

Building Multivariate Models from Compressed Data

Syed A. Intiaz, M. A. A. Shoukat Choudhury, Sirish L. Shah*

Department of Chemical & Materials Engineering,

University of Alberta, Edmonton, Canada T6G 2G6

Abstract

The effect of data compression on multivariate analysis, especially on PCA based modelling, is evaluated in this paper. A comparative study between the ‘Swinging Door’ and ‘Wavelet compression’ algorithms is performed in the context of multivariate data analysis. It is demonstrated that Wavelet compression preserves the correlation between different variables better than Swinging Door compression. It is also demonstrated that the impact of compression increases as the dynamics of the processes become more faster and more stochastic in nature. Instead of interpolation based reconstruction of ‘Swinging Door’ compressed data and subsequent modelling, an iterative missing data technique is suggested for building PCA model from Swinging Door compressed data. The performance of the proposed methodology is demonstrated using a simulated flow-network system and an industrial data set.

Keywords: Missing Data, PCA, Data Compression, Wavelet Compression, Multivariate Analysis

1 Introduction

Data compression is a widely used practice in process industries. The current industrial practice in data archiving is to archive or store compressed data using the vendor supplied compression

*Corresponding author: Sirish L. Shah, Tel: +1 780 492 5162; Fax: +1 780 492 2881; E-mail address: sirish.shah@ualberta.ca

algorithms. As the name suggests the main objective of compression is to compress data file to reduce the size of data file so that storage space is minimized or reduced. Compression is now redundant since storage is relatively inexpensive yet industrial practitioners continue to compress data as a default practice. However, if the main purpose of data compression is to facilitate transmission of data through telecommunication or satellites then data compression can be justified.

Whenever possible, we suggest using uncompressed data for any analysis. However, in many situations when historical data has to be analyzed for investigative purposes such as post-mortem of faults, one may have no choice other than using compressed data from the data historian. In many other situations we may be required to use compressed data for analysis for reasons such as:(1) data analyst may be located at a remote place and it may not be possible to reset the compression factor and collect uncompressed data for analysis; (2) sometimes it may be of interest to compare current performance index of a control loop with the historical performance index of the loop when the controller was originally tuned. For calculating the past performance index one has to rely on compressed data obtained from the process historian.

Although compressed data is regularly used for different analysis, it is also well known that analysis of compressed data can lead to erroneous results in data based analysis. The effect of data compression on various univariate statistics, such as, mean, standard deviation, as well as various loop performance indicators are well studied¹. The effect of data compression on pattern matching was studied by². In their study, the data compression algorithms were assessed on the basis of not only how accurately they represent process data but also how they affect the identification of similar patterns from historical data. However, to the best knowledge of the authors, the effect of compression on multivariate data analysis and model building has not been studied so far.

The data historian currently used in industries mostly use direct methods (for example, Swinging Door data compression) for compressing data. Such compressed data are usually reconstructed using univariate methods, such as, linear interpolation. These reconstruction methods do not take into account the changes that take place in other variables, and as such linear interpolation-based data reconstruction algorithms may destroy the correlation between different signals. So the reconstruction may not be reliable depending on the end use of the data. In particular such techniques may be potentially detrimental if the reconstructed data is used for multivariate analysis

since such analysis makes use of the correlation between different variables. The main objective of this paper is to investigate impact of data compression on multivariate data analysis, specifically Principal Components Analysis (PCA).

The major contributions of this paper can be listed as:

- Data compression is formulated as a missing data problem and compression mechanism has been characterized from a missing data perspective.
- Multivariate reconstruction of compressed data has been proposed. An iterative missing data reconstruction technique (PCAIA) is combined with the compression detection algorithm for building model from compressed data and restoring the correlation structure of the data matrices.
- A comparative study between the competing compression algorithms (e.g., Swinging Door compression and Wavelet compression) is conducted in the context of multivariate modelling. New insight has been given as to why Wavelet compression performs better in preserving the correlation structure of process data.
- A study is performed to show how compression affects models of the processes with different dynamic behavior starting from a highly stochastic to a slowly varying auto-regressive process.
- Finally, the impact of compression on multivariate model building is demonstrated using a simulated flow-network system and an industrial data set from petroleum refinery.

2 Overview of Data Compression Methods

In process industries the measurements from all on-line sensors are first transmitted to the DCS systems. Most DCS systems are repository of raw data for a short period. However, for long term storage data are first compressed and stored in the data historian. Data in its compressed form are stored as a sparse matrix of raw values or coefficients in the transformed space. Since most data analysis techniques can only deal with a complete data matrix and time domain data, it is necessary to reconstruct the compressed data to a complete data matrix in the original time domain.

Therefore each data compression algorithm also has an accompanying reconstruction algorithm. The combined compression and reconstruction is referred to as ‘compression algorithm’. There is a wide variety of compression algorithms described in the literature especially in the context of image compression. Compression algorithms can be divided into two main groups: 1) Direct method and 2) Transform method.

Direct methods are rule based methods which store data by looking at its deviation from the trend of the signal. Some of the popular direct methods are, piecewise linear compression³, Box-Car, Backward Slope, a combination of these two methods called Box-Car-Backward-Slope (BCBS) and the Swinging Door algorithm⁴. Direct methods make the archiving decision in real time as the data are recorded from the process. Therefore Direct methods have been the methods of choice for most industrial data archiving systems, for example, *AspenTech*[®] uses an adaptive method based on Box-Car-Backward-Slope (BCBS) in their data historian and *OSI*[®] uses a variant of Swinging Door algorithm in their PI historian^{6;12}.

Transform methods perform an integral transform of the original data set and then transform it to a set of coefficients in the new space. Compression is performed on these transformed coefficients. Example of some commonly used transforms are, Laplace transform, Fourier transform, and Wavelet transform. However, Wavelet transform is most suitable from a data compression perspective and most of the transformed compression algorithms are based on Wavelet transformation. Superior performance of Wavelet compression has been demonstrated in different context including, compression and subsequent reconstruction of process data from paper making machine⁷, on-line feature extraction and noise removal from non-stationary signals⁸ and pattern matching in historical data². All these applications are off-line in the sense that compression is applied on that data set after a batch of data has been collected. An online data compression strategy using Wavelets have also been developed by⁶. This algorithm works sequentially, i.e., with the arrival of each new point the algorithm computes all approximation coefficients and updates the multi-resolution tree. An efficient bookkeeping methodology has also been proposed, which improves compression ratios significantly over the batch or off-line version of Wavelet compression.

In this study, the Swinging Door compression and the Wavelet compression algorithms are taken as two representative algorithms from the direct and transform methods, respectively.

Swinging Door compression algorithm is based on the idea that within a signal trend it may be possible to identify many linear segments. Therefore storing only the end points of these linear segments may be sufficient to capture the main dynamics of the system. Swinging Door compression acts sequentially on each data point and therefore it can be applied in an on-line fashion to compress the data. The working principle of Swinging Door compression algorithm can be found in¹ and⁹. The Swinging Door algorithm uses a linear interpolation method to reconstruct the signal. Therefore the reconstructed signals will have many linear segments in between the raw data points. Linear interpolation will create intermittent points at regular time interval as specified by the user. The reconstruction criteria is to minimize the deviation of the reconstructed signal from the actual signal and not aimed towards preserving the variance of the signal or the correlation between different variables in the reconstructed signals. Since multivariate analysis makes use of the correlation between the variables, linear interpolation type reconstruction is clearly unsatisfactory for such analysis.

Wavelet compression and reconstruction is based on Wavelet Transform and Inverse Wavelet Transform respectively. The main objective of Wavelet Transform is to locate a frequency component as well as the exact time of occurrence. In this sense it is very similar to Short Time Fourier Transform (STFT). However, a Wavelet transform does it more efficiently by dividing data, functions, or operators into different frequency components and then processing each component with a resolution matched to its scale. For example, a high time resolution (narrow window) is used for high frequency signals and low time resolution (wide window) is used for low frequency signals¹⁰. A tutorial on Wavelet Transform can be found in¹¹. During data compression only the high frequency information is lost. This is commensurate with the needs of the process and control engineers since most of the high frequency signals come from disturbances, are short lived and not of interest. On the other hand, process dynamics are mostly in low frequency region and persist throughout the duration. Wavelet compression reconstruction is implemented in three main steps: (i) Wavelet Transform (ii) Thresholding and (iii) Inverse Wavelet Transform¹². The transformation of a signal during these steps is shown in Figure 1.

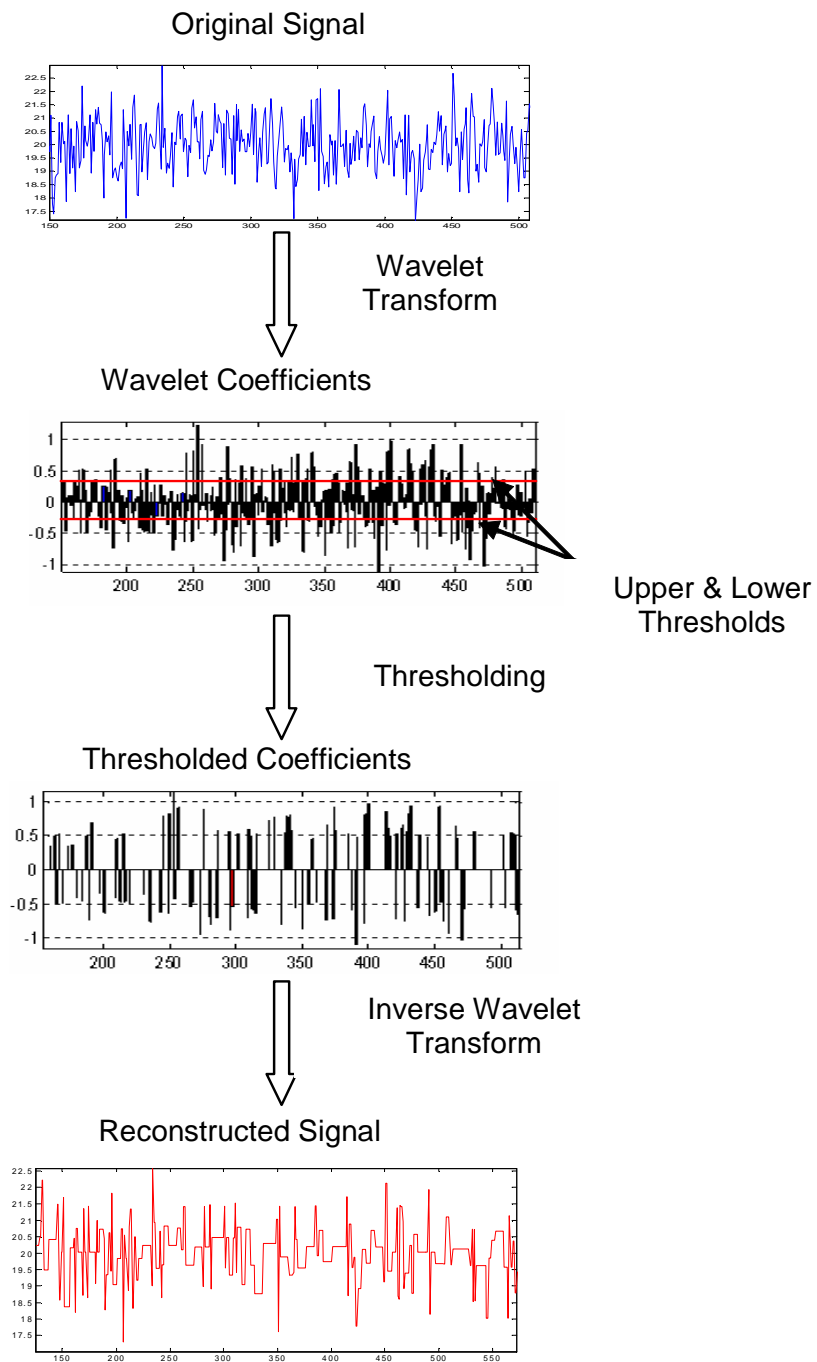


Figure 1: *Schematic representation of Wavelet Compression and Reconstruction Algorithm*

3 Formulation of Compression as a Missing Data Problem

Data historians used in process industries almost exclusively use direct methods for compressing data. In this section we will formulate data compression as a missing data problem. Process historians use decompression algorithms to provide a data matrix with the specified sampling rate. These decompression methods mostly use linear interpolations to fill the points in-between the originally stored spot values. The reconstructed signals from Swinging Door compression algorithm are shown in Figure 2. The reconstructed signals show many linear segments.

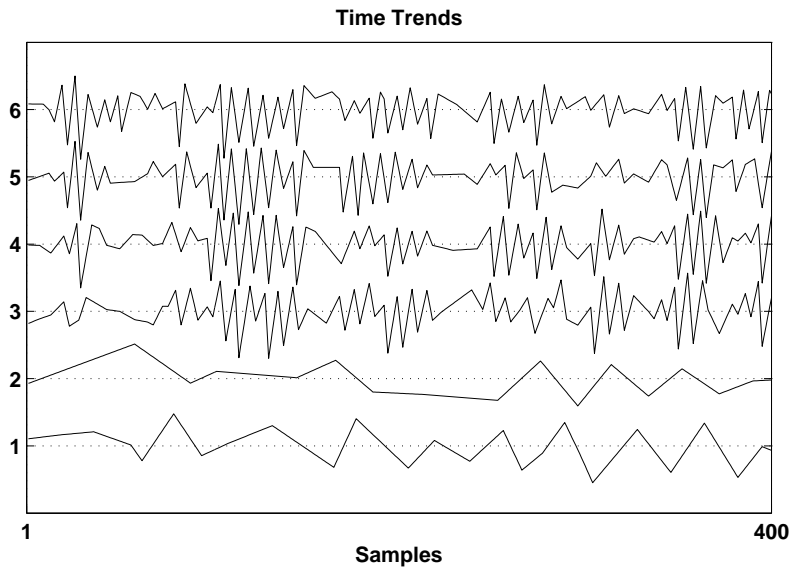


Figure 2: *Data from several loops of a refinery process archived using a Swinging Door compression algorithm to a factor of 10 and subsequently reconstructed using the built-in reconstruction algorithm.*

In order to cast the problem in a missing data formulation, the first step is to take out the interpolated points. Only the spot points are retained and subsequently used for building the model using multivariate missing data handling techniques. This is illustrated in Figure 3. The measurements from the level loops of a distillation column were compressed by a factor of three using Swinging Door compression algorithm. The signals were reconstructed using linear interpolation. Data matrix of the corresponding signals are shown in Figure 3(a), where the linearly interpolated points are replaced with ‘NaN’. This shows the distribution of the originally stored spot values.

(a)	18.915	10.199	29.486	30.427	19.639	19.639
20.241	9.0755	NaN	NaN	20.288	20.288	
NaN	NaN	NaN	NaN	NaN	NaN	
18.505	7.7958	26.58	26	18.539	18.539	
NaN	NaN	NaN	NaN	NaN	NaN	
20.931	11.231	32.214	31.923	20.941	20.941	
NaN	NaN	NaN	NaN	NaN	NaN	
22.26	10.549	32.853	32.312	21.679	21.679	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	13.139	32.418	32.736	NaN	NaN	
18.484	NaN	NaN	NaN	18.877	18.877	
NaN	NaN	NaN	NaN	NaN	NaN	
21.029	10.2	31.665	NaN	20.87	20.87	
NaN	NaN	NaN	30.56	NaN	NaN	
19.052	12.804	NaN	NaN	NaN	NaN	
NaN	NaN	32.119	32.276	19.494	19.494	
NaN	9.8168	NaN	NaN	NaN	NaN	
20.95	NaN	31.356	31.186	21.305	21.305	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	8.4052	28.349	27.593	NaN	NaN	
18.455	NaN	NaN	NaN	18.735	18.735	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	10.087	NaN	30.08	NaN	NaN	
21.213	NaN	33.521	NaN	20.886	20.886	
NaN	13.015	NaN	33.423	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	8.5436	28.526	28.927	NaN	NaN	
20.217	NaN	NaN	NaN	19.898	19.898	
NaN	9.5583	NaN	30.567	NaN	NaN	
NaN	NaN	NaN	NaN	21.527	21.527	
NaN	10.053	31.034	30.56	NaN	NaN	
20.579	NaN	NaN	NaN	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	9.539	29.484	29.418	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	13.166	33.806	33.689	20.463	20.463	
19.614	NaN	NaN	NaN	NaN	NaN	
NaN	8.2582	27.895	27.337	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	NaN	NaN	30.397	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
19.559	11.375	30.664	31.456	19.307	19.307	

(b)	18.915	10.199	29.486	30.427	19.639	19.639
20.241	9.0755	NaN	NaN	20.288	20.288	
18.505	7.7958	26.58	26	18.539	18.539	
NaN	NaN	NaN	NaN	NaN	NaN	
20.931	11.231	32.214	31.923	20.941	20.941	
22.26	10.549	32.853	32.312	21.679	21.679	
NaN	NaN	NaN	NaN	NaN	NaN	
18.484	NaN	NaN	NaN	18.877	18.877	
21.029	10.2	31.665	NaN	20.87	20.87	
NaN	NaN	NaN	30.56	NaN	NaN	
19.052	12.804	NaN	NaN	NaN	NaN	
NaN	NaN	32.119	32.276	19.494	19.494	
NaN	9.8168	NaN	NaN	NaN	NaN	
20.95	NaN	31.356	31.186	21.305	21.305	
NaN	8.4052	28.349	27.593	NaN	NaN	
18.455	NaN	NaN	NaN	18.735	18.735	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	10.087	NaN	30.08	NaN	NaN	
21.213	NaN	33.521	NaN	20.886	20.886	
NaN	13.015	NaN	33.423	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	8.5436	28.526	28.927	NaN	NaN	
20.217	NaN	NaN	NaN	19.898	19.898	
NaN	9.5583	NaN	30.567	NaN	NaN	
NaN	NaN	NaN	NaN	21.527	21.527	
NaN	10.053	31.034	30.56	NaN	NaN	
20.579	NaN	NaN	NaN	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	9.539	29.484	29.418	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	13.166	33.806	33.689	20.463	20.463	
19.614	NaN	NaN	NaN	NaN	NaN	
NaN	8.2582	27.895	27.337	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
NaN	NaN	NaN	30.397	NaN	NaN	
NaN	NaN	NaN	NaN	NaN	NaN	
19.559	11.375	30.664	31.456	19.307	19.307	

Figure 3: (a) Data matrix after the linearly interpolated points have been replaced by 'NaN' (b) Data matrix after removing the rows from data matrix (a) which do not contain a single spot value

Some of the rows do not contain a single spot value in the row. These rows have been shaded in the data matrix. Since these rows do not contain any information they were removed and the new data matrix is shown in Figure 3(b). This is the missing data formulation of the compressed data, where the missing values exist all over the data matrix. Multivariate missing data handling techniques may be used to predict missing values in such a data matrix.

3.1 Characterization of Compression Mechanism

In order to reconstruct the missing values, it is important to characterize the mechanism that generated the missing values. In the missing data literature, mechanisms are classified in three categories: i) Missing Completely At Random (MCAR), ii) Missing At Random (MAR) and iii) Non

Ignorable (NI) mechanism. Details of the definitions can be found in^{13;14}. These classifications provide a guideline for reconstruction and possible implications of any assumption. Here we give a brief description of these mechanisms to classify compression in light of these definitions. As shown in Figure 3(b), the original spot values and missing values are spread all over the data matrix. Any data set, $Y = (y_{ij})$, containing an observed part and a missing part is represented as $Y = (Y_{obs}, Y_{mis})$. This notation will also be used in this paper. A matrix $M = (m_{ij})$, referred as the missingness matrix, is used for indexing the missing and the observed part. Each element of M is a single binary item indicating whether y_{ij} is observed ($m_{ij} = 1$) or missing ($m_{ij} = 0$). In the statistics literature, missingness is treated as a random phenomena. The distribution of M , called missingness mechanism, is characterized by $f(M|Y, \phi)$, the conditional distribution of M given Y , where ϕ denotes parameters unrelated with Y . Classification of missingness mechanism is based on the conditionalities:

1. Missing Completely At Random(MCAR)

In this case missingness does not depend on any part of the data Y either missing or observed.

$$f(M|Y, \phi) = f(M|\phi)$$

2. Missing At Random(MAR)

Missingness depends only on the observed component Y_{obs} and not on the missing component Y_{mis} of the data matrix.

$$f(M|Y, \phi) = f(M|Y_{obs}, \phi)$$

3. Non Ignorable Mechanism (NI)

If the mechanism of missingness is dependent on both the observed and the missing part of the data then the mechanism is Non Ignorable.

Under MCAR and MAR conditions the mechanism that led to missing data can be ignored in the reconstruction process. Any model based on the observed data will give reasonable reconstruction. For Non-Ignorable cases the missing mechanism has to be taken into account in the reconstruction

of missing data. In many cases it is not possible to include the mechanism in the reconstruction process. To circumvent this, it is customary to assume data as MAR and build the model only based on the observed part of the data at the expense of some accuracy.

Now let us analyze where compression mechanism stands according to the above definitions. While storing a value using Swinging Door, the deviation of the value from the linear trend is calculated. If the point is outside the desired bound only then is it archived, otherwise it is discarded. So essentially the distribution of M will be dependent on both Y_{obs} and Y_{mis} . Therefore, from the view point of missing data, compression is a ‘Non-Ignorable(NI)’ mechanism. This indicates that a method that is inverse to the compression algorithm should be used for exact reconstruction of the signal. However, the mechanism used for compressing the data is an irreversible one and it is not possible to include it in any form in the multivariate reconstruction process. In the absence of any such mechanism, we will assume that the missing mechanism of all data is ‘Missing at Random (MAR)’ and use the model based on the observed data to reconstruct the missing part of the data matrix. Since there is a probability that the observed part of the data may be systematically different than the missing part, it may introduce some error in the model or the reconstructed signal. However, this is the best practice in this situation.

Due to compression the percentage of missing data is very high, for example, for a compression factor of 3 approximately 66% of the data is missing and at a compression factor of 10 only one out of ten points is recorded which means 90% of the data is missing. So, from a missing data view-point compression can be seen as Non-Ignorable mechanism with a very high percentage of missing data.

4 Reconstruction of Swinging Door Compressed Data using PCAIA

PCA based missing data handling techniques have been used to reconstruct small amount of missing data and perform PCA based process monitoring in the presence of missing values in the data matrix. The details of the methods can be found in^{5;15}. However, in the current study we show the

application of the methods in a completely new context. Missing data handling method is used for restoring the correlation structure and building multivariate model from compressed data. First, data compression is cast as a missing data problem and subsequently the Principal Component Analysis Iterative Algorithm (PCAIA)¹⁵ is used for building the model. Reconstructing signals from compressed data using missing data handling techniques is challenging because most of the techniques are not suitable for dealing with such high percentage of missing data. However, if used judiciously, missing data handling techniques can be useful in extracting the true correlation between the variables. The implementation steps of the algorithm are shown via a flow diagram in Figure 4. The method is suitable for working with compressed data from any direct method. The retrieved data matrix from the process historian contains some originally stored spot values and linearly interpolated points in between them. The first part of the reconstruction algorithm is to find the original stored points. To find these data points a compression detection algorithm was used¹. This algorithm can find the spot values from signals which were reconstructed using linear interpolation. Since the reconstructed signal is piecewise linear, it will have discontinuity only at the locations of the spot values. Therefore, the locations of the spot values are given by the locations of the non zero double derivatives. Second derivatives are calculated at each point of the signal using the difference relationship shown in Equation 1.

$$\begin{aligned}\Delta(\Delta\hat{y})_i &= \frac{(\hat{y}_{i+1} - \hat{y}_i)/h - (\hat{y}_i - \hat{y}_{i-1})/h}{h} \\ &= \frac{\hat{y}_{i+1} - 2\hat{y}_i + \hat{y}_{i-1}}{h^2}\end{aligned}\quad (1)$$

where \hat{y}_i is the reconstructed signal and h is the sampling interval. If N is the total length of the signal, index i ranges from 2 to $(N-1)$. Only the spot values are retained, and the rest of the points in the data matrix are considered as missing. This is illustrated in Figure 3(a) where the missing values have been indicated by ‘NaNs’. However, at this stage the ‘percentage of missing data’ in the data matrix would be high since in many situations we may encounter highly compressed data, e.g., for a compression factor of five, 80% of the data would be missing. This poses difficulty in reconstruction as most iterative missing data handling techniques do not converge for more than 20% missing data in the data matrix. Therefore a multistage procedure is applied to bring down the percentage of missing data in the data matrix. In the first step all rows which do not have

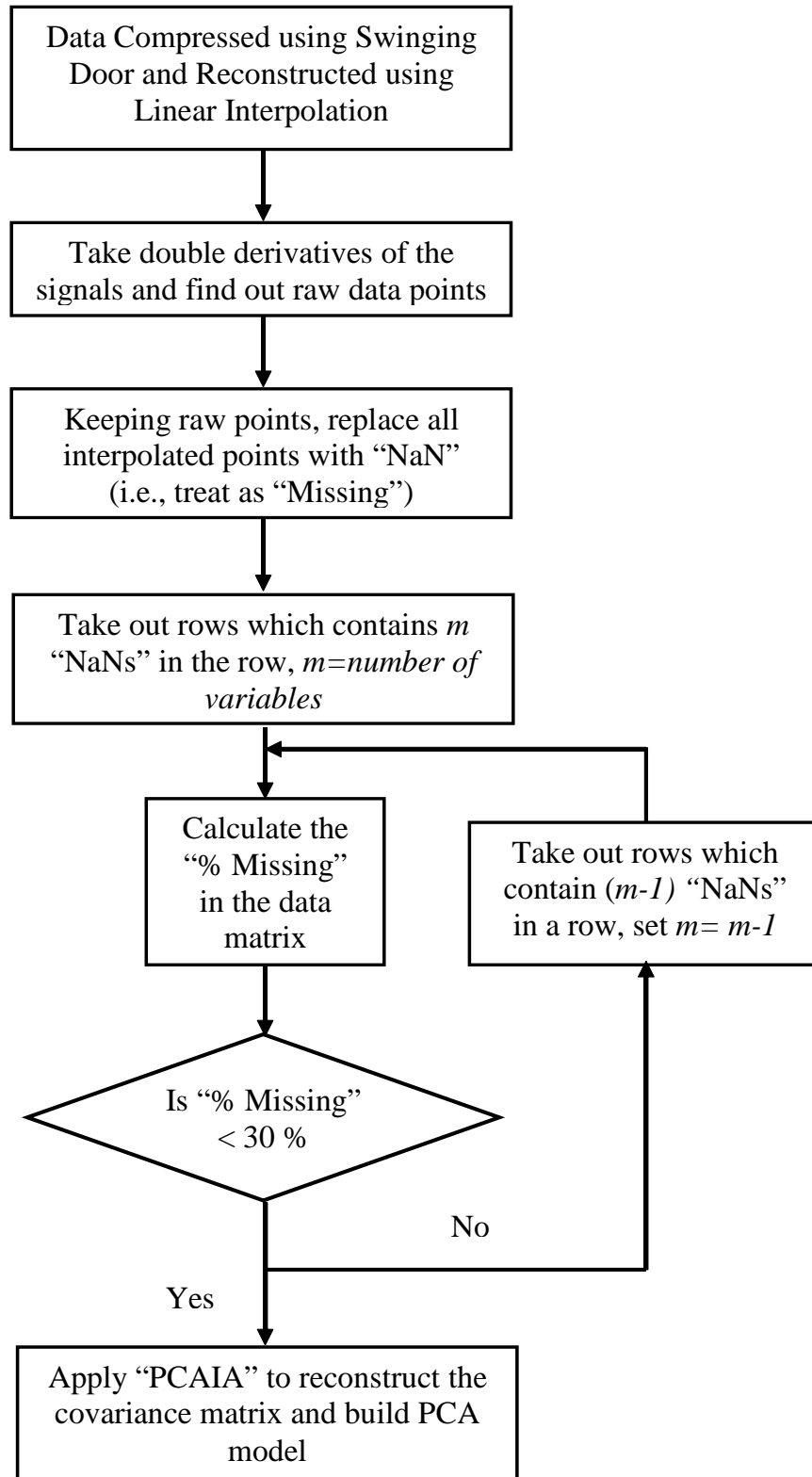


Figure 4: *Proposed algorithm for building PCA model from Swinging Door Compressed and Linearly Reconstructed Data*

any original points are taken out of the data matrix. These rows do not contain any information. This is illustrated in Figure 3(a) where all the rows which do not contain a single spot value are shaded. This data matrix was obtained from a data set which was compressed by a factor of three. Therefore 66% of the data are missing at this stage. After removing the rows which do not contain a single spot value, i.e., the shaded rows in Figure 3(a), the new data matrix takes the form shown in Figure 3(b). Clearly because of the removal of the intermediate rows from the data matrix, the time steps between the rows are no longer uniform. However, since we are interested in static models this does not affect the analysis. In order to build dynamic model (e.g., Dynamic PCA) from serially correlated data in the time direction, we need to first include the lagged variables in the data matrix and create the lagged data matrix. Then only the non-informative rows can be eliminated from the lagged data matrix. The ratio of spot values to missing values improved at this stage and the ‘percentage missing data’ in the new data matrix (Figure 3(b)) reduced to ‘50%’. In the next phase, rows which contain only one spot value are taken out of the data matrix. This will help to further reduce the percentage of missing values in the data matrix. The procedure is repeated until the percentage of missing values in the data matrix comes down to 30%, e.g., removing rows with two original values in the next step. The PCA based missing data handling technique gave good estimates of model and the iterative algorithm converged well up to 30% of missing values in the data matrix. After doing extensive simulation studies we arrived at this number. However, it is not possible to take out all the missing values and create a complete data matrix, because the original spot values of different variables are not aligned with each other. If only complete rows are retained it will drastically reduce the sample size. Furthermore in reality chemical processes are often time varying and the correlation of the data changes with time. We often capture an average correlation of the data of the entire time period. Therefore selecting rich data from a one time interval may have very different correlation structure than the average correlation of the entire data set. After the percentage of missing data is within 30%, Principal Component Analysis Iterative Algorithm (PCAIA) is used to restore the correlation structure and build PCA model from the data matrix. Reduction of ‘% missing data’ in the data matrix is particularly important for convergence of the algorithm. However, in many cases where the missing values are distributed randomly instead of blocks of missing values the algorithm may be able to handle a larger percentage of missing values. The implementation of PCAIA is carried out as follows:

1. The missing values of the data matrix are filled with the unconditional mean of the variables. For example, the missing values of the data matrix are filled by the column averages of \mathbf{Y}_{obs} which gives the augmented data matrix $\mathbf{Y}_{aug} = [\mathbf{Y}_{obs}, \mathbf{Y}_{mis}]$ where $\mathbf{Y}_{mis} = mean(\mathbf{Y}_{obs})$ and $\mathbf{Y}_{aug} \in \mathbb{R}^{N \times n}$.
2. Singular Value Decomposition (SVD) is performed on the augmented data matrix. The loading matrix \mathbf{P} is used to predict the noise free values $\hat{\mathbf{X}} = \mathbf{YPP}^T$.
3. Missing values are filled with predicted values, $\hat{\mathbf{X}}$ and the augmented data matrix will be $\mathbf{Y}_{aug} = [\mathbf{Y}_{obs}, \hat{\mathbf{X}}_{mis}]$, where $\hat{\mathbf{X}}_{mis}$ are predicted values in the previous step.
4. Convergence is monitored by observing the sum of squared errors between the observed values and corresponding predicted values from step (2).

$$SSE_{obs} = \sum_{i=1}^N \sum_{j=1}^n \left(Y_{ij} - \hat{X}_{ij} \right)_{obs}^2$$

Step (2) and step(3) are repeated until convergence.

In the current study we assumed that the model order or the dimensions of the loading matrix \mathbf{P} are known. However, in many real applications the model order may not be known exactly. Because of missing data, the percentage variance explained by the PCs becomes a function of missing data and model order selection gets complicated. In the presence of missing values a cross-validation based detailed method is incorporated into the algorithm to find out the model order¹⁶.

Remark PCAIA is a pseudo version of the more general Expectation Maximization (EM) algorithm¹⁷. Here it may be interesting to explore the link with EM. Similar to EM we can identify the two major iterative steps of the algorithm.

Parameter Estimation step is similar to the Maximization (M-Step) of the EM algorithm. From the augmented data matrix, where missing values are filled with conditional expected values, the loadings of the PCs are calculated. These are the parameters in this case. However, the method is optimal in the least squares sense contrary to the Maximum Likelihood Estimates obtained in EM.

Missing Value Estimation resembles the Expectation step (E-step) of the EM algorithm. Using the estimated parameters, missing values are estimated in this step. These values are used to fill

the missing values and get a better augmented data matrix. In the Expectation step of the EM algorithm missing values are not directly estimated, rather the expectation of the sufficient statistics of the log-likelihood function are calculated. Therefore, the two methods will be only equivalent when the log-likelihood is linear in data or in other words the sufficient statistics of the log-likelihood equation are function of the data values only.

5 Results and Discussions

The results of the analysis are demonstrated using two examples, a simulated flow-network system and an industrial case study. The industrial data is taken from a petroleum refining process. The description of the Flow-network system and the refinery data are given below:

5.1 Simulation Example

The flow-network process, shown in Figure 5, will be used to compare the relative advantages and disadvantages of different methods. This is a benchmark example used by¹⁸ and a similar example was used by¹⁹ to evaluate different properties of Bayesian PCA. It is assumed that the fluid flowing through the network is incompressible and there is no time delay in the process. The constraint model \mathbf{A} , of the process can be obtained easily from the mass balance equation at the junctions. The following four mass balance equations can be written for this flow-network system.

$$x_1 + x_2 - x_3 = 0$$

$$x_3 - x_4 = 0$$

$$x_4 - x_5 - x_2 = 0$$

$$x_5 - x_6 = 0$$

Thus the constraint model is:

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & -1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

The rank of the constraint matrix is four, which is also known as the order of the constraint model. In the above example x_1 and x_2 were chosen as independent variables. These are two deterministic signals and output of auto regressive (AR) processes given by,

$$\begin{aligned} x_1(k) &= ax_1(k-1) + \nu_k \\ x_2(k) &= bx_2(k-1) + \omega_k \end{aligned} \quad (2)$$

where the values of a and b are between 0 and 1, ν_k and ω_k are zero mean Gaussian noises. The rest of the flow rates, x_3 to x_6 were calculated from the mass balance equations. These variables are noise free and satisfy the model,

$$\mathbf{A}\mathbf{X}^T = 0$$

where $\mathbf{X} = [X_1 \ X_2 \ X_3 \ X_4 \ X_5 \ X_6]$ and X_1 to X_6 are vectors containing the actual flow values at each sampling point. However, in process industries the actual values of the variables are generally not available, only the noise corrupted variables \mathbf{Y} are available,

$$\mathbf{Y} = \mathbf{X} + \varepsilon$$

where ε is a matrix containing the measurement noise. Measurement noises of the variables at any sample i are uncorrelated with unequal variances (*i.e.* $\varepsilon_i \sim N(0, \sigma_j^2 I)$, $j = 1, 2, \dots, 6$).

Data generated from this simulated system were compressed using both Swinging Door and Wavelet compression algorithms and subsequently decompressed using the commonly used built-in reconstruction methods, and also the proposed PCAIA. To investigate the effect of compression on model quality, PCA models were built from the decompressed data sets and the estimated models were compared with the true model. The total data length for current study is 2000 samples.

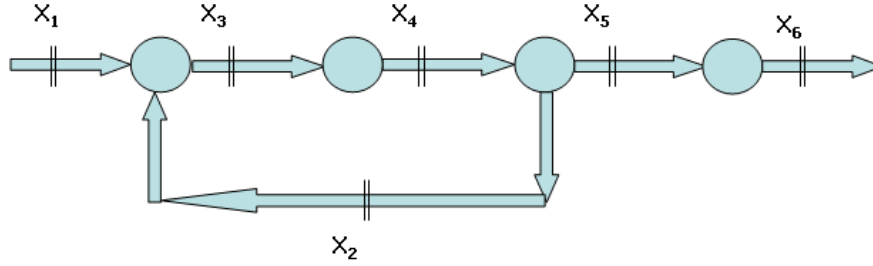


Figure 5: *Schematic Diagram of the Flow-Network*

5.2 Industrial Case Study

The industrial data used in this analysis were obtained from a petroleum refining process. All six variables are level measurements at different locations of a distillation column. The sampling time interval for the data is 60 sec and the total length of the data set is 20000 samples. The data was obtained in uncompressed form. For investigative purpose it was compressed to different compression levels. Due to the proprietary nature of the refining process, no process information is provided here.

5.3 Performance measure for model quality

Principal Component Analysis (PCA) is obtained by the Singular Value Decomposition (SVD) of the covariance matrix where the loadings of the PCs are given by the eigenvectors. In a multidimensional problem the eigenvectors can be multiplied using any non-singular matrix to define the same hyperplane. The exact value of each of the element depends on how the basis vectors are selected. So a direct comparison of the parameter values with actual model parameters is not feasible. Instead one should examine if the hyperplane defined by the estimated model is in agreement with the actual model hyperplane. In this study the subspace angle, θ is used to measure such agreement.

Let F and G be given subspaces of real space \mathfrak{R}^m , $u \in F$, $v \in G$, and assume for convenience that $p = \dim(F) \geq \dim(G) = q \geq 1$. The smallest angle $\theta_1(F, G) = \theta_1 \in [0, \pi/2]$ between F and G is

defined by

$$\cos(\theta_1) = \max_{u \in F} \max_{v \in G} u^T v$$

Assume that the maximum is attained for $u = u_1$ and $v = v_1$. Continuing in this way until one of the subspace is empty, we are led to the following definition.

The principal angles $\theta_1 \in [0, \pi/2]$ between F and G are recursively defined for $k = 1, 2, \dots, q$ by

$$\cos(\theta_k) = \max_{u \in F} \max_{v \in G} u^T v = u_k^T v_k, \|u\|_2 = 1, \|v\|_2 = 1$$

subject to the constraints

$$u_j^T u = 0, v_j^T v = 0$$

where σ_k is an eigenvalue of $F^T G$. Therefore subspace angle or principal angle is the minimum angle between the subspaces²¹.

On the other hand, similarity index is a combined index defined by,

$$\theta_0^2 = \frac{1}{q} \sum_{i=1}^q \cos^2(\theta_i) = \frac{1}{q} \sum_{i=1}^q \lambda_i$$

Where λ_i is the eigenvalue of $F^T G G^T F$. The value of the similarity index is between 0 and 1, where 1 means that the two subspaces are linearly dependent²⁰. Clearly these two indicators have the same origin. Only difference is in one case the minimum angle between the subspaces is reported, in other case the overall distance is reported. However, these two quantity give comparable results. In the current study we used subspace angle to quantify the model quality. The in built function ‘subspace.m’ from Matlab’s ‘Data analysis and Fourier transforms’ toolbox was used to calculate the subspace angle. The details of the algorithm can be found in²².

If the angle is small, the two matrices will be nearly linearly dependent which means the estimated model is closer to the actual model. In reality the exact value of A is seldom known so subspace angle cannot be used for monitoring convergence. Convergence of PCAIA was monitored using

the calculated sum squared errors of the observed values and corresponding predicted values. In addition to that for the simulation cases subspace angles were used to reaffirm the claims made about the performance of the algorithm.

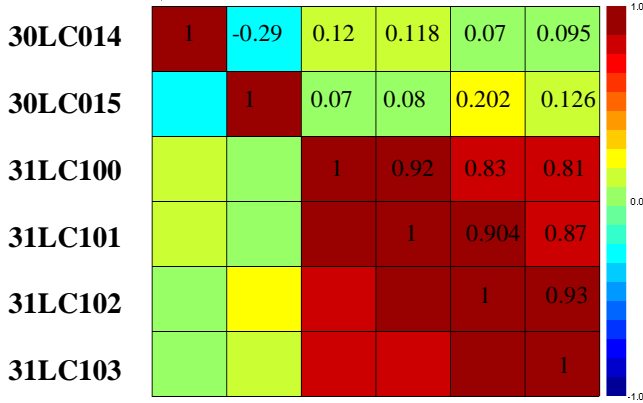
5.4 Effect of Compression on Correlation Structure

Almost all multivariate statistical data analysis methods, for example, pattern matching of historical data, fault detection and isolation using PCA, make use of the correlation between the variables. It is important to understand how compression affects the correlation structure of the data. A variety of industrial data has been used to visualize the effect of compression on correlation structure. The petroleum refining example described earlier will be stated here. The correlation matrix of the raw uncompressed data set is mapped in the color coded plot in Figure 6(a). The colors in the color-map indicate the magnitude of the correlation. This data set was compressed using Swinging Door and Wavelet compression algorithms to a compression factor of 10 and subsequently reconstructed using linear interpolation and Inverse Wavelet Transform respectively. The correlation color-map of the reconstructed data are shown in Figures 6(b) and (c) respectively. It is evident from the correlation color map that in the process of compression via the Swinging Door algorithm and linear reconstruction, the correlation between the variables has been severely distorted at this level of compression and the structure is significantly different from the true correlation structure shown in Figure 6(a). On the other hand, reconstructed data from Wavelet compression retains the true correlation structure in most parts. Although Wavelet compression is able to retain most of the significant correlation structure of the data, none of the current commercially available data historians use Wavelet Compression Algorithms. Swinging Door Compression or similar direct methods are used almost exclusively by commercial process historians. Therefore, in order to use the Swinging Door compressed data, especially for multivariate analysis, alternative methods should be used to reconstruct the compressed data so that it retains the true correlation structure between the variables. Instead of linear interpolation based methods, it is recommended that PCAIA be used to reconstruct the Swinging Door compressed data set. The correlation structure of the reconstructed data using PCAIA is shown in Figure 6(d). A comparison of Figures 6(a) and 6(d) shows that the PCAIA based reconstruction significantly restores the true

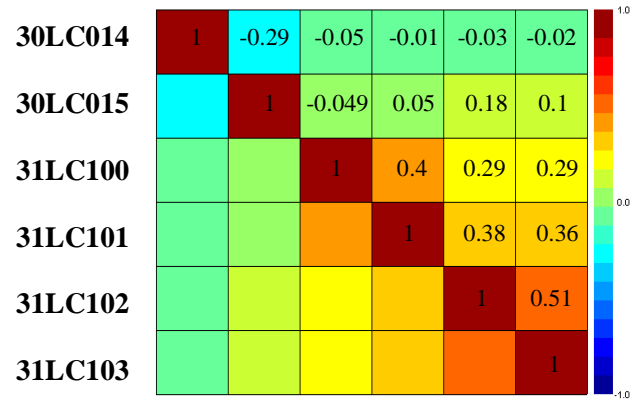
correlation between the variables.

5.5 Compression and Process Dynamics

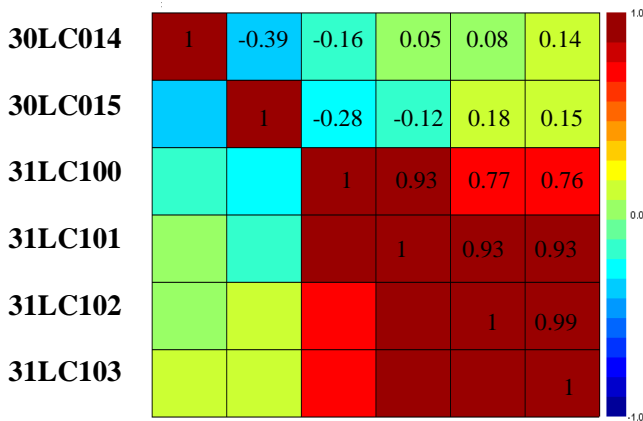
In order to get a quantitative measure of the interaction between compression and process dynamics, a parametric study was conducted using the Flow-network system. The independent flow-rates were generated using Equation 2. The input to the transfer function is a Gaussian random signal. Different dynamic behavior of the process were simulated by varying coefficients a and b from 0 to 0.9. As the coefficients vary from 0 to 0.9, the process gradually moves from a completely stochastic system to a slowly moving autoregressive process. All the signals were compressed by a factor of 3 using both Swinging Door and Wavelet compression algorithms. The signals were then decompressed using linear interpolation and Inverse Wavelet Transform respectively, and subsequently models were built from these decompressed data sets. The reported values are average of twenty simulations. In each simulation the independent signal was varied by using a different random part to the Auto Regressive process as well as the random measurement noise added to the signals were different (i.e., seed of the random number generator was changed). The error bars show the standard deviation due to such changes. Figure 7 shows the deviation of the estimated models (i.e. subspace angle) from true model with the change of the process dynamics. It is evident from the results that, the effect of compression is more severe on the multivariate model when the process exhibits faster dynamic behavior. However, as the individual signals become more predictable the effect in multivariate model building also gets minimal. The estimated model from ‘Wavelet Compressed and Inverse Wavelet reconstructed’ data has a smaller subspace angle than ‘Swinging Door compressed and Linearly Interpolated’ data in this region. However, as the coefficients of the AR models increase beyond 0.3, the subspace angles for the estimated models from both methods become equal. So the quality of the models are similar for processes with slow dynamics.



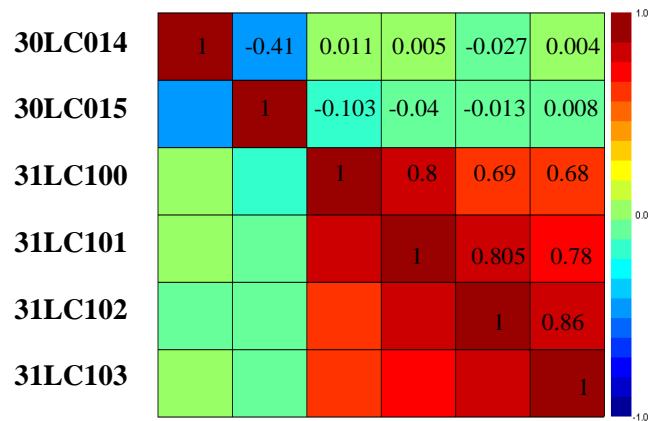
(a) Correlation Color-map of Uncompressed Data



(b) Correlation Color-map of Swinging Door Compressed and Linear Interpolation based Reconstructed Data



(c) Correlation Color-map of Swinging Door Compressed and PCAIA based Reconstructed Data



(d) Correlation Color-map of Wavelet Compressed and Inverse Wavelet based Reconstructed Data

Figure 6: Correlation color map of variables from a petroleum refining process. The intensity of the color shows the level of correlation between the variables. (It is recommended that this figure be viewed in color)

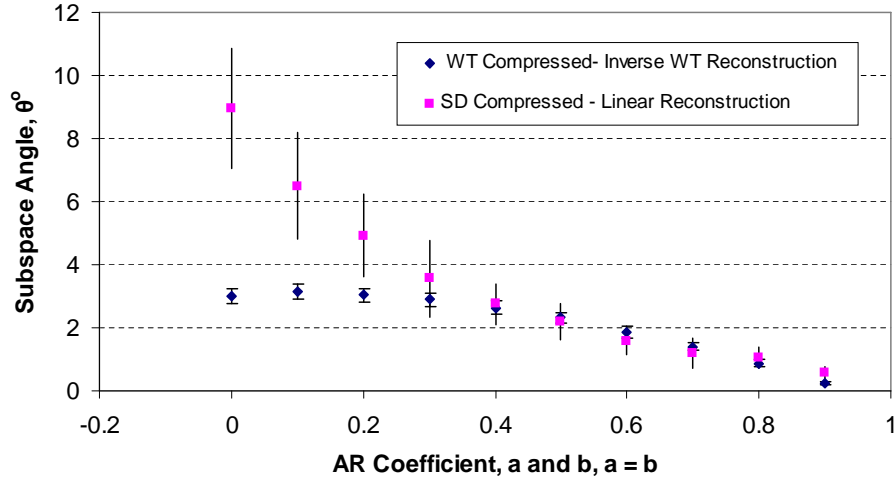


Figure 7: Variation of subspace angle with the change of the dynamic behavior of the flow-network system. a and b are coefficients of Equation 2. Both a and b were changed and $a=b$ in each case.

5.6 Improving model quality using Missing Data Handling Technique

In the previous section it was observed in the Flow-network system that, compression severely affects the model quality if the auto regressive coefficients are below 0.3. In this section we compare the performance of the proposed PCAIA method with the linear interpolation method in building a PCA model from Swinging Door compressed data. We also plot the subspace angle of the models obtained from data which were compressed using Wavelet Transform and reconstructed using Inverse Wavelet Transform.

5.6.1 Flow-network Example

The flow-rates x_1 and x_2 of the flow-network system are the output of the auto regressive process given in Equation 2 with coefficients $a = b = 0.3$. The methodology of building a PCA model from compressed data using missing data handling technique has been outlined in Section 3.2. Results of the analysis are presented in Figure 8. PCAIA was initialized with the column averages. Other initial values (e.g., linearly interpolated values or the values from Swinging Door Reconstruction) were also tried. However it did not provide any additional advantage. It is evident from the

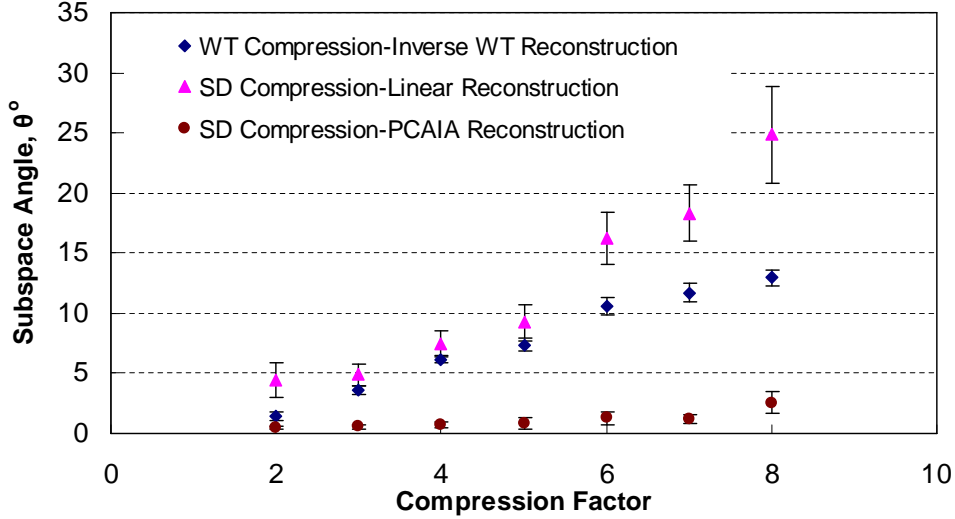


Figure 8: Variation of subspace angle with compression ratio. Compressed data from flow-network system was reconstructed using the three reconstruction methods, and subsequently used for building model.

figure that estimated model from ‘Wavelet Compressed and Inverse Wavelet Reconstructed’ and ‘Swinging Door Compressed and Linearly Reconstructed’ data have poor quality as the data is compressed beyond a compression factor of 3. On the other hand, PCAIA based modeling provides minimum subspace angle, i.e., the best model quality among the three methods. Models estimated using PCAIA has better quality up to compression ratio as high as 8. It clearly demonstrates that instead of using linear interpolation to reconstruct Swinging Door compressed data, use of PCAIA can be significantly beneficial in terms of a model that preserves the multivariate relationships between the variables. The main reason for the improvement is that, in PCAIA the missing values due to compression were reconstructed in a multivariate framework. As a result, the method accounted for the changes that took place to other variables as well. On the other hand, in linear interpolation a signal is reconstructed in a univariate framework, i.e., using only that particular variable, thus the reconstruction is not reliable if changes occur in other correlated variables at those instants. In those cases linear interpolation will miss the excitations and capture only the average behavior of the signal.

Since compression leads to high percentage of missing data the convergence of the iterative algorithm is an important concern. For the Flow-network example, the true model was available,

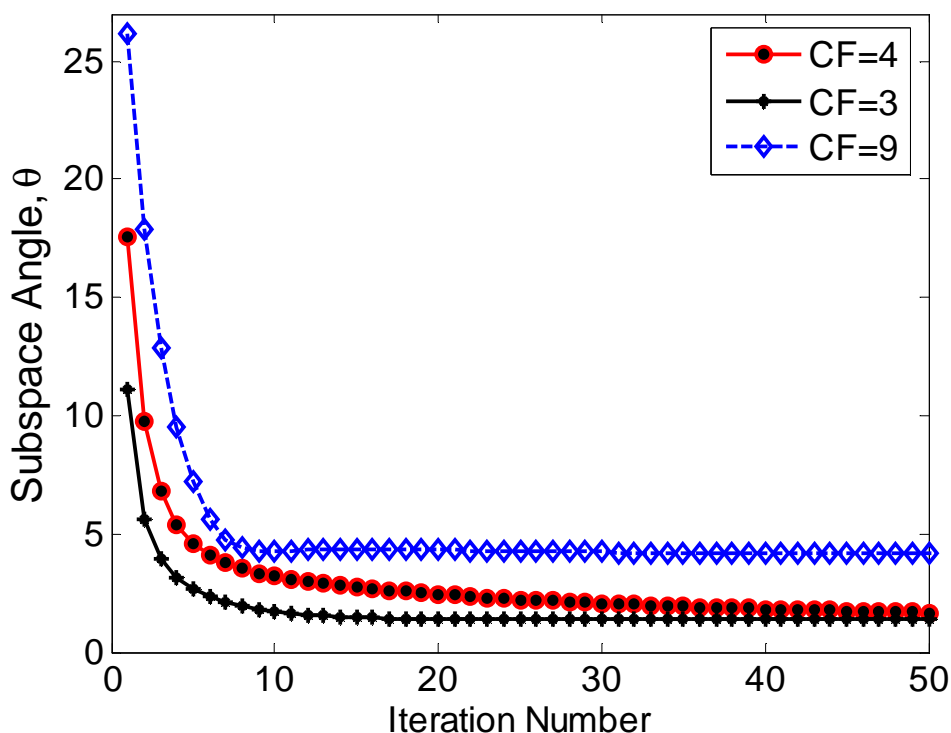


Figure 9: *Convergence of PCAIA at different compression ratio.*

so the change of model quality (i.e. subspace angle) at each iterative step was tracked. The subspace angle, as a measure of model accuracy, has been plotted against iteration number in Figure 9. The plot shows a monotonic convergence of the subspace angle at each successive step. The algorithm converges in less than 10 iterations even for highly compressed data. However, when the percentage of missing data is more than 40% (not shown in the figure) some divergent behavior was also observed. In those cases more stringent criteria has to be set and only rows with very few missing values should be retained, so that the percentage of missing data is within the manageable range. Similar to any data driven modelling the current methodology also assumes that the process is sufficiently excited.

5.6.2 Industrial Case Study: Refinery Data

The refinery data set used for correlation structure analysis is also used to investigate the performance of different compression algorithms and PCAIA, in a multivariate modelling context.

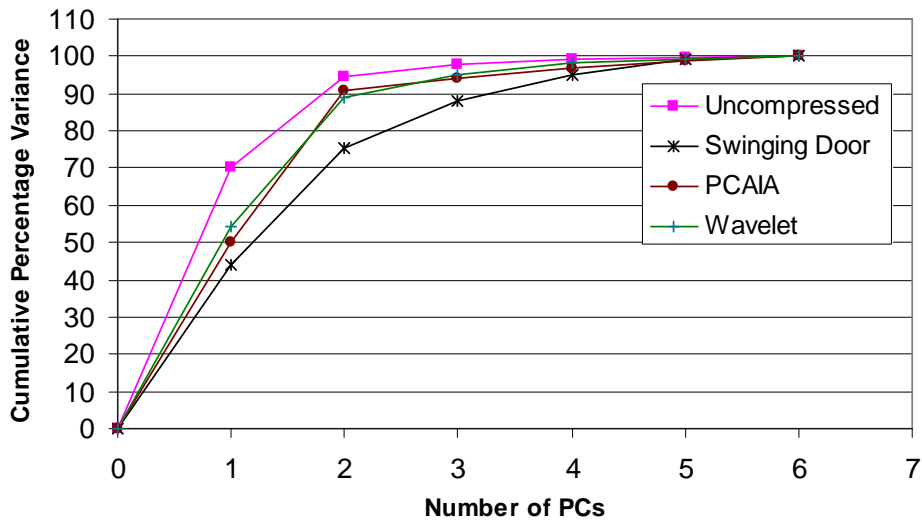


Figure 10: *Cumulative percentage of total variance explained by principal components from reconstructed data using different methods.*

Since this is an industrial data set the actual model of the process is unknown. In order to get a performance metric of model accuracy, first a benchmark model is built from the uncompressed raw data set. Subsequently models estimated from the reconstructed data sets are compared with this benchmark model. The percentage of total variance explained by the PCs calculated from the original uncompressed data with that of from various reconstructed data are plotted in Figure 10. The eigenvalue distribution of the Swinging Door Compressed and Linearly reconstructed data set is quite different from the uncompressed data set. For example, for the uncompressed data 90% of the total variance is explained by the first two PCs whereas it would require took four PCs to capture 90% variance for the linearly interpolated data. This poses a serious problem in selecting the order of a PCA model as most of the model order selection criteria are based on the analysis of variance. The calculated eigenvalues from Swinging Door Compressed and PCAIA reconstructed data set, and Wavelet Compressed and Inversed Wavelet Reconstructed data set are closer to the eigenvalues calculated from the original data set and the percentage variance explained by the major PCs are also very similar to the uncompressed data set. Thus the model order selection will be more precise for these two cases.

Figure 11 compares the quality of the models obtained using reconstructed data from three

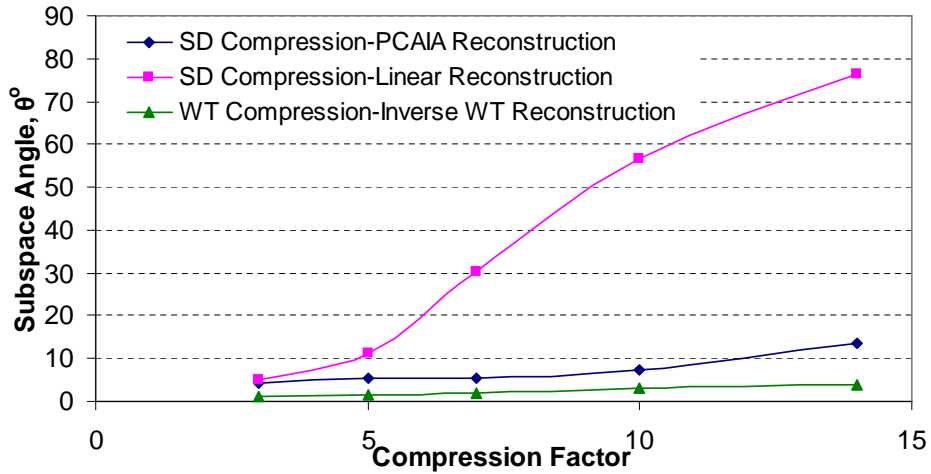


Figure 11: Comparison of estimated model quality from reconstructed data using different methods.

different reconstruction techniques. It may be noted here that ‘Linear Interpolation’ and ‘PCAIA’ reconstructed the compressed data from Swinging Door Compression algorithm while ‘Inverse Wavelet Transform’ reconstructed data which has been compressed using ‘Wavelet Transform’. Since the true model of the process is unknown, the model obtained from the uncompressed data was taken as the benchmark. Subspace Angles of all models obtained from the reconstructed data were calculated relative to this benchmark model. The model built from the ‘Swinging Door Compressed and Linearly Reconstructed’ data has very poor quality at moderate to high compression ratios, as linear interpolation destroys the correlation structure. By using PCAIA, instead of linear interpolation based reconstruction, significant improvement could be achieved in model quality. The estimated model from ‘Wavelet Compressed and Inverse Wavelet reconstructed’ data has the best quality. This is in contrast to the observation in the simulated flow-network system, where PCAIA had the best performance. Such result is not unexpected since the true dynamic nature of the process is not known, and the effect of compression depends on the dynamic behavior of the process. Moreover, the process may be non-stationary and nonlinear to some extent and after discarding the rows which do not contain any original values the sample size became quite small and PCAIA was applied only on that smaller sample size. As a result such small samples may not be completely representative of the process and the method may have suffered from small sample limitations.

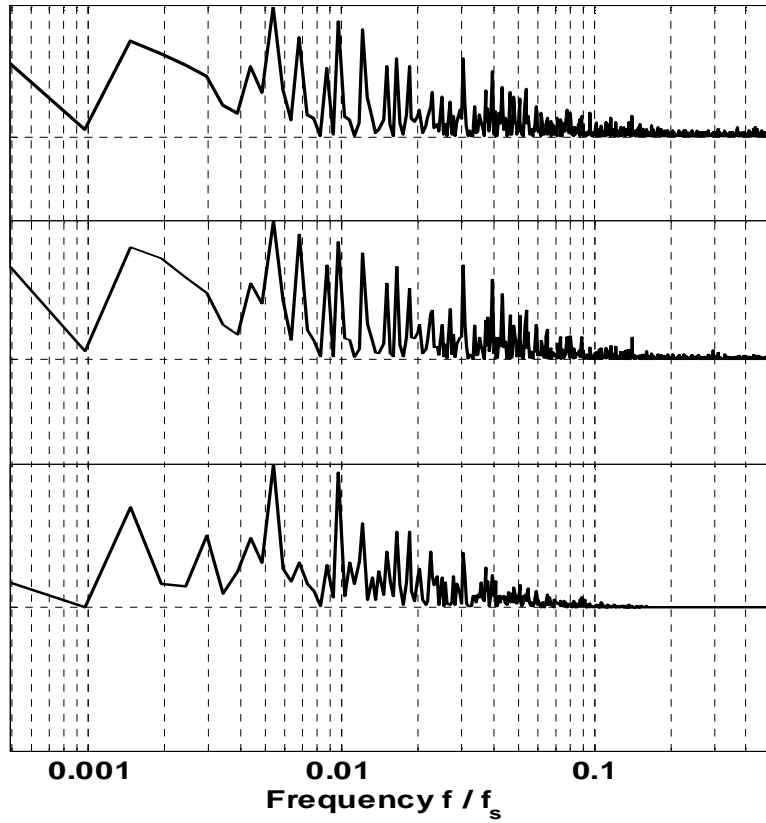


Figure 12: *Power Spectra Plot: Original Signal(Top) Reconstructed Signal from Wavelet Compression(Middle) Reconstructed Signal from Swinging Door Compression(Bottom).*

It was also observed in the analysis that Wavelet compression preserves the correlation structures between the variables better than Swinging Door algorithm. The primary reason for this behavior is, the correlation matrix captures the low and medium frequency information. The high frequency excitations in the signals are usually due to random noise and averages out while calculating the correlation matrix. Though Wavelet compression is univariate, during the compression and reconstruction it only chops the high frequency information. On the other hand, during the compression and reconstruction of Swinging Door algorithm part of the low and medium frequency information is lost. This is illustrated in Figure 12, where the spectral density plot of a signal and the reconstructed signals from Wavelet compression and Swinging Door compression are shown. It clearly shows that, in Wavelet reconstructed signal all of the low and medium frequency information remained intact while in the Swinging Door reconstructed signal part of the low and medium frequency information got lost. This will ultimately have an effect on the correlation structure between the variables.

We recommend the use of ‘Wavelet Compression and Inverse Wavelet based Reconstruction’ algorithms for process historian. However process industries almost exclusively use ‘Swinging Door type Compression and Linear Interpolation’ based algorithms in their data historian and this trend will continue to exist for some time. Instead of linear interpolation based reconstruction, missing data techniques based on PCAIA can be used to recover the correlation structure for building multivariate models. The ‘Inverse Wavelet based Reconstruction’ can only reconstruct data which have been compressed using ‘Wavelet Compression’. Therefore ‘Inverse Wavelet based Reconstruction’ is not an alternative to PCAIA in reconstructing Swinging Door compressed data.

6 Conclusions

A detailed study on the effect of compression on multivariate analysis, especially PCA-based modelling has been performed. Compression has been formulated and characterized as a missing data problem. A missing data handling technique (PCAIA) has been used successfully to build model from compressed data. The following conclusions can be drawn from this study:

- Linear interpolation methods to reconstruct compressed data from direct compression methods (i.e. Swinging Door) are not suitable for multivariate analysis. Estimated models from such data can be of poor quality and unreliable.
- A significant improvement in model quality can be achieved by using missing data handling technique to build multivariate models from compressed data.
- The impact of compression on model building increases with the increasing stochastic and dynamic nature of the processes.
- Transform compression methods (i.e., Wavelet Compression) are better in retaining the correlation structure of the signals, and as such decompressed data from transform compression algorithms are suitable for multivariate analysis. However, the performance may deteriorate if the signals have excitation only in the high frequency range.

References

- [1] N. F. Thornhill, M. A. A. S. Choudhury, and S. L. Shah. The impact of compression on data-driven process analysis. *Journal of Process Control*, 14:389–398, 2004.
- [2] A. Singhal and D. E. Seborg. Effect of data compression on pattern matching in historical data. *Industrial Engineering Chemistry Research*, pages 267–274, 2005.
- [3] J. H. Hale and H. L. Sellars. Historic data recording for process computers. *Chemical Engineering Progress*, 77:38–43, 1981.
- [4] E. H. Bristol. Swinging Door trending: adaptive trend recording. *ISA National Conference Proceedings*, pages 749–753, 1990.
- [5] Taylor P.A. Nelson, P.R.C. and J.F. MacGregor. Missing data methods in PCA and PLS: Score calculations with incomplete observations. *Chemometrics and Intelligent Laboratory Systems*, 35:45–65, 1996.
- [6] M. Misra, S. J. Qin, S. Kumar, and D. Seemann. On-line data compression and error analysis using wavelet technology. *AIChE Journal*, 46(1):119–132, January 2000.

- [7] Z. Nestic, G. Davis, and G. Dumont. Paper machining compression using wavelets. *Tappi Journal*, 80:191–203, 1997.
- [8] B. Bakshi and G. Stephanopoulos. Compression of chemical process data through functional approximation and feature extraction. *AIChE Journal*, 42:477–492, 1996.
- [9] M. A. A. Shoukat Choudhury. Detection and diagnosis of valve stiction. *Phd dissertation, University of Alberta*, 2004.
- [10] R. M. Rao and A. S. Bopardikar. Wavelet transforms : introduction to theory and applications . *Reading, MA : Addison-Wesley*, 1998.
- [11] R. Polikar. The wavelet tutorial. Technical report, 2005. <http://www.public.iastate.edu/~rpolikar/WAVELETS/WTutorial.html/>.
- [12] J. W. Matthew, A. Liakopoulos, B. Dragana, and C. Georgakis. A practical assessment of process data compression techniques. *Industrial Engineering Chemistry Research*, 37:267–274, 1998.
- [13] D. B. Rubin. Formalizing subjective notions about the effect of nonrespondents in sample surveys. *Journal of American Statistical Association*, 72:538–543, 1977.
- [14] R.J.A. Little and D.B. Rubin. *Statistical Analysis with Missing Data*, volume 2. John Willy and Sons, 2002.
- [15] B. Grung and R. Manne. Missing values in principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 42:125–139, 1998.
- [16] B. Walczak and D.L. Massart. Dealing with missing data. *Chemometrics and Intelligent Laboratory Systems*, 58:15–27, 2001.
- [17] Laird N. M. Dempster, A. P. and D. B. Rubin. Maximum likelihood from incomplete data via EM algorithm. *Journal of the Royal Statistical Society. Series B(Methodological)*, 39(1):1–38, 1977.
- [18] S. Narasimhan and S. L. Shah. Model identification and error covariance matrix estimation from noisy data using PCA. *Control Engineering Practice*, December, 2006.

- [19] M. N. Nounou, B. Bakshi, P. K. Goel and X. Shen. Bayesian principal component analysis. *Journal of Chemometrics*, 16:576–595, 2002.
- [20] W. J. Krzanowski. Between-Groups Comparison of Principal Components. *Journal of American Statistical Association*, 74(367):703–707, 1979.
- [21] A. Bjorck and G. Golub. Numerical methods for computing angles between linear subspaces. *Mathematical Computations*, 27:579–594, 1973.
- [22] A. W. Knyazev and M. E. Argentati. Principal Angles between Subspaces in an A-based Scalar Product: Algorithms and Perturbation Estimates data. *SIAM Journal of Scientific Computations*, 23(6):2008–2040, 2002.