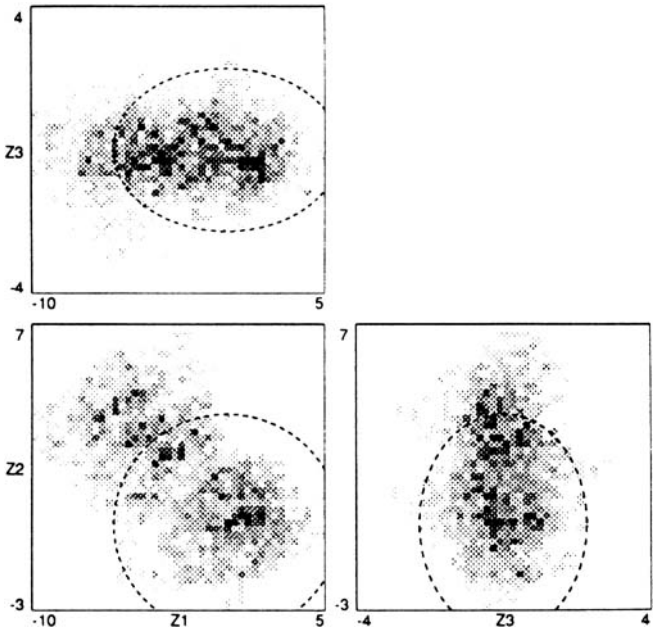

Figure 11. PCA score plot for disturbance data.

Modeling

Modeling is an integral part of any control system design, and this is especially true of the PLS compensation control scheme presented here. Although a model is provided in the Shell problem statement, for the purposes of this work it was assumed an appropriate model identification would be required. For the purpose of generating a data set for the identification, the nominal plant model was used in place of the physical system, and an "experiment" was performed on it. The model was simulated using CONSYD program SMXPO in open-loop mode. Three pseudorandom binary signals with clock intervals of 70 minutes were applied to the controllable inputs of the system, and the system response was recorded. The clock interval of 70 minutes was selected based on the assumed *a priori* knowledge that the dominant time scale of the process was approximately 70 minutes. In addition, the process noise used previously was included in the output. Thus, the identification is being performed in the presence of correlated noise.

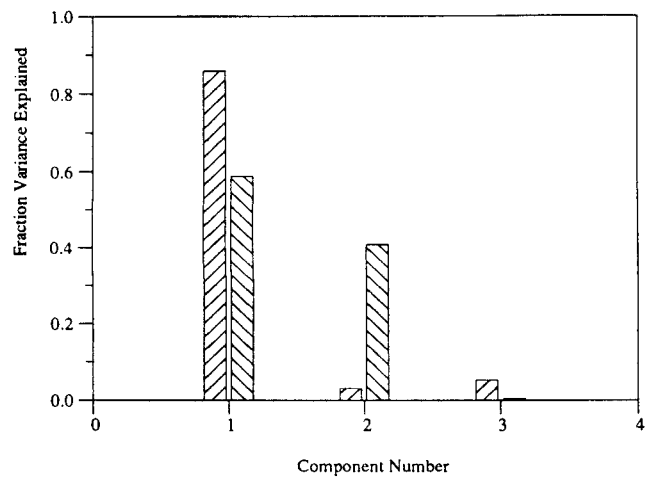

Figure 12. Fraction of variance explained by component—primary variables.

Left bar of each pair is output fraction, and right is input fraction.

The CONSYD program CHEMOMETRIC was used to model the data. The output variables were partitioned into two sets: a set of primary variables with production specifications given by the problem statement (TEP, SEP, and BRT) and a set of secondary variables with no specifications (TT, URT, SDT, IRT). All variables were centered around their steady-state values to facilitate the use of transfer function models. The TEP was scaled to unit variance, the SDT was scaled to a variance of 0.4, and the BRT was scaled to a variance of 0.1. These scalings were chosen to emphasize control of the TEP, which has a transient operating constraint which must be observed, and to de-emphasize the control of the BRT, which has no setpoint specification and is used only to optimize the BRD. The input variables (TD, SD, and BRD) were filtered by the transfer function matrix

$$\frac{e^{-20s}}{40s+1}I$$

to incorporate these dynamics into the model. These dynamics were selected by examining the parameters in Table 2 and taking a crude "eyeball" average of the values corresponding to the three variables which are to be controlled. In practice, the filter would be chosen based on the modeler's preliminary knowledge of the dominant time scale of the process. A standard PLS model was then constructed to relate the outputs to the resulting transformed inputs.

The inner relation obtained for the primary variables was:

$$\boldsymbol{B}=\begin{bmatrix} 40.72 & & \\ & 9.46 & \\ & & 125.35 \end{bmatrix}$$

The input and output loading matrices were:

$$\boldsymbol{Q}=\begin{bmatrix} 0.858 & 0.038 & 0.910 \\ 0.451 & 0.890 & 0.410 \\ 0.245 & 0.454 & 0.056 \end{bmatrix}$$

and

$$\boldsymbol{P}=\begin{bmatrix} -0.046 & 0.040 & 0.995 \\ 0.259 & 0.939 & -0.079 \\ 0.965 & -0.341 & 0.053 \end{bmatrix}$$

Figure 12 shows the fraction of the variance explained by each component of the PLS model for the primary variables. The first component accounts for about 85% of the variation in the output. Figure 13 shows the PLS model prediction for the three primary variables. Some error is visible in the fit, but it is quite good considering the PLS model contains only five parameters, two of which (the time constant and delay in the transfer function used to transform the inputs) were selected *a priori*. The original plant model for these three variables

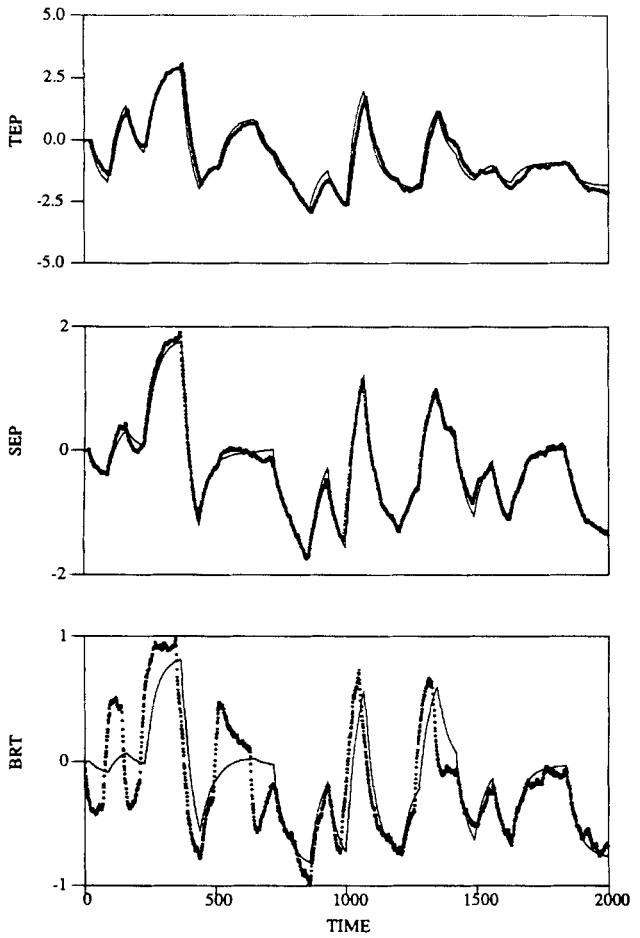

Figure 13. PLS model prediction for primary variables.

contains 27 parameters. For the TEP and SEP, the fit is much better than for the BRT. Lack of fit in this variable is due to the low weight it was assigned using the PLS scaling factors, but this is acceptable because it is used only to optimize the steam make, while the other two variables have setpoint specifications.

Additional dynamics could have been incorporated in both the primary and secondary variable models by replacing the static inner relations by dynamic ones. (Program CHEMOMETRIC has a feature for allowing the user to apply an arbitrary inner relation in place of the diagonal matrix of constants used in standard PLS. The ability to use an arbitrary function to relate the input and output scores is obtained by linking a user-written subroutine to the main program. This subroutine includes the input and output score vectors of the current component among its arguments. Thus, it can compute or estimate the parameters of, an arbitrarily complicated function relating these score vectors. This approach is similar to that used for analyzing and simulating nonlinear processes in other CONSYD programs.) This would have improved the fit slightly, but was considered unjustified. The slight improvement would not be worth the added complexity.

Control

The basic PLS control scheme is based on the model for the primary variables. A precompensator and postcompensator, (W_xP) and $(Q^\dagger W_y^{-1})$, respectively, were computed. (Recall that W_x and W_y are the scaling matrices applied to the data before implementing the PLS algorithm. Thus, they must be applied in the control system to keep the scaling consistent.) The precompensator and postcompensator were:

$$(W_xP) = \begin{bmatrix} -0.09168 & 0.08025 & 1.99092 \\ 0.12954 & 0.46967 & -0.03950 \\ 0.48238 & -0.17033 & 0.02653 \end{bmatrix}$$

and

$$(Q^\dagger W_y^{-1}) = \begin{bmatrix} 15.0907 & -17.0790 & 17.9166 \\ -8.3399 & 7.2954 & -1.3278 \\ 1.4978 & 15.7961 & -16.8319 \end{bmatrix}$$

The model for the compensated plant includes the B matrix from the PLS model (in this case, a diagonal constant matrix) and the dynamics used to transform the input variables:

$$\frac{e^{-20s}}{40s+1}\begin{bmatrix} 40.72 & & \\ & 9.46 & \\ & & 125.35 \end{bmatrix}.$$

This is the PLS model obtained previously. Based on this model, the control system will be designed. Some mismatch exists between the actual compensated plant and the model. The steady-state gain matrix of the compensated plant is:

$$G_p^*(0) = (Q^\dagger W_y^{-1})G_p(0)(W_xP)$$

$$= \begin{bmatrix} 44.89 & -9.34 & 96.97 \\ -1.23 & 11.29 & -1.59 \\ -.83 & 10.72 & 33.98 \end{bmatrix} \quad (10)$$

The first component shows good agreement between the plant and the model (44.89 for the plant vs. 40.72 for the model). The second and third components are much poorer, showing disagreement between the diagonal elements of the plant and the model as well as off-diagonal elements of significant magnitude. Overall, the plant/model agreement seems bad until the relative amounts of variance explained by the components is considered. The first component accounts for over 85% of the total variance and is modeled reasonably well. Validity of the model has already been demonstrated in that the input/output error was relatively small. Thus, the mismatch in the second and third components is relatively unimportant.

A simple rule-of-thumb based on the direct synthesis method of controller design was used to tune PI controllers on each of the three loops. Direct synthesis controllers are designed by specifying the closed-loop transfer function of the process (Seborg et al., 1989). Together with the model of the open-loop process, which is also given in the form of a transfer function, the required transfer function for the controller can be computer. If a first-order closed-loop response is specified for a first-order open-loop process, the resulting controller is of the PI form with the integral time constant equal to the time constant of the process and the controller gain a simple func-

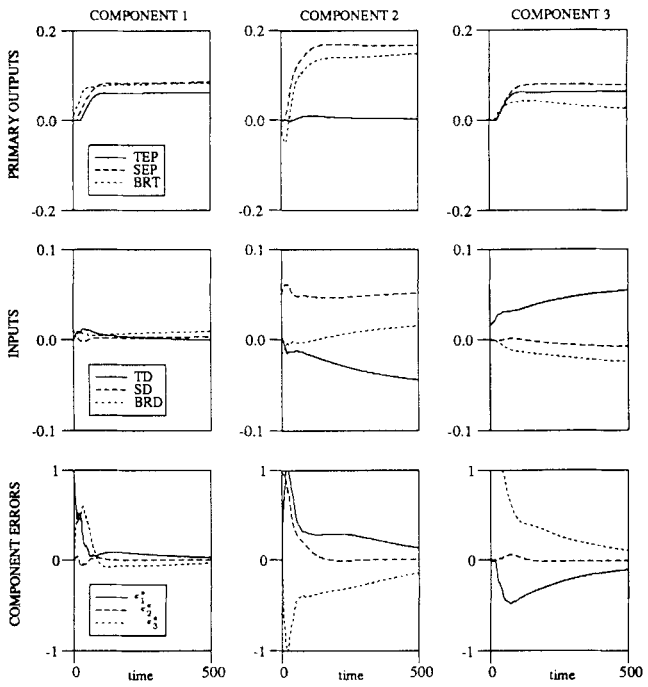

Figure 14. Directional responses.

The first column of plots is associated with a unit step in the first component error. The second and third columns are similar associated with steps in the second and third components, respectively. Note the speed of the closed-loop response as well as the small magnitude of required control action and lack of interaction between the three components when the process is moved in the first component direction, as opposed to the other two component directions.

tion of the process time constant, the process gain, and the specified time constant for the first-order closed-loop response. For this process, the model for each element is not simply first-order, but first-order with dead time. Thus, a rule-of-thumb was used which is justified elsewhere (Kaspar, 1992). This rule was:

$$K_c = \frac{\tau + \tau_d}{K_p \theta}$$

and

$$\tau_I = \tau + \tau_d$$

where τ is the process time constant, τ_d is the process delay, K_p is the process gain, K_c and τ_I are the PI tuning parameters, and θ is a pseudo tuning parameter which can be thought of as the time scale of the desired response. For this example, $\theta = 60$ was selected. This approach gave the following controller:

$$G_c = \left(1 + \frac{0.0167}{s}\right) \begin{bmatrix} 0.02456 & & \\ & 0.10571 & \\ & & 0.00798 \end{bmatrix}.$$

Using this controller and the compensators, the control system shown in Figure 4 was implemented using an updated version of CONSYD program SMXPO.

Exploration into how the performance of the control system depends on the direction in which the system perturbed is instructive. This perturbation can correspond either to a setpoint change or to a disturbance. For purposes of illustration, three setpoint changes were introduced into the process. The changes correspond to introduction of unit error into each of the three components (that is, to making a unit step change in each element of c_d in the bottom diagram in Figure 4). Figure 14 shows the directional responses of the system. Columns correspond to the three component directions (first to third components are shown from left to right). The first row corresponds to the output variables, the second row to the input variables, and the third row to the component error signals. This figure illustrates a number of points. Ideally, there should be no interaction between the components, and the response of each component should be independent of the other two. However, the examination of the steady-state gain matrix of the actual compensated plant indicates that some interaction can be expected in this case due to the plant/model mismatch. This interaction can be observed in Figure 14. If there were no interaction, in the bottom left plot, which shows the component errors when the system is responding to a setpoint change in the direction of the first component, the dotted and dashed lines would remain at zero, while the solid line (representing the first component error) would start at one and go to zero as the control system acted. Similarly, in the second and third columns of the bottom row, only the component corresponding to the perturbation direction would deviate from zero. However, it can be seen that all of the component errors are disturbed by setpoint changes in the direction of any single component. This interaction is the result of modeling error. More precisely, it is due to the fact the simple model structure selected is incapable of representing the true plant perfectly. The second point illustrated by this diagram is the idea that performance will degrade with increasing component number. This is seen clearly in both the top row of plots, which show the physical variables, and the bottom row of plots, which show the component errors. For the setpoint change in the direction of the first component, the variables and the component errors reach steady state very quickly. For the second and third components, however, the response is more sluggish and the interaction between the components is worse. The third point illustrated by this diagram is the idea that the required control action will increase in magnitude with increasing component number. This can be seen in the middle row of plots. For the first component, very little control action is required. For the second component, much larger control action is required. For the third component, even more control action is required, and the system is still far from steady state even after 500 minutes.

Figure 15 shows the response of the nominal system under PLS control to the disturbances and setpoint changes specified for the nominal case in the Shell Control Problem. The disturbances were each steps of $+0.5$, and the setpoint for the BRT was reduced to its lower operating constraint of -0.5. Control action brings TEP and SEP to their predisturbance values. As required to maximize the steam make, the BRT is driven to its lower operating constraint.

It was expected that the application of more advanced control strategies within the PLS compensation framework would give improved performance over the diagonal PI controller.

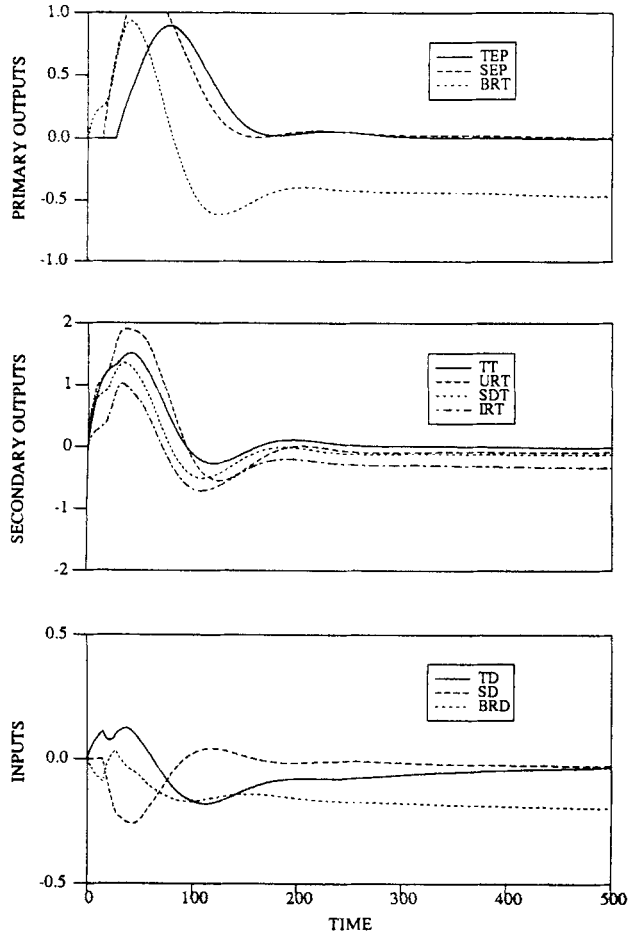

Figure 15. Closed-loop response—PLS control scheme.

After the introduction of steps in both disturbance variables at $t=0$, the TEP and SEP are brought back to their setpoints. As required to maximize the steam make, the BRT is driven to its lower operating constraint. The manipulated variables move in a smooth manner characteristic of a conservative controller.

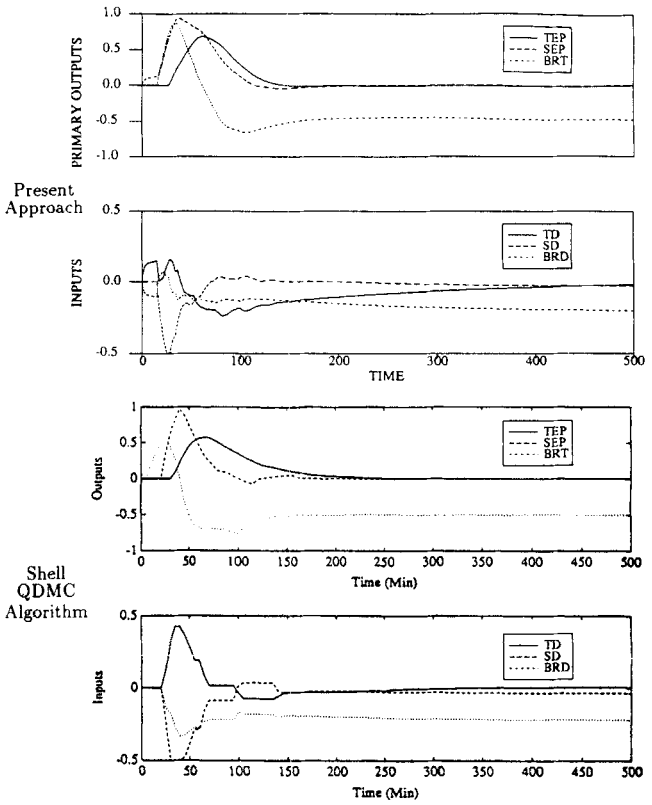

Figure 16. PLS Smith predictor control scheme and comparison with QDMC results.

The top pair of plots show the results obtained using the current approach, while the bottom pair shows the results of Cuthrell obtained using a QDMC controller (Cuthrell et al., 1990).

As a preliminary investigation into the performance of the PLS compensation scheme when more advanced control methods are used within this framework, a Smith predictor—the earliest and simplest of the model-predictive control algorithms—was implemented on the transformed variables (as shown in Figure 4). Use of the Smith predictor moves the delay outside the characteristic equation. This permits tighter tuning of the controller than would otherwise be possible. A PI controller was used with the Smith predictor. The tuning parameters of the PI controller were again computed using the direct synthesis approach. Because the delay can now be considered as outside the loop, only the first-order lag of 40 minutes is considered in computing the tuning parameters. The controller can be tuned much more tightly than without the Smith predictor, so a characteristic time of $\theta=10$ was selected. The diagonal PI controller used was:

$$\left(1+\frac{0.025}{s}\right)\begin{bmatrix}0.09823 & & \\ & 0.42283 & \\ & & 0.03191\end{bmatrix}$$

Note that this is a much tighter controller than was used without the Smith predictor. Figure 16 shows the response of the system to the same disturbances and steam make optimization as before. The settling time and transient maxima are very similar to those for the QDMC controller, although the optimization of the steam make it not as aggressive.

Monitoring the Controlled Process

It is of interest to monitor the controlled system both to determine how well the control system is performing and to detect any disturbances entering the process. The first objective may be met by looking at PCA score plots. Figure 17 shows the score plot of the closed-loop system using the PI controller without the Smith predictor as it responds to the inherent process noise. Figure 18 shows the closed-loop system as it responds to both the process noise and step changes of $+0.01$ in both disturbance variables occurring at $t=1{,}000$. In Figure 17, the points are found to fall within essentially the same regions as was the case in Figure 9. Thus, the process is statistically "in control." In Figure 18, which shows the score plot for the closed-loop process responding to both the process noise and the step disturbance, the points are also found to fall within the previously defined regions. This result contrasts with that shown in Figure 11, in which, for the open-loop response of the process to the same combination of process noise and disturbances, points fall clearly outside the control regions after the disturbance enters the system. Thus, the disturbance is effectively rejected by the control system, and the

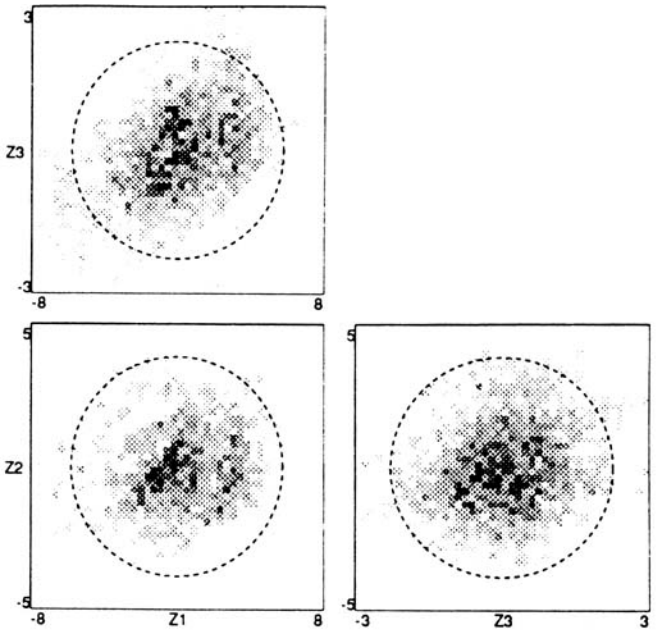

Figure 17. PCA score plot for noise data with control.

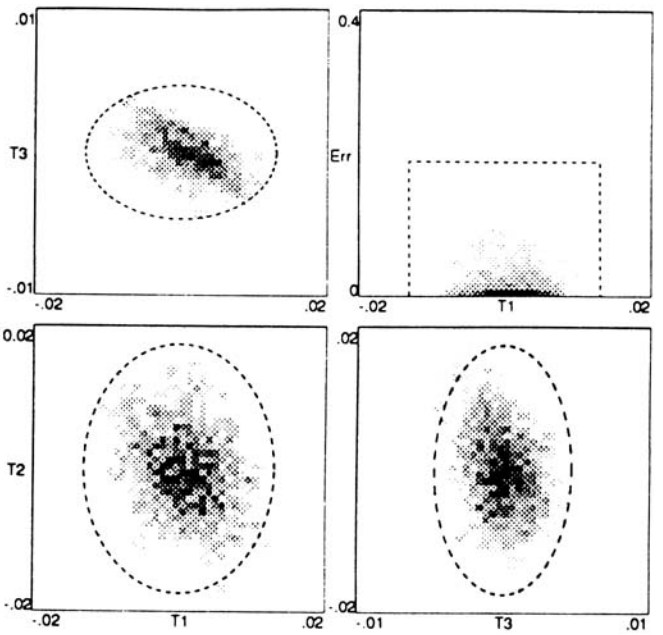

Figure 19. PLS score plot for noise data with control.

system remains in statistical control in spite of the disturbance.

Because the control system keeps the process "in control" according to the PCA monitoring method in the presence of disturbances, some other monitoring method must be used to detect the disturbances. An obvious candidate is an input score plot based on the same PLS model as used to design the controller. Figure 19 shows the PLS score plot for the closed-loop system responding to the process noise. As with the PCA score plots, the points fall within well-defined regions. The prediction error at upper right also falls within a well-defined region. Figure 20 shows the PLS score plot for the closed-loop system responding to the small step disturbances in addition to the process noise. These disturbances manifest themselves in the points on the score axes beginning to fall outside the control regions and the prediction error increasing dramatically after the introduction of the disturbances at $t = 1{,}000$. Thus, the PLS score plot is capable of detecting disturbances entering a closed-loop system even when the controller works well to keep the process under control.

Summary and Discussion

The PCA and PLS techniques have been applied to the Shell Standard Control Problem. The usefulness of these techniques

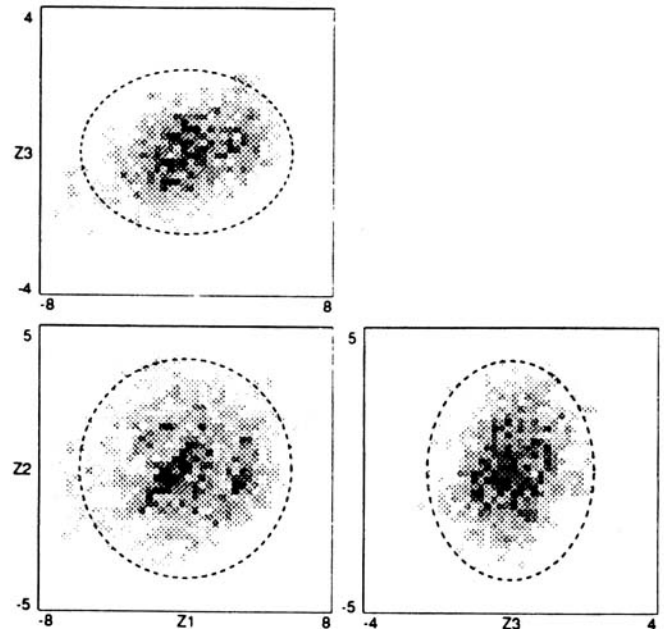

Figure 18. PCA score plot for disturbance data with control.

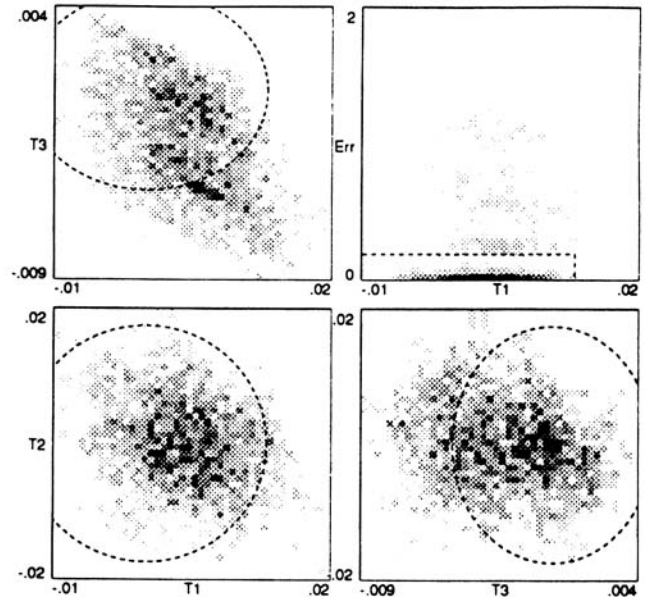

Figure 20. PLS score plot for disturbance data with control.

for preliminary data analysis, monitoring the process (both before and after control system implementation), dynamic modeling, and designing the control system has been demonstrated.

A PCA model was developed from a data set consisting only of "process noise." The loading plot for this model displayed two distinct clusters of points representing the seven variables of the system. The draw and reflux temperatures were found in one cluster and the end point temperatures in the other. Thus, it was demonstrated that physically related (and therefore statistically correlated) variables exhibit clustering in a loading plot, and that such a plot can be used as a tool for detecting such relationships.

The score plot for this same PCA model was used for monitoring the open-loop process. It was shown that control regions can be established on such a score plot using normal operating data and that disturbances which depart from the norm will cause points on the score plot to fall outside these control regions.

A dynamic model was developed for the process by transforming the inputs using a diagonal filter containing the appropriate "average" dynamics of the process. These dynamics were selected *a priori* and were not optimized to obtain a better fit of the data. The filtered inputs and the outputs were then related by a standard PLS model. Because the filtering dynamics used were of the special form of a scalar transfer function multiplying an identity matrix of appropriate dimension, the dynamics could be easily moved "inside" the PLS model and combined with the algebraic PLS inner relation to give a dynamic diagonal inner relation. The five parameter (two of them selected *a priori*) PLS model fit the data quite well.

This dynamic PLS model was used to design a feedback control system for the process. The input and output loading matrices served in the role of pre- and postcompensators so that the appropriate model for control system design was the diagonal dynamic inner relation. It was shown that, due to modeling error, the actual compensated plant differed from the inner relation, but the bulk of the disagreement was in the second and third components. Because of the small contribution of these components to the total variance of the process, the departure was considered tolerable. A simple diagonal PI controller was designed based on the inner relation. Rules-of-thumb derived from the direct synthesis approach to controller design were used to compute the controller tuning parameters. The servo response of the closed-loop system in each of the three component directions was examined. As expected, the response was fastest and required the least control action when the system was perturbed in the direction of the first component. The second and third components showed slower responses and required stronger control action. There was some interaction between the components due to the plant/model mismatch. A preliminary investigation into the applicability of more advanced control strategies within this compensation framework was made by implementing a Smith predictor. The performance of this control scheme was roughly comparable to the performance of the QDMC controller from the literature (Cuthrell et al., 1990). It will be interesting to see the results of future work in which the application of more advanced control algorithms such as model-predictive control and robust control within the PLS compensation structure will be considered.

Use of the PCA model to monitor the closed-loop process showed that the controller effectively rejected disturbances and maintained the system within the control regions. However, detection of the disturbances required using the PLS input score plot. It was shown that disturbances manifest themselves clearly in the prediction error portion of the PLS score plot, and thus they can be detected easily even when they are too small to significantly affect the output variables.

Conclusions

In addition to their well-established role as analysis and modeling tools, chemometric methods such as PCA and PLS can be useful in control system design. Their utility arises from the desirable properties of these models with respect to dimensionality reduction and concentration of the variance of the process in the first few components. The benefits include partial decoupling, automatic "efficient" loop pairing, and simplicity of design.

In the future it is expected the benefits will be shown to extend to natural handling of nonsquare and poorly conditioned systems, and the use of more advanced control strategies within this compensation framework will result in further performance and robustness improvements. Additional improvements are expected from better ways of incorporating dynamics into the PLS model. It should even be possible to extend the method to nonlinear systems by suitable modification of these modeling techniques. Additionally, other dynamic representations, such as discrete time models, could be used instead of the continuous transfer function as variations on the standard algebraic PLS inner relation.

Acknowledgments

Financial support for this work was provided by grants from the U.S. Department of Energy, E. I. DuPont de Nemours and Co., and the Dow Chemical Company.

Notation

B = inner relation in PLS model
E = residual of the input matrix in PLS
F = prediction error in PLS
F^* = residual of the output matrix in PLS
G_c = controller transfer function matrix
G_d = disturbance transfer function matrix
G_p = process transfer function matrix
G_p^* = transfer function matrix describing the compensated plant
I = identity matrix
L = diagonal matrix of eigenvalues in the spectral decomposition
P = loading matrix for the inputs in PCA and PLS
Q = loading matrix for the outputs in PLS
t = time
T = input score matrix
U = left singular vectors in SVD context; the scores of the output data in PLS context
V = the right singular vectors in SVD
W_x = matrix of scaling factors for the inputs
W_y = matrix of scaling factors for the outputs
X = input data matrix
Y = output data matrix
$'$ = matrix transpose
$\hat{}$ = estimate

Greek letters

α = transfer function for filtering inputs

Γ = covariance matrix of a stochastic process
Σ = matrix of singular values
θ = tuning parameter for specifying closed-loop time constant

Vector/matrix convention

The convention for X, Y, T, and U
x = vector of functions of time for which X is a particular realization
x_i = ith column of X
x_i = function of time for which x_i is a particular realization
The convention of P and Q
p_i = ith column of P
The convention for Σ, L, and B
σ_i = ith diagonal element of Σ

Literature Cited

Cooley, W. W., and P. R. Lohnes, *Multivariate Data Analysis*, Wiley, New York (1971).

Cuthrell, J. E., D. E. Rivera, W. J. Schmidt, and J. A. Vegeais, "Solution of the Shell Standard Control Problem," *The Second Shell Process Control Workshop*, p. 27, D. M. Prett, C. E. Garcia, and B. L. Ramaker, eds., Butterworth, Stoneham, MA (1990).

Geladi, P., and B. R. Kowalski, "Partial Least-Squares Regression: a Tutorial," *Analytica Chimica Acta*, **185**, 1 (1986).

Höskuldsson, A., "PLS Regression Methods," *J. of Chemometrics*, **2**, 211 (1988).

Jutan, A., J. F. MacGregor, and J. D. Wright, "Multivariate Computer Control of a Butane Hydrogenolysis Reactor: II. Data Collection, Parameter Estimation, and Stochastic Disturbance Identification," *AIChE J.*, **23**(5), 742 (1977).

Kaspar, M. H., "Model Identification for Chemical Process Control," PhD Thesis, Univ. of Wisconsin—Madison (1992).

Kresta, J., J. F. MacGregor, and T. E. Marlin, "Multivariate Statistical Monitoring of Process Operating Performance," AIChE Meeting (1989).

Liao, J. C., "Fermentation Data Analysis and State Estimation in the Presence of Incomplete Mass Balance," *Biotechnol. and Bioeng.*, **33**, 613 (1989).

MacGregor, J. F., T. E. Marlin, J. Kresta, and B. Skagerberg, "Multivariate Statistical Methods in Process Analysis and Control," *CPC-IV* (1991).

Mardia, K. V., J. T. Kent, and L. M. Bibby, *Multivariate Analysis*, Academic Press, New York (1979).

Musumarra, G., G. Scarlata, G. Romano, and S. Clementi, "Identification of Drugs by Principal Component Analysis of r_f Data Obtained by TLC in Different Eluent Systems," *J. Analytical Toxicol.*, **7**, 286 (1983).

Press, S. J., *Applied Multivariate Analysis*, Holt, Rinehart, and Winston, New York (1972).

Prett, D. M., and M. Morari, eds., *Shell Process Control Workshop*, Stoneham, MA. Shell Development Co., Butterworth Publishers, *Proc. of the Shell Process Control Workshop*, Houston (Dec. 15-18, 1987).

Ricker, N. L., "The Use of Biased Least-Squares Estimators for Parameters in Discrete-Time Pulse Response Models," *Ind. and Eng. Chemistry Res.*, **27**, 343 (1988).

Roffel, J. J., J. F. MacGregor, and T. W. Hoffman, "The Design and Implementation of a Multivariable Controller for a Continuous Polybutadiene Polymerization Train," *Proc. IFAC Symp. on Dynamics and Control of Chemical Reactors, Distillation Columns, and Batch Processes*, Pergamon Press (1989).

Seborg, D. E., T. F. Edgar, and D. A. Mellichamp, *Process Dynamics and Control*, Wiley, New York (1989).

Wise, B. M., and N. L. Ricker, "Feedback Strategies in Multiple Sensor Systems," *AIChE Symp. Ser.*, **85**(267), 19 (1989).

Wise, B. M., N. L. Ricker, and D. J. Veltkamp, "Upset and Sensor Failure Detection in Multivariate Processes," Symp. on Statistics and Quality Control, AIChE Meeting (Nov. 5-10, 1989).

Wold, S., et al., "Pattern Recognition: Finding and Using Regularities in Multivariate Data," *Food Research and Data Analysis*, Applied Science, London (1982).

Wold, S., et al., Multivariate Data Analysis in Chemistry. In B. R. Kowalski (ed.), *Chemometrics. Mathematics and Statistics in Chemistry* (pp. 17-95). D. Reidel Publishing Company (1984).

Wold, S., A. Ruhe, H. Wold, and W. J. Dunn III, "The Collinearity Problem in Linear Regression: the Partial Least Squares (PLS) Approach to Generalized Inverses," *SIAM J. of Scientific and Statistical Comput.*, **5**(3), 735 (1984b).

Wright, J. D., J. F. MacGregor, A. Jutan, J. P. Tremblay, and A. Wong, "Inferential Control of an Exothermic Packed Bed Reactor," *Proc. Joint Automatic Control Conf.*, p. 1516, San Francisco (1977).

Manuscript received May 18, 1992, and revision received June 10, 1992.