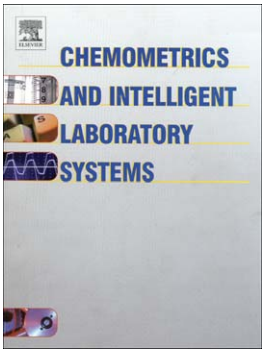
Accepted Manuscript



Defining the structure of DPCA models and its impact on Process Monitoring and Prediction activities

Tiago J. Rato, Marco S. Reis

PII: S0169-7439(13)00049-X
DOI: doi: 10.1016/j.chemolab.2013.03.009
Reference: CHEMOM 2620

To appear in: *Chemometrics and Intelligent Laboratory Systems*

Received date: 23 April 2012
Revised date: 30 November 2012
Accepted date: 18 March 2013

Please cite this article as: Tiago J. Rato, Marco S. Reis, Defining the structure of DPCA models and its impact on Process Monitoring and Prediction activities, *Chemometrics and Intelligent Laboratory Systems* (2013), doi: 10.1016/j.chemolab.2013.03.009

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

ACCEPTED MANUSCRIPT

Defining the structure of DPCA models and its impact on Process Monitoring and Prediction activities

Tiago J. Rato, Marco S. Reis*

CIEPQPF, Department of Chemical Engineering, University of Coimbra,

Rua Sílvio Lima, 3030-790, Coimbra, Portugal,

**Corresponding author: e-mail: marco@eq.uc.pt, phone: +351 239 798 700, FAX: +351 239 798 703*

ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

Abstract

Dynamic Principal Components Analysis (DPCA) is an extension of Principal Components Analysis (PCA), developed in order to add the ability to capture the autocorrelative behaviour of processes, to the existent and well known PCA capability for modelling cross-correlation between variables. The simultaneous modelling of the dependencies along the "variable" and "time" modes, allows for a more compact and rigorous description of the normal behaviour of processes, laying the ground for the development of, for instance, improved Statistical Process Monitoring (SPM) methodologies, able to robustly detect finer deviations from normal operation conditions. A key point in the application of DPCA is the definition of its structure, namely the selection of the number of time-shifted replicates for each variable to include, and the number of components to retain in the final model. In order to address the first of these two fundamental design aspects of DPCA, and arguably the most complex one, we propose two new lag selection methods.The first method estimates a single lag structure for all variables, whereas the second one refines this procedure, providing the specific number of lags to be used for each individual variable. The application of these two proposed methodologies to several case studies led to a more rigorous estimation of the number of lags really involved in the dynamical mechanisms of the processes under analysis. This feature can be explored for implementing improved system identification, process monitoring and process control tasks that rely upon a DPCA modelling framework.

Keywords: Lag selection; Dynamic principal component analysis (DPCA); Multivariate statistical process control (MSPC); System Identification.

ACCEPTED MANUSCRIPT

1 Introduction Equation Chapter (Next) Section 1

Statistical Process Monitoring (SPM) methodologies have been routinely applied in many different industrial contexts, from laboratories to discrete manufacturing industries and chemical processing industries. With the increasing availability of data through faster and more informative sensors and measurement systems, the dynamical or autocorrelated nature of systems becomes an aspect that must be incorporated into SPM methodologies. The usual *i.i.d.* assumption for the definition of the Normal Operation Conditions (NOC) region is no longer valid under these circumstances, and several alternative methodologies were proposed to the classic univariate [1-3], multivariate [4-6] and mega-variate [7-10] approaches. These can be organized into three distinct classes of methods: i) methods based on correcting/adjusting control limits for the existent SPM methods, using knowledge of the specific dynamical model underlying data generation [11]; ii) methods based on time series modelling followed by the monitoring of one-step-ahead prediction residuals [12, 13]; iii) methods based on time-domain variable transformations, that diagonalize, in an approximate way, the autocorrelation matrix of process data [14, 15].

The first class of approaches (i), is restricted to very particular situations (univariate processes with rather simple dynamical structures), for which correction formulas were derived and made available. As to the time-series based approach (ii), an usually criticism concerns the difficulty of defining proper time-series model structures (the specification problem), which requires a significant amount of expertise. Perhaps even more important than this, the fact that estimating classic multivariate time-series models (e.g., VARMA, VARIMA) for small-medium sized systems (> 10 variables) is a complex or maybe unfeasible task, limits their use in practice. Finally, the third class of approaches (iii) does provide effective solutions to the autocorrelation problem, but its implementation requires a high load of computational programming. The current lack of software packages through which such methods can be conveniently made available, has been hindering their diffusion into practical applications.

ACCEPTED MANUSCRIPT

However, an alternative approach has quickly gained popularity, given its conceptual simplicity and relationship with a well-known and accepted technique: SPM using dynamic principal components analysis (DPCA) [16]. DPCA is a methodology proposed by Ku *et al.* in 1995, which essentially attempts to model the autocorrelation structure present in data, through a "time lag shift" method. This method consists in including time lagged replicates of the variables under analysis, in order to capture simultaneously the static relationships and the dynamical structure, through the application of standard PCA. DPCA has been applied in different application scenarios, that include not only multivariate process control and fault identification [16-19] but also maintenance activities planning [20] and sensitivity analysis [21]. On a different context, DPCA was also applied in economical forecasts after the initial work of Brillinger [22, 23]; other related applications include the construction and analysis of economic indicators [24] and volatility modeling [25].

A key point in the implementation of the DPCA method is the selection of the number of lags to be used, *i.e.* the number of shifted versions for each variable to include in the DPCA model. This problem is similar to selecting the lag structure in time-series models (ARMA, ARIMA, ARMAX, etc.) [26, 27]. The solution proposed by Ku *et al.* (1995) [16], consists in implementing parallel analysis, a technique that combines the scree plot obtained from a PCA analysis applied to the collected data, with the scree plot resulting from the analysis of a random data set of the same size. The interception of these two curves represents the cut-off for the selection of the number of components to retain. This is followed by the analysis of the correlations exhibited by the scores, in order to determine the number of linear relationships present in data. The underlying reasoning is that the scores corresponding to low magnitude eigenvalues correspond to the existence of linear relationships (static and/or dynamic), involving the variables under analysis, including their time-shifted versions. Such scores should also be approximately uncorrelated, as the authors illustrated with resource to several examples. Time-shifted variables are added until no additional linear relationships are detected. The existence of a new linear relationship is verified through the difference between the number of low magnitude eigenvalues (associated with uncorrelated scores) obtained with the addition of a new time-shifted variable, and the expected number of such coefficients assuming that the previous lag structure was correct.

ACCEPTED MANUSCRIPT

Other approaches to the lag-selection problem were also proposed. Autoregressive (AR) models were employed to determine the number of lags to use in DPCA [28]. In this reference, the authors suggested the application of an AR model only to the output variable, from which a single lag is proposed for all the input variables. This is a very simple approach that does not explicitly incorporate the relationships between variables. Wachs and Lewin [29] proposed the delay-adjusted PCA, that determines the most appropriated time delays, between inputs and outputs variables, by shifting inputs until their correlation with the outputs is maximized (maximum of the cross-correlation function). This approach assumes a two block variable structure (X and Y), where the output variables are correlated among themselves with no delays present, and inputs are independent of each other. The authors point out that this may not always be true, especially when analyzing closed-loop data. Guerfel *et al* [30] proposed an approach where the number of lags is selected as the minimum number needed for detecting a specific fault, therefore requiring a priory knowledge of possible systems faults. Other proposed methods result from identification techniques based on Akaike information criterion, such as those employed by Li and Qin [18] and by Russel *et al.* [19]. However, the first approach assumes a two block variable structure (X and Y) and both methodologies propose a unique delay structure for all variables, which may not be true in general.

In this article, we propose a new method to determine, in a more rigorous way, not only the maximum number of shifts to adopt in DPCA models, but also the specific lag structure for all variables. Therefore, contrary to the works published so far, the number of time-shifts used for describing the dynamic behavior of each variable can be different. Furthermore, no explicit segmentation as input/output variables is strictly required. The proposed approach thus addresses a current major weakness of the DPCA methodology, which constitutes a central problem in the implementation of the method in real world application scenarios. We illustrate the advantages of adopting the proposed method in different process system activities, such as process monitoring and system identification.

The rest of the article is organized as follows. In the next section, we present the methods used in this study, in order to set the necessary background knowledge and

ACCEPTED MANUSCRIPT

clarify nomenclature. Next, we briefly review the approach of Ku *et al.* for lag selection in DPCA [16], and introduce our proposed method. Then, the results regarding the comparison of the two approaches for estimating the lag structure (the method of Ku *et al.*, and our method), are presented. The advantages of the proposed methodology are illustrated through several case studies regarding process monitoring and system identification applications. Finally, we summarize the contributions presented in this paper and conclude with some remarks regarding future work.

2 Methods

In this section we briefly review the methods involved in the work presented in this article. As most of them are well known and extensively referred in the literature, such as PCA [31-33] and DPCA [16], we provide just a short overview, mostly for the purpose of setting the nomenclature to be followed in the next sections.

2.1 Principal components analysis (PCA)

PCA is essentially a data reduction technique that compresses the original m-dimensional space of original variables into a lower dimensional subspace of latent variables, say with p dimensions, by retaining its dominant variability features while discarding the random, non-structured variation. The latent variables are usually called the Principal Components (PCs), which are linear combinations of the original ones. The PCs are organized in a sequence with decreasing explanation power of the whole original variability. With this ordering, the first few components are usually able to retain most of the variability present in all the original variables. The PCs are also uncorrelated quantities. The basic equation connecting the ($n\times m$) matrix of original variables, $\mathbf{X}$, (variables are always disposed column-wise in this article) and the ($n\times p$) matrix with the new transformed variables, $\mathbf{T}$, called the scores matrix (each columns contain the scores for a given PC), is the following one:

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

ACCEPTED MANUSCRIPT

where **P** is the ($m \times p$) matrix containing the coefficients of the linear combinations, also called loadings (loading matrix)and **E** is the ($n \times m$) residual matrix. In this representation n is the number of observations, m the number of variables. The number of retained PC's (p, also called the pseudo-rank of **X**) is defined using one of the methodologies available, such as the Kaiser method, Scree test, parallel analysis, information theory (Akaike information criterion, minimum description length), cross-validation and cross-validation ratio, variance of the reconstruction error, F-test, among others [31, 34-36].

PCA is a scale-dependent technique, which means that its outcome depends on any multiplicative factor affecting the variables. Therefore, in order to balance the specific weight of each variable in the analysis, they are usually pre-processed in some meaningful way. The most well-known pre-processing procedure consists in scaling all variables to zero mean and unit variance, called *autoscaling*.

2.2 Dynamic principal components analysis (DPCA)

PCA is a multivariate methodology devoted to the analysis of the correlation structure linking the variables mode of the **X** matrix. When this is the only relevant structure present in data, the method can be very useful and efficient in summarising all the regularities in the data, either regarding variables (groups of correlated variables) or observations (clusters, trends, outliers). Under these conditions, all the important features are contained in the scores, loadings and residuals, for which several tools were developed to facilitate the subsequent analysis. However, if the **X** matrix also presents correlation along the observations mode, i.e., if variables have autocorrelation, PCA does not provide the full picture of the data correlation structure because of its blindness regarding such dimension. In other words, PCA tacitly assumes that all variables are uncorrelated in time, and is better applied in contexts where such hypothesis is valid, at least with good approximation. But this is a feature that is often not met in practice, especially in systems with inertia-inducing units or simply as a consequence of the high sampling rates that are currently easily achieved by modern instrumentation.

ACCEPTED MANUSCRIPT

In order to address this issue, Ku *et al.* [16] tried to incorporate the description of variables autocorrelation into the standard PCA framework, by introducing time-shifted replicates as additional variables in the **X** matrix. This extended matrix, $\tilde{\mathbf{X}}$, opens the possibility to model the relationships between variables (correlation) and between observations (auto-correlation and cross-correlation, depending on whether the variables involved are the same or not). In fact, it amounts to an implicit linear time series model structure that is incorporated into a conventional PCA framework, representing a (latent) vector autoregressive (VAR) process. Processes containing moving average (MA) terms will be approximated by a finite length VAR model. The inclusion of time-shifted variables can be represented according to Equation(2),

$$\tilde{\mathbf{X}} = \left[\overbrace{\mathbf{x}_1(0)\cdots\mathbf{x}_m(0)}^{\mathbf{x}(0)} \; \overbrace{\mathbf{x}_1(1)\cdots\mathbf{x}_m(1)}^{\mathbf{x}(1)} \cdots \overbrace{\mathbf{x}_1(l)\cdots\mathbf{x}_m(l)}^{\mathbf{x}(m)} \right] \tag{2}$$

where $\mathbf{x}_i(j)$ represents the i^{th} variable (in column format) shifted j times into the past (i.e., with j lags), $\mathbf{x}_i(j)[k] = \mathbf{x}_i(0)[k-j]$ (the indices inside square brackets are the entry identifiers of the column vectors $\mathbf{x}_i(j)$ and $\mathbf{x}_i(0)$, respectively); in equation (2), $\mathbf{x}(j)$ is the submatrix containing all the original variables shifted j times; and $\tilde{\mathbf{X}}$ is the resulting extended matrix (with l lags). Written in this way, the extended matrix has the form of a Hankel matrix, which is found frequently in System Identification methodologies and procedures [27]. However, defining the fine lag-structure for DPCA may involve the use of different lags for the variables, and the final extended matrix may no longer retain such a simple shape.

Therefore, in simple terms, DPCA is essentially the same as the original PCA approach, except that the data matrix is now composed of additional time shifted replicates of the original variables. The central problem is then, how to define properly and in a consistent and rigorous way, the number of lags to adopt, l, in order to capture both the static (between variables) and dynamic (across observations) relationships. After the construction of the extended matrix of process variables, Ku *et al.* [16] proposed a methodology based on the analysis of the noise subspace, which is composed by the PCs with small eigenvalues associated. This methodology is shortly described in the next section.

ACCEPTED MANUSCRIPT

2.3 Selecting the lag structure in DPCA

In this section we briefly review the benchmark method for selecting the number of time-shifted replicates to include in the extended matrix, and present the methodologies proposed in this work. The benchmark is the method most extensively described in the literature dealing directly with the lag structure definition problem [16].

2.3.1 Defining the lag structure using the method proposed by Ku *et al.*

When the last PCs have little variation, the corresponding eigenvector represents an exact or near-exact linear relation between the original variables [32]. This characteristic was explored by Ku *et al*. [16], who proposed an algorithm based on the identification of the number of linear relationships needed to describe the system, in order to determine the number of lags (l) to be used in the definition of the extended matrix for a DPCA model. The extended matrix has, in this case, the simple form of a Hankel matrix. The presence of linear relationships, originated from static or dynamic relations, manifests itself through two types of effects: i) small eigenvalues in the spectral decomposition of the covariance matrix for $\tilde{\mathbf{X}}$, ii) and by the fact that the corresponding scores should be, in these conditions, approximately independent (a feature that can be checked through, for instance, auto- and cross-correlation plots). The pseudocode for the algorithm proposed by the authors is presented in Table 1 [16].

ACCEPTED MANUSCRIPT

Table 1. Pseudocode for the lag-selection methodology proposed by Ku *et al*. [16].

1.	Set $l = 0$;
2.	Form the extended data matrix $\tilde{\mathbf{X}} = \left[\mathbf{x}(0)\ \mathbf{x}(1) \cdots \mathbf{x}(l)\right]$;
3.	Perform PCA and calculate all the principal scores;
4.	Set $j = m \times (l+1)$ and $r(l) = 0$;
5.	Determine if the j^{th} component represents a linear relation. If yes proceed, if no go to step 7;
6.	Set $j = j - 1$ and $r(l) = r(l) + 1$, repeat 5;
7.	Calculate the number of new relationships: $$r_{new}(l) = r(l) - \sum_{i=0}^{l-1} (l - i + 1) r_{new}(i) \quad (3)$$
8.	If $r_{new}(l) \leq 0$, go to step 10, otherwise proceed;
9.	Set $l = l + 1$, go to step 2;
10.	Stop.

The above procedure assumes the implementation of a methodology for selecting the number of principal components, for which the authors suggest the use of parallel analysis followed by the analysis of cross-correlation plots of the principal component scores. The number of linear relationships identified is given by the difference between the number of variables considered in the extended matrix and the number of principal components retained.

According to the authors, the number of lags obtained by this procedure is usually 1 or 2, depending on the order of the dynamic system, and is the same for all variables. However, they refer that, in the case of nonlinear systems, l could be set to higher values, in order to get a better linear approximation of the nonlinear relationships [16].

This methodology will be adopted in this work as the benchmark method against which the proposed approaches will be tested and compared. In this decision, we took into account the fact that it is eventually the most well known and widely used approach for addressing this problem, which furthermore is theoretical driven and thoroughly tested. Other less tested methods available in the literature either present similar limitations or require the consideration of two blocks of variables (please refer to the review presented in the Introduction section). However, to the best of our knowledge, no method is

ACCEPTED MANUSCRIPT

available yet, that is able to estimate the complete lag structure, as Method II described bellow.

2.3.2 Defining the coarse lag structure: Method I – Selection of the maximum number of lags

In this section, we propose an alternative way to select a single number of time-shifts for all variables, regarding that proposed by the benchmark method. In this case, the extended matrix to be used in DPCA has the form of a Hankel matrix. This method can be used separately, or in a first stage preceding the implementation of Method II, to be described in the next section, which will refine the number of shifts to consider for each individual variable.

The proposed method has roots in the work of Ku *et al.* [16], and is focused on following the singular values obtained in each stage, where the extended matrix is augmented with the replicates for an additional time-shift. By analysing the sequence of the singular values, one can estimate the point (i.e., the number of lags), after which the introduction of additional time-shifted variables become redundant, i.e., are no longer necessary for explaining the existent stationary and dynamic relationships.

The proposed method assumes the existence of a total of m linear dynamical relationships to be identified (as many as the variables involved), whose order is not known a priori, but is at least 1 (the simplest dynamical model). This simple hypothesis allows for the derivation of an algorithm that consistently leads to better estimates of the dynamical lag-structure involved, as illustrated in the results section. The algorithm consists in sequentially introducing an additional set of time-shift replicates for all original variables (which corresponds to the consideration of one more lag in the extended matrix), after which the singular values are computed for the corresponding covariance matrix. This procedure is repeated until a pre-defined upper limit on the number lags to use is achieved, l_{max}, where it stops (this limit is usually a number high enough for allowing the description of all the dynamical features present in the data, but it can be adjusted if, during the analysis, it is concluded that it was underestimated

ACCEPTED MANUSCRIPT

initially). In each stage, l (which also coincides with the number of lags introduced in the extended matrix), the following quantities are computed from the singular values:

- ***Key Singular Value***. The key singular value in the l^{th} stage ($l \geq 1$), ***KSV***(l), is defined as the $(m \times l + 1)^{th}$ singular value, after sorting the set of singular values according to the decreasing order of their magnitude. As there are m linear dynamical relationships to be identified, by hypothesis, only after introducing the number of lags necessary to model the linear dependency with the highest delay (i.e., that goes further into the past, or having the maximum number of lags in the DPCA model), one is expected to get a small magnitude for the singular value in such a position. Therefore, the key singular value should signal the point where the linear relationship requiring (more lags to be properly described) was finally achieved. The other relationships, requiring a lower number of lags, give rise to multiple low singular values (one per additional lag, after the numbers of lags necessary to fully describe their linear relations are achieved). All of them appear at the end of the ordered vector of singular values. It can be shown that, only after the point where there is a sufficient number of lags to describe all linear descriptions present, a small value appears in the *KSV*, indicating that the addition of variable replicates with more lags is no longer relevant for the DPCA model.

- ***Key Singular Value Ratio***. We have seen that, under the assumption of having m dynamical relationships present, a small value for the *KSV* at stage l, indicates that one has attained the point where no more lags necessary to add. However, due to the presence of noise and small-moderate non-linear effects, the identification of this condition is not always clear. Therefore, we introduce a second element in the algorithm, in order to increase the robustness of the detection of the maximum number of lags required. In fact, following the behaviour of the successive values obtained for the *KSV*(l), (l=1,2,…), we have empirically observed that there is a point where it decays more or less sharply (corresponding to the stage where *KSV*(l) starts getting low values) and then becomes approximately constant (see the left plot in Figure 1, for an example). Therefore, by defining the Ratio of successive Key Singular Values at stage l,

ACCEPTED MANUSCRIPT

KSVR(*l*), by $KSVR(l) = KSV(l) / KSV(l-1)$, one can capture this behaviour more effectively. In fact, from this definition, one can verify that the required number of lags should have a low value for *KSVR*, indicating that a significant decrease in the *KSV* has just occurred, i.e., the current singular value is significant lower than the previous one. After this point, the ratio tends to have values closer to 1 and to be approximately constant.

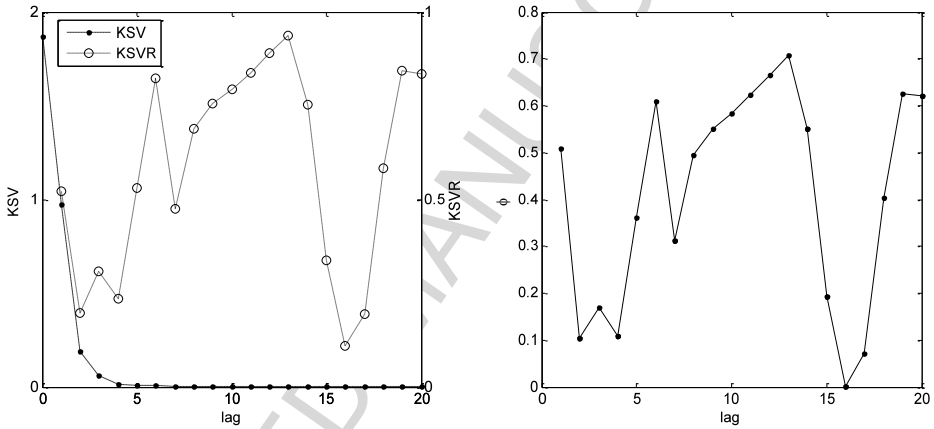


Figure 1. *KSV* and *KSVR* obtained in the analysis of the Wood and Berry case study (left). The analysis of the parameter *KSVR*, leads, in this case, to an estimated maximum number of lags of 16. Also shown, is the objective function for the auxiliary optimization problem for selecting the number of lags (right).

In resume, the maximum number of lags to be considered in the extended matrix for implementing DPCA should obey the following two criteria: (i) have a small *KSV* and (ii) have a low value for *KSVR*.

In order to find the number of lags that match both of these conditions, we implement a procedure that seeks for the number of lags introduced, *l*, for which *KSV* and *KSVR* are closer to their minimums – the minimums attained individually in the analysis, i.e., min(*KSV*) and min(*KSVR*). This task is performed by minimizing the objective function, *Distance To Optimum*, ϕ, given by Equation (4), where KSV_N and $KSVR_N$ are normalized versions of *KSV* and *KSVR*, constructed in order to remove the effects of scale and provide equal weight to both criteria in the analysis (Equations(5) and (6)).

ACCEPTED MANUSCRIPT

$$\phi = \sqrt{KSV_N(l)^2 + KSVR_N(l)^2} \tag{4}$$

$$KSV_N(l) = \frac{KSV(l) - \min(KSV)}{\max(KSV) - \min(KSV)} \tag{5}$$

$$KSVR_N(l) = \frac{KSVR(l) - \min(KSVR)}{\max(KSVR) - \min(KSVR)} \tag{6}$$

The pseudocode for the proposed algorithm for estimating the maximum number of lags to use in DPCA, is presented in Table 2. The plot in the right hand side of Figure 1 illustrates this objective function for the Wood and Berry case study.

Table 2. Pseudocode for the Method I: selection of the maximum number of lags to use in DPCA.

1. Set $l = 0$;
2. Form the extended data matrix $\tilde{\mathbf{X}} = \left[\mathbf{x}(0)\ \mathbf{x}(1) \cdots \mathbf{x}(l)\right]$;
3. Perform the singular value decomposition of the covariance of the extended matrix: $\boldsymbol{\Sigma}_{\tilde{\mathbf{X}}} = \mathbf{USV}^{\mathrm{T}}$;
4. Set $KSV(l) = s_{ml+1}$;[(1)]
5. If $l > 0$ set $KSVR(l) = KSV(l)/KSV(l-1)$;
6. If $l < l_{max}$, set $l = l + 1$ and go to step 2, otherwise proceed;
7. Normalize *KSV* and *KSVR*;
8. Determine: $\arg\min_{l \in [1, lmax]} \sqrt{KSV_N(l)^2 + KSVR_N(l)^2}$, s.t. $l \geq l^*$ (where l^* is the first l, such that $KSVR(l) < KSVR(l-1)$);[(2)]

Notes: [(1)] s_{ml+1} is the $(ml+1)^{\text{th}}$ singular value of $\boldsymbol{\Sigma}_{\tilde{\mathbf{X}}}$. [(2)] A justification for this condition is provided in section 3.1.3.1.

2.3.3 Defining the fine lag structure: Method II – Selection of the number of lags for each variable

Method I provides an approach for selecting a single number of lags to be used for all variables (as in the benchmark method). Such lag is the one corresponding to the dynamic relation requiring a longer tail into the past, in terms of the number of lags involved. However, analysing different multivariate systems, one can verify that quite

ACCEPTED MANUSCRIPT

often the order of the dynamics is not the same for all variables. Therefore, the number of lags required to describe them will also be different. Under these circumstances, it is both opportune and important to devise a methodology for fine tuning the number of lags to be adopted for each variable, in order to increase de accuracy and stability of the DPCA approach. In order to obtain such a finer selection of the number of lags for each variable, we propose a second algorithm (Method II) that presents some similarities with Method I, but that includes, in each stage, a variable-wise analysis.

In Method II, at each stage, k, m versions of the extended matrix are analyzed, $\tilde{\mathbf{X}}^1(k), \tilde{\mathbf{X}}^2(k), \ldots, \tilde{\mathbf{X}}^m(k)$. These matrices are obtained from that relative to the preceding stage, $\tilde{\mathbf{X}}(k-1)$, modified by the inclusion of a single time-shifted variable, with one more lag than the number of lags used in the preceding stage for the same variable. In more precise terms, let us define $l^{(k)}$, as the m-dimensional vector containing in its entries the number of lags considered for each variable in stage k: $\lambda_i^{(k)}$ ($i = 1, \ldots, m$). In other words, the first entry of $l^{(k)}$ contains the number of shifted versions for x_1, $\lambda_1^{(k)}$, the second contains the number of shifted version for x_2, $\lambda_2^{(k)}$, and so on and so forward, until the m^{th} entry:

$$l^{(k)} = \begin{bmatrix} \lambda_1^{(k)} & \lambda_2^{(k)} & \cdots & \lambda_m^{(k)} \end{bmatrix} \tag{7}$$

Let us also consider the m-dimensional indicator vector, $\delta(k)$, as a vector composed by zeros, except for the k^{th} position, where it has a 1:

$$\delta^{(k)} = \begin{bmatrix} \overset{1}{0} & \overset{2}{0} & \cdots & \overset{k}{1} & \cdots & \overset{m}{0} \end{bmatrix} \tag{8}$$

In this circumstances, the lag structure corresponding to the i^{th} version of the extended matrix at stage k, $\tilde{\mathbf{X}}^i(k)$, summarized in the vector $l_i^{(k)}$, is given by (see also Figure 2 for more details about this process):

ACCEPTED MANUSCRIPT

$$l_i^{(k)} = l^{(k-1)} + \delta^{(i)} \rightarrow \tilde{\mathbf{X}}^i(k) \tag{9}$$

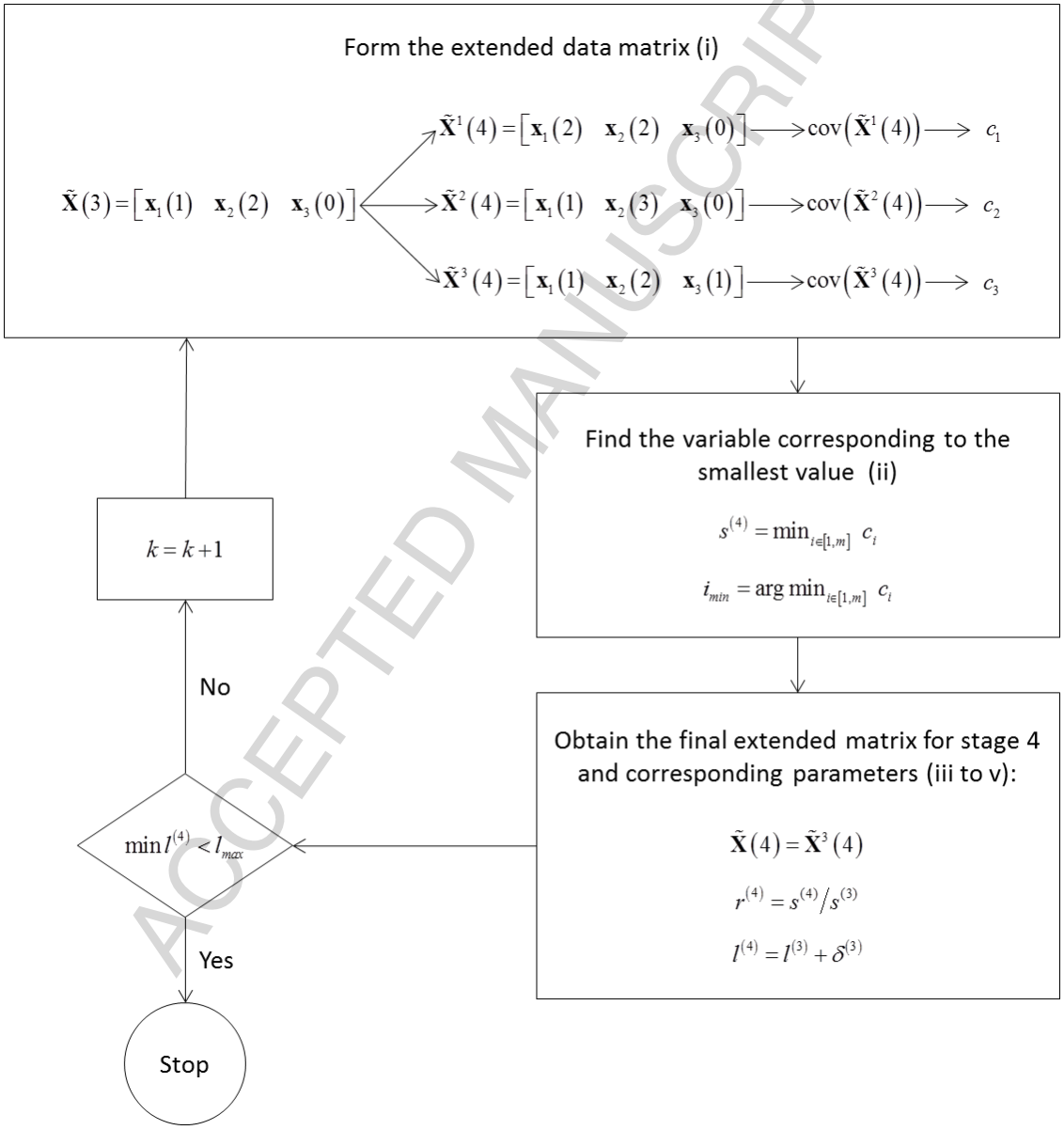


Figure 2. Illustration of the implementation of Method 2. In this example, the stage 4 of a 3 variables system is presented. On the previous stages 1 lag was selected for $\mathbf{x}_1$ and 2 lags to $\mathbf{x}_2$. In the current stage, the several versions of the extended data matrices are defined (three in this case), and their singular values determined. Based on the analysis of quantities computed from them, variable $\mathbf{x}_3$ is selected to be incorporated into the extended matrix with an additional time-shift.

ACCEPTED MANUSCRIPT

Now, for each version of the extended matrix, $\tilde{\mathbf{X}}^i(k)$, the smallest singular value of the corresponding covariance matrix, $\mathrm{cov}\left(\tilde{\mathbf{X}}^i(k)\right)$, is determined and saved. Let us call it, c_i. This set of values, c_i (i= 1, …,m), are then analyzed in order to find the minimum at stage k, say ($s^{(k)}$). The lagged variable to be incorporated in the extended matrix is the one for which such minimum was found. This will result in the extended matrix for stage k, $\tilde{\mathbf{X}}(k)$, and the procedure is repeated again for stage k+1, where another lagged version of some variable is added again (which can be any of the m variables under consideration). This process continuous until the maximum number of lags to analyze is attained. The total number of stages will always be equal to the sum of the lags for all variables, as in each stage a single lag is introduced for some variable under analysis. The maximum number of lags to analyze, $l_{\max}$, is a parameter that can be either provided after the implementation of Method I, or selected in a more conservative way (i.e., slightly above of what we expect to be a reasonable value for this parameter).

In order to remove potentially redundant lagged variables that might be included in this forward addition process, a final pruning stage is performed, where the results obtained at all stages are analyzed for their significance and improvement of the objective function, using a similar criterion to the one presented before, in Equation (4) (the only difference lies in the redefinition of the normalization factors). The complete procedure for implementing Method II is summarized in the pseudocode presented in Table 3.

ACCEPTED MANUSCRIPT

Table 3. Pseudocode for the algorithm that estimates the fine lag structure of the extended matrix, for implementing DPCA (Method II).

1. Set stage $k=0$, and initialize the lag vector (whose i^{th} entry, contains the number of lags for the corresponding i^{th} variable), $l^{(k)}=[0 \quad \cdots \quad 0]$;
2. Form $\tilde{\mathbf{X}}(0)$ and determine the smallest singular value ($s^{(0)}$);
3. Set $k=k+1$;
   i. For $i=1$ to m
      - Form $\tilde{\mathbf{X}}^i(k)$, whose lagged structure is given by the vector: $l_i^{(k)}=l^{(k-1)}+\delta^{(i)}$;
      - Compute the covariance matrix of $\tilde{\mathbf{X}}^i(k)$;
      - Determine the smallest singular value of the covariance matrix obtained in the previous step: c_i;

   ii. Find the variable corresponding to the smallest value of the set $\{c_i\}_{i=1,\ldots,m}$: $s^{(k)}=\min_{i\in[1,m]} c_i$; **(1)**

   iii. Obtain the final extended matrix for stage k: $\tilde{\mathbf{X}}(k)$; with $l^{(k)}=l^{(k-1)}+\delta^{(i_{min})}$, where $i_{min}=\arg\min_{i\in[1,m]} c_i$; **(1)**

   iv. Set $r^{(k)}=s^{(k)}/s^{(k-1)}$;

   v. If $\min l^{(k)} < l_{max}$, go to step 3, otherwise proceed;
4. Determine the stage that provides the best description of the linear dynamics involved: $k^*=\arg\min_{k\in[1,k_{\max}]}\phi(k)=\sqrt{\left(\tilde{s}^{(k)}\right)^2+\left(\tilde{r}^{(k)}\right)^2}$ **(2)**, s.t. $l\geq l^*$ (where l^* is the first l, such that $r(l)<r(l-1)$ **(3)**).

Notes: **(1)** in ii), $s^{(k)}$ contains the minimum singular values, whereas in iii), i_{min} corresponds to the index for the minimum value.
(2) $\tilde{y}$ represents the normalized score corresponding to y.
(3) A justification for the condition used in stage 4 is provided in section 3.1.3.1.

3 Results

In this section we present the results obtained from the application of the methods presented in the previous section, to different test scenarios. In the first subsection, we demonstrate the improved accuracy in estimating the lag structure obtained with the proposed methodologies. Then, we illustrate the consequences of using such improved methods, in the tasks of statistical process monitoring (SPM) and systems identification (SI).

ACCEPTED MANUSCRIPT

3.1.1 Comparative assessment study

In order to demonstrate the increased lag estimation accuracy of the proposed methodologies, we consider two testing scenarios. In the first scenario, a large number of systems from the same class were randomly generated, and then the Benchmark and Method I were employed in order to estimate the appropriate maximum number of lags necessary to describe the dynamical relationship for each realization of the model structure. In the second scenario, several multivariate systems found in the literature, with known dynamics, are employed, in order to test the estimation accuracy performances of Method II in selecting the specific number of lags for each variable.

3.1.2 Systems with random lag structure

In order to access the accuracy performance of the methods presented in section 2, they were employed in the estimation of the number of lags for a large number of systems with randomly generated structures. The systems under study were based on the following continuous first order dynamic transfer function with time delay, defined by

$$g = \frac{Ke^{-\theta s}}{\tau s + 1} \tag{10}$$

where K is the system gain, τ the time constant and θ the time delay. A large number of different realizations of this set of parameters were generated (following independent uniform distributions), which will imply different time lags in the corresponding discrete models.

Following this procedure, 5000 SISO systems and another 5000 MIMO 2x2 systems were generated, all of them subjected to additive white noise (d) with different magnitudes of signal-to-noise ratio (SNR) and noise structures (with and without autocorrelation). The SNR is defined by:

$$SNR = 10\log_{10}\left(\frac{\mathrm{var}(x)}{\mathrm{var}(d)}\right) \tag{11}$$

ACCEPTED MANUSCRIPT

The deviations obtained between the estimated and the true maximum delay of the system (equivalent to the maximum lag), are presented graphically in Figures 3 to 6.

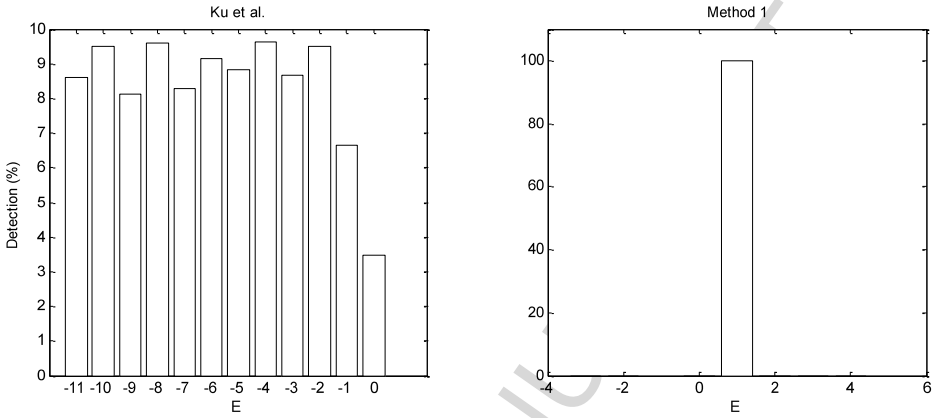


Figure 3. Graphical representation of the deviation between the estimated and the true number of lags (Estimated - True) for SISO systems, corrupted with additive white noise (20 dB), without autocorrelation.

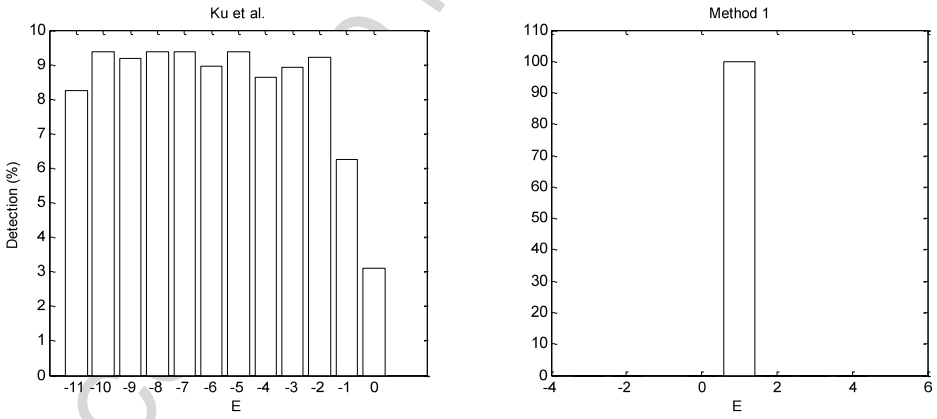


Figure 4. Graphical representation of the deviation between the estimated and the true number of lags (Estimated - True) for SISO systems, corrupted with additive autocorrelated noise (20 dB).

ACCEPTED MANUSCRIPT

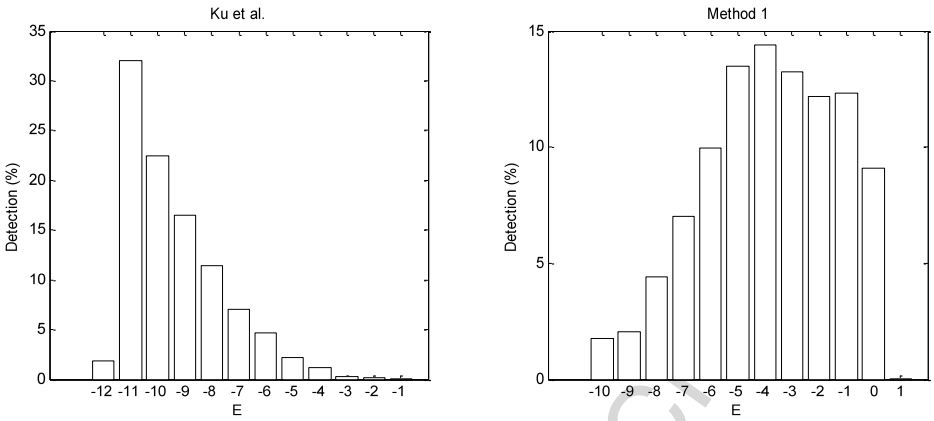


Figure 5. Graphical representation of the deviation between the estimated and true number of lags (Estimated - True) for MIMO 2x2 systems, corrupted with additive white noise (20 dB), without autocorrelation.

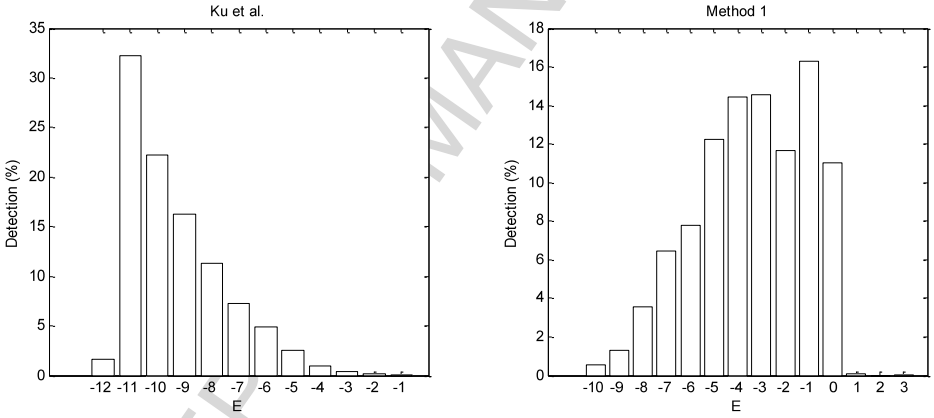


Figure 6. Graphical representation of the deviation between the estimated and true number of lags (Estimated - True) for MIMO 2x2 systems, corrupted with additive autocorrelated noise (20 dB).

Regarding SISO systems, Figures 3 and 4 indicate that the deviations distribution from application of Ku *et al.* method is almost uniform between -11 and -2. This may happen because the true delays were also generated by an uniform distribution and the estimates from this method typically vary little, indicating a 0-lag model in about 95% of the SISO systems, and a 1-lag model in approximately 95% of the MIMO 2x2 systems. These results are consistent with the authors' comments about their method, namely that it usually provides estimates of the systems order in the range 1-2. However, the method fails in the estimation of the true number of lags, which necessarily leads to less adequate DPCA models, which are not correctly modelling the dynamic behaviour of the systems.

ACCEPTED MANUSCRIPT

On the other hand, our proposed method estimates 1 more lag than the correct one in all the generated SISO systems (Figures 3 and 4). In the case of the MIMO 2x2 systems (Figures 4 and 5), one can see that it also presents some estimation error, which nevertheless is lower than that for the Ku *et al.* method. In fact, the absolute deviations obtained with the Ku *et al*. method (e_1) are greater than the ones obtained by our proposed method (e_2). This can be seen in Figure 7, where the difference $D = e_1 - e_2$ is represented. From this figure we can conclude that our approach indeed presents higher estimation accuracy of the true number of lags. The statistical significance of the difference between the methods, was also assessed with a permutation test [37] leading to highly significant *p*-values, much lower that 0.01, confirming the lower absolute deviations obtained with the proposed method.

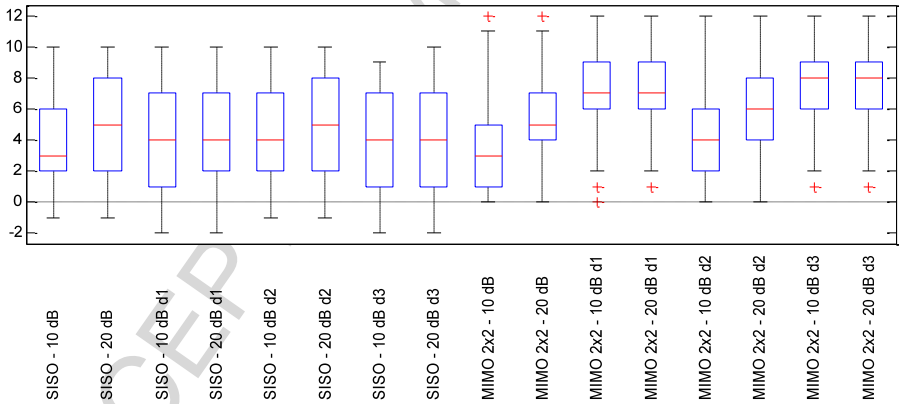


Figure 7. Graphical representation of the difference between the absolute deviation obtained on the number of lags estimated by the Ku *et al.* method and our proposed method. d1, d2 and d3 refer to 3 different dynamic transfer functions applied to the additive autocorrelated noise.

3.1.3 Multivariate dynamic systems collected from the literature

We have also applied the lag estimation methods to three MIMO systems found in the literature, namely those proposed by (i) Wood and Berry, (ii) Wardle and Wood, and (iii) Ogunnaike and Ray. The corresponding transfer functions are presented in Table 4 [38].

ACCEPTED MANUSCRIPT

In this study the proposed method was applied in two parts: in the first part, we employed Method 2 for selecting the number of lags for the output variables only. In the second part, the same was done for the input variables, using the previously selected number of lags for the outputs. This procedure turned out to be the most effective one for handling complex higher-order dynamical systems.

Table 4. Transfer functions for the three MIMO systems used in the comparison study.

	Wood and Berry	Wardle and Wood	Ogunnaike and Ray
g_{11}	$\frac{12.8e^{-s}}{16.7s+1}$	$\frac{0.126e^{-6s}}{60s+1}$	$\frac{0.66e^{-2.6s}}{6.7s+1}$
g_{12}	$\frac{-18.9e^{-3s}}{21s+1}$	$\frac{-0.101e^{-12s}}{(48s+1)(45s+1)}$	$\frac{-0.61e^{-3.5s}}{8.64s+1}$
g_{13}			$\frac{-0.0049e^{-s}}{9.06s+1}$
g_{21}	$\frac{6.6e^{-7s}}{10.9s+1}$	$\frac{0.094e^{-8s}}{38s+1}$	$\frac{1.11e^{-6.5s}}{3.25s+1}$
g_{22}	$\frac{-19.4e^{-3s}}{14.4s+1}$	$\frac{-0.12^{-8s}}{35s+1}$	$\frac{-2.36e^{-3s}}{5s+1}$
g_{23}			$\frac{-0.01e^{-1.2s}}{7.09s+1}$
g_{31}			$\frac{-34.68e^{-9.2s}}{8.15s+1}$
g_{32}			$\frac{46.2e^{-9.4s}}{10.9s+1}$
g_{33}			$\frac{0.87(11.6s+1)e^{-s}}{(3.89s+1)(18.8s+1)}$
True number of lags (after discretization)	$[2 \quad 2 \quad 9 \quad 5]$	$[2 \quad 2 \quad 10 \quad 15]$	$[3 \quad 3 \quad 4 \quad 14 \quad 14 \quad 5]$

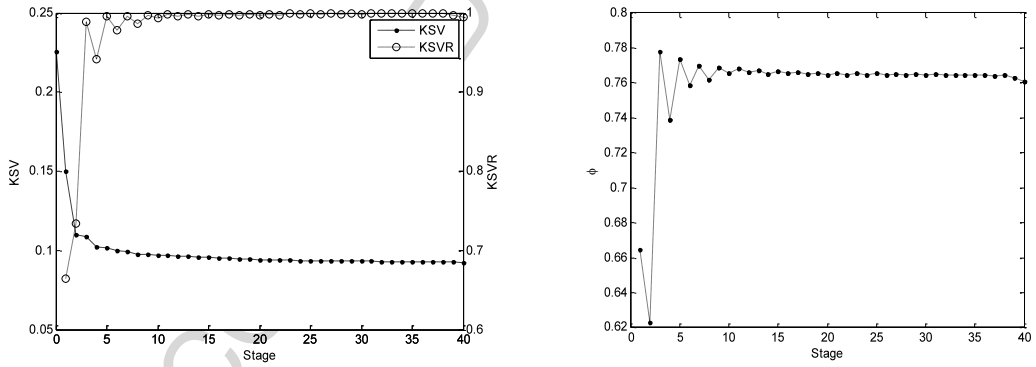
The true number of lags for each system is a function of the sampling rate adopted, and was determined here through the Matlab function *c2d* (that converts a continuous model into a discrete one), and by the transformation equations described by Roffel [39]. All systems were simulated using the Matlab function *lsim*, and subject to a SNR of 10 dB. Each data set analyzed was composed by 3000 samples.

ACCEPTED MANUSCRIPT

3.1.3.1 Wood and Berry system

By application of our proposed method to the Wood and Berry system, on the first part of the procedure, the number of lags of the outputs was determined according to the algorithm presented in Table 3. This algorithm makes use of the singular values of a data matrix successively extended by additional lagged variables and the ratio between the singular values before and after the inclusion of each lagged variable. These values are introduced into an optimization function (ϕ), from which the lagged variable leading to the lower value is the one to be included in the extended matrix. This procedure is repeated until the maximum number of stages is achieved.

In the case of the Wood and Berry systems, the results obtained in each stage are presented in Figure 8.



(a) (b)

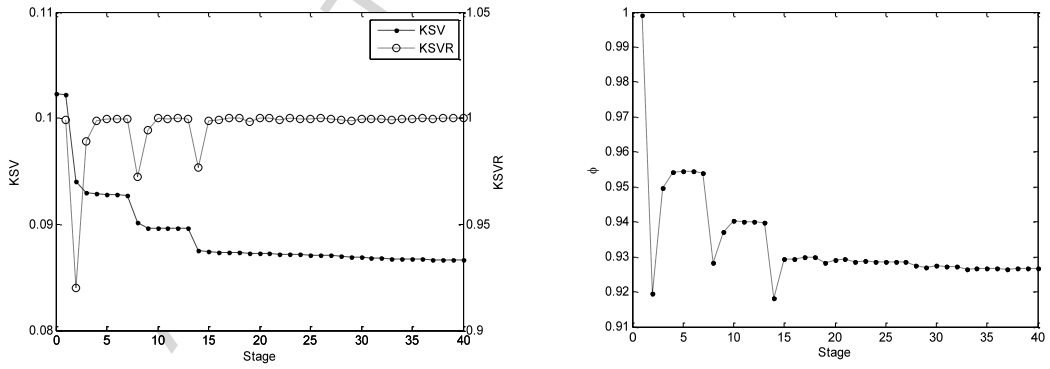
Figure 8. Graphical representation of (a) the singular values, their ratio and (b) the output from the optimization algorithm, ϕ, during the first part of the proposed method, in the Wood and Berry system.

From Figure 8 (b) we can verify that stage k=2 led to the lowest value of ϕ. However, this value is greatly affected by the singular values ratio, $\tilde{r}^{(k)}$ (see Table 3, stage 4), which is expected to be low in the first stages, because of the rapid decrease in the singular values (s) after the first inclusions of lags. Furthermore, the use of the ratio in the optimization functions, ϕ, is mainly for the purpose of identifying significant changes in s, once its value is already low, and not in the initial stages. This is the reason way the ratio condition is present in the proposed methods (Method 1 see Table 2, stage 8; Method 3 see Table 3, stage 4). It should be also noted, that the decreasing

ACCEPTED MANUSCRIPT

profile of s is related to the amount of variability explained. Thus, after the point where all significant lagged variables have been included, the decrease on the singular values will be lower. Consequently, the desirable combination of lags should have a low value for the singular values, s, followed by an almost constant profile (which is equivalent to a ratio near 1). From Figure 8 (a) we observe that stage 4 (that has the second lowest ϕ) has a closer match to these specifications, and was therefore, chosen for providing the number of lags in the output variables.

On the second part of the method, the number of lags for the input variables is also determined, given the information about the number of lags for the output variables. Following the same procedures and considerations, we select stage 14 for the outputs (see Figure 9). In this part, the singular value profile is not as ambiguous as in the outputs case, and a clearer, almost constant, profile appears after stage 14 (see Figure 9 (a)), which is also identified by the lowest of ϕ. By selecting stage 14, we obtain a lag vector of $\begin{bmatrix}2 & 2 & 9 & 5\end{bmatrix}$ which is equal to the theoretical one (see Table 4).



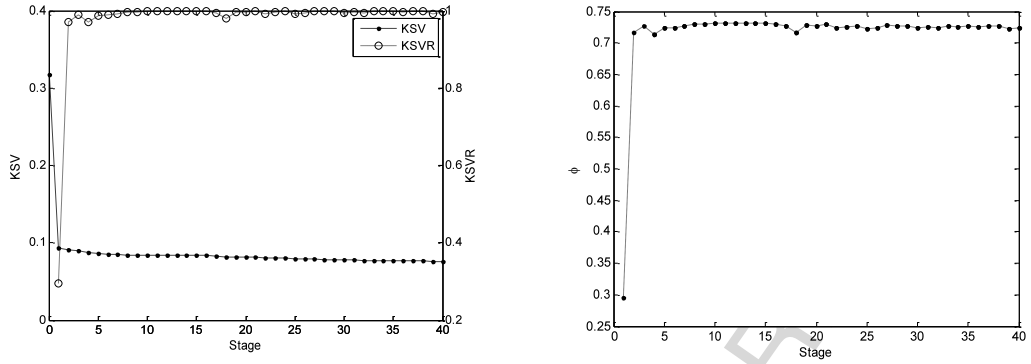
(a) (b)

Figure 9. Graphical representation of (a) the singular values, their ratio and (b) ϕ, during the second part of the proposed method, in the Wood and Berry system.

3.1.3.2 Wardle and Wood system

The second case study was the Wardle and Wood system. The two part procedure was also implemented, as for the previous example. From the first part of the method we obtained the profiles presented in Figure 10.

ACCEPTED MANUSCRIPT



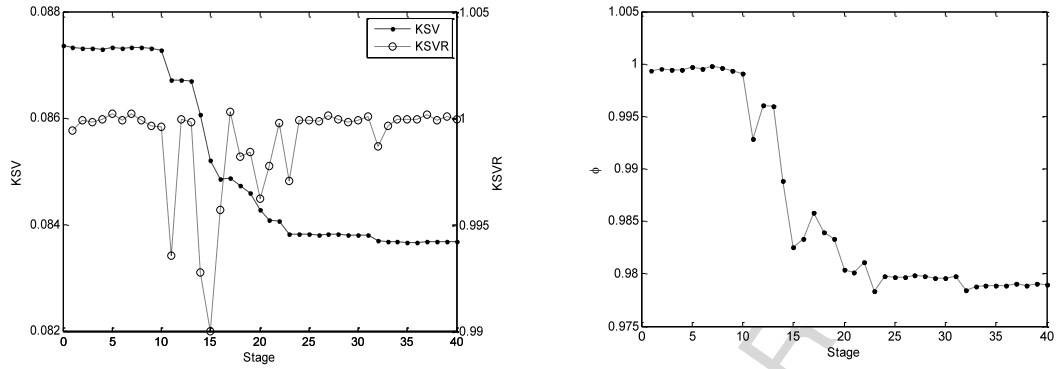
(a) (b)

Figure 10. Graphical representation of (a) the singular values, their ratio and (b) ϕ, during the first part of the proposed method, in the Wardle and Wood system.

As can be seen on Figure 10 (b), the minimum of ϕ is obtained on stage 1. However, as in the case of the Wood and Berry system, the ratio in stage 1 has a big effect on the value of ϕ and it also results from the high reduction in the singular value after the addition of the first lagged variable, and therefore should not be considered, in accordance with the ratio condition. Stage 4 respects this condition and has the 2nd lowest ϕ, along with an almost constant singular values profile after it. Given these considerations, we select stage 4 for the first part of the method.

In the second part of the method, we obtain the minimum value of ϕ in stage 23 (see Figure 11(b)). Note that this stage is also related to the almost constant profile of the singular values (Figure 11(a)), even with the small decrease in stage 32, that is considered as less relevant due to its smaller ratio. Therefore, stage 23 is the one selected, giving a lag vector of $\begin{bmatrix}2 & 2 & 10 & 14\end{bmatrix}$ which is quite similar to the theoretical one, $\begin{bmatrix}3 & 2 & 10 & 15\end{bmatrix}$ (see Table 4).

ACCEPTED MANUSCRIPT



(a) (b)

Figure 11. Graphical representation of (a) the singular values, their ratio and (b) ϕ, during the second part of the proposed method, in the Wardle and Wood system.

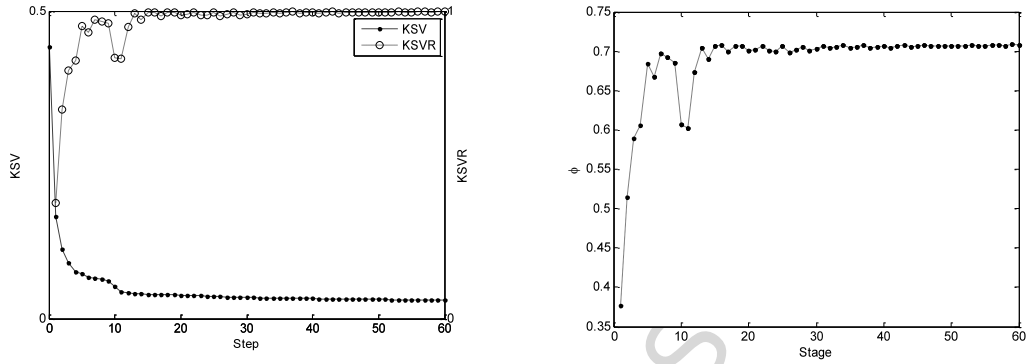
3.1.3.3 Ogunnaike and Ray system

The third case study regards the Ogunnaike and Ray system. From the application of the proposed method, we obtained the profiles presented in Figure 12 for the output variables (first part). In this case, the progressive decrease in s is more evident, and therefore a careful analysis of the first stages should be conducted. As can be seen in Figure 12 (b), the first stages have the lowest values of ϕ. However, as explained before, the first stages correspond to a rapid, but not stable, reduction of s. Note that the required stage of almost constant values of s, only starts between stage 10 and 20 (Figure 12 (a)). The algorithm is capable of dealing with this situation by the inclusion of the ratio condition explained in Section 3.1.3.1. With this condition, the algorithm selects stage 11 for the outputs. This could be considered a good choice since s is already low and, in the subsequent stages, no significant decrease of s occurred.

In the second part, we obtain a characteristic profile that becomes almost constant after stage 30, where the lowest ratio is obtained, i.e., the greatest decrease in s relatively to its previous value (see Figure 13).

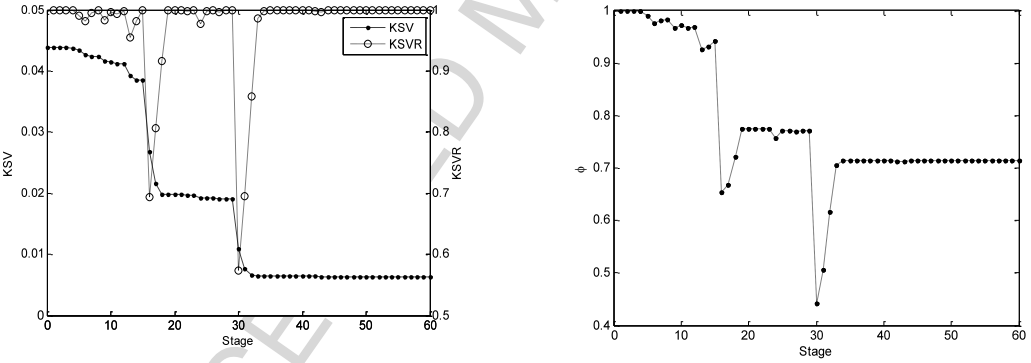
Stage 30 is, in fact, the stage selected by the algorithm, which leads to a lag vector of $\begin{bmatrix}4 & 3 & 4 & 14 & 14 & 2\end{bmatrix}$. This lag vector is quite similar to the theoretical one, $\begin{bmatrix}3 & 3 & 4 & 14 & 14 & 5\end{bmatrix}$ (see Table 4).

ACCEPTED MANUSCRIPT



(a) (b)

Figure 12. Graphical representation of (a) the singular values, their ratio and (b) ϕ, during the first part of the proposed method, in the Ogunnaike and Ray system.



(a) (b)

Figure 13. Graphical representation of (a) the singular values, their ratio and (b) ϕ, during the second part of the proposed method, in the Ogunnaike and Ray system.

3.2 Implementation of the proposed methodologies in Statistical Process Monitoring (SPM) and System Identification (SI) activities

The proposed lag selection method aims to better estimate the real dynamic relationships involving all the system variables, leading to more precise and reliable models. This will have a natural impact in the activities built over DPCA models, such as statistical process monitoring (SPM) and system identification (SI), as analyzed in the following two subsections.

ACCEPTED MANUSCRIPT

3.2.1 Case study: Statistical Process Monitoring

In this section we assess and compare the effect of using the proposed lag selection method in DPCA models, when applied to multivariate statistical process monitoring. For such, several monitoring methodologies were implemented, namely the well known PCA-MSPC procedure [7-10], and two other related procedures, based on DPCA models. One of these methods uses the current lag selection method, proposed by Ku *et al.,* DPCA-LS1, and the other employs our proposed methodology, DPCA-LS2.

The system studied was the Tennessee Eastman benchmark process, developed by Downs and Vogel [40], which has been widely used for comparing process monitoring and control approaches. The simulation model has 41 measurements (XMEAS), 12 manipulated (XMV) variables and allows for the analysis of 21 process upsets; more details are provided elsewhere [19, 40].

In this study we have used the data provided by Braatz *et al.* [41], where the control system implemented is the one after Lyman and Georgakis [42]. Each data set contains 960 observations with a sample interval of 3 min. The faults are introduced 8 hours after the initial simulation instant. All the manipulated and measurement variables, except the agitation speed of the reactor's stirrer (which is always constant) were collected, giving a total of 52 variables.

The data set without faults was used to estimate the PCA and DPCA models. The number of principal components was determined by parallel analysis and the number of lags for the DPCA model was first selected with the approach proposed by Ku *et al.* [16]. Using these methods we constructed a PCA model with 17 PCs and a DPCA model with 3 lags and 29 PCs (DPCA-LS1). These estimates are in line with those obtained before by Russell *et al.* [19]. Implementing DPCA-LS2, one would obtain the fine lag structure presented in Table 5, along with a total of 69 PCs.

ACCEPTED MANUSCRIPT

Table 5. Number of lags for each variable obtained with DPCA-LS2, in the Tennessee Eastman process.

XMEAS(1)	17	XMEAS(14)	4	XMEAS(27)	17	XMEAS(40)	12
XMEAS(2)	17	XMEAS(15)	17	XMEAS(28)	13	XMEAS(41)	17
XMEAS(3)	8	XMEAS(16)	12	XMEAS(29)	3	XMV(1)	17
XMEAS(4)	17	XMEAS(17)	17	XMEAS(30)	17	XMV(2)	17
XMEAS(5)	17	XMEAS(18)	17	XMEAS(31)	17	XMV(3)	17
XMEAS(6)	16	XMEAS(19)	17	XMEAS(32)	8	XMV(4)	17
XMEAS(7)	17	XMEAS(20)	17	XMEAS(33)	8	XMV(5)	15
XMEAS(8)	15	XMEAS(21)	17	XMEAS(34)	17	XMV(6)	16
XMEAS(9)	17	XMEAS(22)	17	XMEAS(35)	17	XMV(7)	17
XMEAS(10)	17	XMEAS(23)	17	XMEAS(36)	17	XMV(8)	17
XMEAS(11)	16	XMEAS(24)	17	XMEAS(37)	17	XMV(9)	16
XMEAS(12)	17	XMEAS(25)	17	XMEAS(38)	17	XMV(10)	17
XMEAS(13)	17	XMEAS(26)	17	XMEAS(39)	4	XMV(11)	17

The pair of monitoring statistics (T^2 and Q) for each model (PCA, DPCA-LS1 and DPCA-LS2) were then applied to a second data set representing normal operation conditions, in order to determine their respective control limits. The control limit was set by trial and error, so that all the statistics lead to the same false alarm rate of 1%. The methods where then applied to the battery of 21 data sets with different types of faults. The results obtained, are presented in Table 6 (detection rate, i.e., the number of detections in the faulty regions over the total number of observations in the faulty regions).

From the analysis of Table 6, it is possible to verify that the DPCA-LS2 statistics have the higher fault detection rates for 17 out of 21 faults, and comparable detection rates on the remaining ones. The superiority of our proposed lag selection method is also formally confirmed upon application of a paired t-test (5% significance level).

ACCEPTED MANUSCRIPT

Table 6. Detection rates for each fault. The best performance for each type of fault, is signalled in bold.

Fault	PCA		DPCA-LS1		DPCA-LS2	
	T^2	Q	T^2	Q	T^2	Q
1	0.991	**0.995**	0.990	0.994	0.989	0.993
2	**0.985**	0.984	0.984	0.981	0.981	**0.985**
3	0.036	0.006	0.035	0.010	**0.059**	0.032
4	0.218	0.980	0.165	**0.999**	0.397	**0.999**
5	0.257	0.217	0.293	0.228	0.326	**0.486**
6	0.989	**0.999**	0.989	**0.999**	0.988	**0.999**
7	**0.999**	**0.999**	0.986	**0.999**	0.888	**0.999**
8	**0.974**	0.968	0.973	**0.974**	0.970	0.971
9	0.034	0.010	0.030	0.002	**0.070**	0.017
10	0.367	0.154	0.439	0.172	0.508	**0.531**
11	0.414	0.638	0.340	0.829	0.542	**0.991**
12	0.985	0.925	0.990	0.964	0.994	**0.996**
13	0.943	**0.950**	0.943	**0.950**	0.938	0.948
14	0.988	**0.999**	0.990	**0.999**	0.996	0.998
15	0.035	0.007	0.059	0.009	**0.072**	0.040
16	0.174	0.137	0.217	0.145	0.305	**0.474**
17	0.787	0.905	0.790	0.953	0.954	**0.969**
18	0.893	**0.901**	0.890	0.898	0.894	**0.901**
19	0.115	0.059	0.046	0.298	0.072	**0.956**
20	0.340	0.423	0.408	0.493	0.609	**0.777**
21	0.362	0.414	0.429	0.409	0.444	**0.456**

3.2.2 Case study 2: System identification

Even though it is not its natural application area, DPCA can also be used in the analysis of input/output systems, namely in SI contexts [16, 18, 29]. In this section, we will address this application scenario, mostly to consolidate the results presented in the previous case studies and to illustrate the added-value of properly estimating the dynamic structure of a DPCA model. Our analysis will be based on the evaluation of the one-step-ahead prediction performance of the models derived from the application of the various lag selection methods under consideration. Input/output relationships are extracted from the singular vectors relative to the smallest singular values, as they represent the linear relations present in the extended data covariance matrix.

The process under analysis is the Wood and Berry system described before (see Table 4), from which 5000 samples were generated with a SNR of 10 dB. By application of

ACCEPTED MANUSCRIPT

the method proposed by Ku *et al.* [16] the number of lags estimated is 1. On the other hand, with our proposed method, the estimated lag vector is $[2 \quad 2 \quad 9 \quad 5]$, as referred in section 3.1.3.1. With such lag structures, the extended data matrices for DPCA using both approaches were constructed and their corresponding covariance matrices determined. Then, the singular value decomposition was applied to each covariance matrix, and the singular vectors relative to the two smallest singular values were used to estimate the intrinsic systems models.

The models thus obtained from the application of the two lag selection approaches where then used to provide one-step-ahead predictions, in independent data sets of 5000 samples, repeated 1000 times. The prediction quality was assessed by the Mean Squared Error (MSE). The results are presented in Table 7. For illustration proposes, the MSE for models with 2 and 9 lags are also presented.

Table 7. Mean squared error of one-step-ahead predictions for the Wood and Berry system.

Number of lags	MSE
1	67,8 ± 2,91
2	311,22 ± 80,05
9	1,89 ± 0,89
$[2 \quad 2 \quad 9 \quad 5]$	1,06 ± 0,56

From the results on Table 7, it can be easily concluded that our proposed method led to the lowest MSE, not only comparing with the results obtained with 1 lag (Ku *et al*. method) but also with the 9 lags model (maximum lag number). This indicates that an individual number of lags for each variables is preferable than an overall lag number, since a more reliable model can be obtained.

ACCEPTED MANUSCRIPT

4 Discussion

In the previous section, we have demonstrated the increased accuracy of the proposed methodologies relatively to the benchmark method proposed by Ku *et al* [16], in several testing scenarios, and illustrated their added value in practice in situations where DPCA is employed for addressing SPM and SI problems. The methodologies proposed are theoretical-driven, but incorporate also results and improvements arising from an extensive analysis of possible alternatives to address the lag selection problem. This dose of empiricism is reflected in the adoption of solutions (such as the ratio restriction) that consistently led to better results, providing flexibility and robustness to the proposed methods. Other alternatives, even when grounded on a well established theoretical background, failed to provide better results, or presented implementation problems. For instance, one approach tested to select the maximum number of lags, consisted in finding the point after which the number of new relationships are the expected ones assuming that all the relevant linear relations were extracted until the previous stage. In fact, if one is able to extract all the relationships with a certain number of lags, by adding one more lag than necessary for all variables, one should theoretically obtain all the previous extracted relationships, replicated one more time. And this process will go on, as more lags are added. Therefore, by finding the onset of such regular behaviour, one could establish the maximum number of lags necessary to describe the dynamical behaviour of all the variables. However, such a methodology led to implementation problems that manifest in imprecise estimates of the onset of the replication process, which translate in worse results that those provided by Method I. As an example, consider the Wood and Berry system presented earlier. For this specific system, the correct number of lags is 9, and therefore, it was expected that the number of linear relations (NLR) increased in a proportional way after this lag. However, as can be seen in Figure 14, despite the near linear relation between the NLR and the number of lags, there is no significant difference in the regions before (I) and after (II) lag 9, making it impossible to identify it as the appropriate one. In this sense, the proposed methods proved to be empirically accurate and stable in most of the circumstances studied, providing a usable solution to this non-trivial problem of model structure definition for DPCA.

ACCEPTED MANUSCRIPT

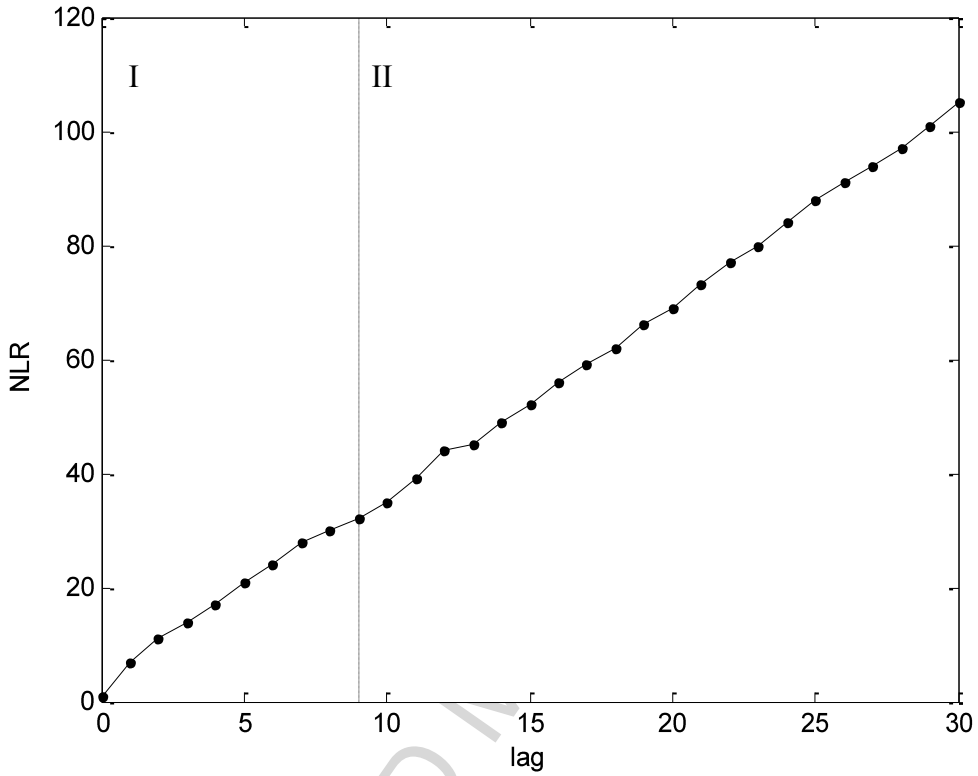


Figure 14. Number of linear relations obtained by parallel analysis in each lag for the Wood and Berry system.

5 Conclusions

In this paper we proposed two new methods for selecting the number of lags in DPCA models. These methods are based on the correspondence between the smallest singular values and near linear relations present on data.

The methods for selecting the maximum number of lags were compared with the procedure proposed by Ku *et al.*[16] and applied to a series of systems with randomly generated structures. From this analysis, we concluded that our proposed method to select the number of lags gave a closer estimation of the lags and was statistically better than the benchmark method. The same conclusion was drawn from the implementation of these methods to several models collected from the literature. We also note that, although the proposed algorithm is capable of selecting the correct stage in most of the times, it is recommend to analyse the graphical representation of the singular values, their ratio and ϕ, and select a stage with simultaneous low values for the singular value

ACCEPTED MANUSCRIPT

and ratio, with no subsequent lower ratio values. This situation should be considered with some attention in the first stages of the procedure, as discussed in the text.

Finally, we also concluded that the use of the proposed methods to select the number of lags, ultimately leads to superior performances in other activities based on DPCA models, such as statistical process monitoring and system identification. Future work will address the application of these methodologies to high-dimensional systems and other application scenarios.

Acknowledgements

Tiago J. Rato acknowledges the Portuguese Foundation for Science and Technology for his PhD grant (grant SFRH/BD/65794/2009). Marco S. Reis also acknowledges financial support through project PTDC/EQU-ESI/108374/2008 co-financed by the Portuguese FCT and European Union's FEDER through "Eixo I do ProgramaOperacionalFactores de Competividade (POFC)" of QREN (with ref. FCOMP-01-0124-FEDER-010397).

References

[1] W.A. Shewhart, Economic Control of Quality of Manufactured Product, Vol. Republished in 1980 as a 50th Anniversary Commemorative Reissue by ASQC Quality Press, D. Van Nostrand Company, Inc., New York, 1931.
[2] J.S. Hunter, Journal of Quality Technology, 18:4 (1986) 203-210.
[3] E.S. Page, Biometrika, 41 (1954) 100-114.
[4] H. Hotelling, The Annals of Mathematical Statistics, 2:3 (1931) 360-378.
[5] C.A. Lowry, W.H. Woodal, C.W. Champ, C.E. Rigdon, Technometrics, 34 (1992) 46-53.
[6] R.B. Crosier, Technometrics, 30:3 (1988) 291-303.
[7] J.E. Jackson, Technometrics, 1:4 (1959) 359-377.
[8] J.E. Jackson, G.S. Mudholkar, Technometrics, 21:3 (1979) 341-349.
[9] J.V. Kresta, J.F. MacGregor, T.E. Marlin, The Canadian Journal of Chemical Engineering, 69 (1991) 35-47.
[10] T. Kourti, J.F. MacGregor, Chemometrics and Intelligent Laboratory Systems, 28 (1995) 3-21.
[11] M.B. Vermaat, R.J.M.M. Does, S. Bisgaard, Quality and Reliability Engineering International, 24 (2008) 573–584.
[12] T.J. Harris, W.H. Ross, The Canadian Journal of Chemical Engineering, 69 (1991) 48-57.

ACCEPTED MANUSCRIPT

[13] D.C. Montgomery, C.M. Mastrangelo, Journal of Quality Technology, 23:3 (1991) 179-193.
[14] B.R. Bakshi, AIChE Journal, 44:7 (1998) 1596-1610.
[15] M.S. Reis, B.R. Bakshi, P.M. Saraiva, AIChE Journal, 54:9 (2008) 2366-2378.
[16] W. Ku, R.H. Storer, C. Georgakis, Chemometrics and Intelligent Laboratory Systems, 30 (1995) 179-196.
[17] C. Lee, S.W. Choi, I.-B. Lee, Chemometrics and Intelligent Laboratory Systems, 70:2 (2004) 165-178.
[18] W. Li, S.J. Qin, Journal of Process Control, 11:6 (2001) 661-678.
[19] E.L. Russell, L.H. Chiang, R.D. Braatz, Chemometrics and Intelligent Laboratory Systems, 51:1 (2000) 81-93.
[20] V. Makis, J. Wu, Y. Gao, European Journal of Operational Research, 174:1 (2006) 112-123.
[21] M.A. Perry, H.P. Wynn, R.A. Bates, Probabilistic Engineering Mechanics, 21:4 (2006) 454-460.
[22] D.R. Brillinger, Royal Statistical Society Conference 1964. Proceedings of the Conference, Cardiff, Vol. City, Year,
[23] G. Elliott, C.W.J. Granger, A. Timmermann, Handbook of Economic Forecasting, Vol. 1, Elsevier, 2006.
[24] J. Breitung, S. Eickmeier, Allgemeines Statistisches Archiv, 90:1 (2006) 27-42.
[25] M.E. Mancino, R. Renò, Applied Mathematical Finance, 12:2 (2005) 187-199.
[26] G.E.P. Box, G.M. Jenkins, G.C. Reinsel, Time Series Analysis - Forecasting and Control, Prentice Hall, Upper Saddle River, NJ, 1994.
[27] L. Ljung, System Identification - Theory for the User, Prentice Hall, Upper Saddle River, NJ, 1999.
[28] F. Tsung, D.W. Apley, IIE Transactions, 34:12 (2002) 1043-1053.
[29] A. Wachs, D.R. Lewin, AIChE Journal, 45:8 (1999) 1688-1700.
[30] M. Guerfel, K.B. Othman, M. Benjeb, 7th IFAC International Symposium on Advanced Control of Chemical Processes 2009. Proceedings of the Conference, Istanbul, Vol. City, Year,
[31] J.E. Jackson, A User's Guide to Principal Components, Wiley, New York, 1991.
[32] I.T. Jolliffe, Principal Component Analysis, Springer, New York, 2002.
[33] R.A. Johnson, D.W. Wichern, Applied Multivariate Statistical Analysis, Prentice Hall, Upper Sadle River, NJ, 2002.
[34] S. Valle, W. Li, S.J. Qin, Industrial & Engineering Chemistry Research, 38 (1999) 4389-4401.
[35] F. Vogt, B. Mizaikoff, Journal of Chemometrics, 17 (2003) 346-357.
[36] S. Wold, Technometrics, 20:4 (1978) 397-405.
[37] F. Pesarin, L. Salmaso, Permutation Tests for Complex Data: Theory, Applications and Software, John Wiley & Sons Ltd., 2010.
[38] W.L. Luyben, Industrial & Engineering Chemistry Process Design and Development, 25:3 (1986) 654-660.
[39] B. Roffel, B. Betlem, Process Dynamics and Control: Modeling for Control and Prediction, John Wiley & Sons Ltd, Padstow, 2006.
[40] J.J. Downs, E.F. Vogel, Computers and Chemical Engineering, 17:3 (1993) 245-255.
[41] R.D. Braatz, Multiscale Systems Research Laboratory. http://brahms.scs.uiuc.edu, 2002.
[42] P.R. Lyman, C. Georgakis, Computers and Chemical Engineering, 19:3 (1995) 321-331.

ACCEPTED MANUSCRIPT

Highlights

- Two new methods for selecting the lag structure and number of principal components for Dynamic Principal Components Analysis (DPCA) models, are proposed.
- Proposed methods are based on the analysis of the successive singular values obtained after the inclusion of lagged variables in the extended matrix of predictors.
- Several examples, involving SISO and MIMO systems, demonstrate the improved accuracy of the proposed methods regarding the current benchmark.
- The methods were also applied to process monitoring and system identification activities, leading to superior performances.

ACCEPTED MANUSCRIPT