Engineering Applications of Artificial Intelligence 71 (2018) 113–124

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

N-dimensional extension of unfold-PCA for granular systems monitoring

Llorenç Burgas *, Joaquim Melendez, Joan Colomer, Joaquim Massana, Carles Pous

University of Girona, Campus Montilivi, P4 Building, Girona, E17071, Catalonia, Spain

ARTICLE INFO

Keywords:
Principal component analysis
Unfold-PCA
MPCA
Building energy monitoring
Data mining
Statistical process monitoring

ABSTRACT

This work is focused on the data based modelling and monitoring of a family of modular systems that have multiple replicated structures with the same nominal variables and show temporal behaviour with certain periodicity. These characteristics are present in many systems in numerous fields such as the construction or energy sector or in industry. The challenge for these systems is to be able to exploit the redundancy in both time and the physical structure.

In this paper the authors present a method for representing such granular systems using N-dimensional data arrays which are then transformed into the suitable 2-dimensional matrices required to perform statistical processing. Here, the focus is on pre-processing data using a non-unique folding–unfolding algorithm in a way that allows for different statistical models to be built in accordance with the monitoring requirements selected. Principal Component Analysis (PCA) is assumed as the underlying principle to carry out the monitoring. Thus, the method extends the Unfold Principal Component Analysis (Unfold-PCA or Multiway PCA), applied to 3D arrays, to deal with N-dimensional matrices. However, this method is general enough to be applied in other multivariate monitoring strategies.

Two of examples in the area of energy efficiency illustrate the application of the method for modelling. Both examples illustrate how when a unique data-set folded and unfolded in different ways, it offers different modelling capabilities. Moreover, one of the examples is extended to exploit real data. In this case, real data collected over a two-year period from a multi-housing social-building located in down town Barcelona (Catalonia) has been used.

1. Introduction

One of main challenges in industry's current transformation to the Industry 4.0 paradigm is to integrate, manage, process and exploit process data to benefit business. While the internet of Things (IoT) paradigm provides the infrastructure required for integration and management, data mining provides the background for processing according to the required exploitation goals. This paper focuses on the goal of such monitoring and assumes that a multivariate data mining technique is used for that purpose. In fact, the paper assumes that Principal Component Analysis (PCA) is the underlying principle to perform the monitoring and it focuses on the problem of organizing data to apply PCA. This method is also general enough to be applied to other multivariate monitoring strategies.

PCA is a well-known multivariate statistical technique which is not only widely used for dimensional reduction, but also for modelling and monitoring continuous processes based on observations provided by sensors (Russell et al., 2000; Edward Jackson and Mudholkar, 1979). PCA helps to control the processes by using the Hotelling's T^2 and SPE indices to provide charts to detect and analyse faults. The isolation of those faults is made with the contribution analysis (Kourti, 2005). However, as many other statistical methodologies, PCA requires a 2D matrix organization of data where columns represent variables and rows observations. Thus, models obtained with this technique gather correlations between the variables according to the observations (conveniently organized into rows) and assume independence between them. In monitoring applications, these observations usually refer to a single time instant (continuous processes). However, variations of PCA for monitoring include extensions for batch process monitoring based on Multiway PCA (MPCA, Nomikos and MacGregor, 1994) and other variants to address real-time (R-PCA, Yu et al., 2017), and outlier detection in an IoT context (Peter He et al., 2017).

The Multiway approach extends the concept of single instant observations to observations that have a temporal extension (typically the duration of the execution of the batch process) and consequently,

* Corresponding author.
E-mail addresses: llorenc.burgas@udg.edu (L. Burgas), joaquim.melendez@udg.edu (J. Melendez), joan.colomer@udg.edu (J. Colomer), joaquim.massana@udg.edu (J. Massana), carles.pous@udg.edu (C. Pous).

https://doi.org/10.1016/j.engappai.2018.02.013
Received 1 February 2017; Received in revised form 2 February 2018; Accepted 19 February 2018

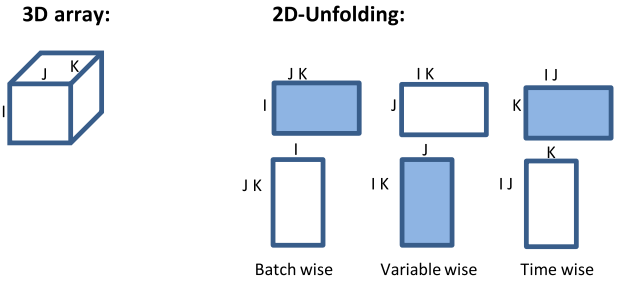

Fig. 1. Graphic representation of all the unfolding possibilities of a 3D matrix.

observations, instead of simple rows, are represented by 2D arrays (variables × samples acquired during the batch execution) and by adding one new dimension to the historic data structure, it now becomes a 3D matrix. Thus, the dimensions of this 3D matrix, containing the historic data of a batch process, are defined by the number of variables being monitored in the process, J, the number of samples acquired at each execution of the batch process, K, and the number of executions included as historic data, I. Again, the I observations represented by these 2D arrays ($I \times K$) containing the data for the monitored variables of a complete execution of a batch process, are assumed to be independent.

Independent of how complex the observations are, the fundamental principles of PCA do not change, but reorganizing (unfolding) the data under study (i.e. to be modelled) into a 2D matrix is required. This implies that, in the case of batch processes, an unfolding preprocessing of the data is required to convert the 3D matrix into a 2D array before applying PCA. This unfolding process is not unique and, depending on how it is done, the interpretations of the results after applying PCA can differ substantially. Thus, there are six known possible combinations to unfold a 3D matrix into a 2D matrix, (see Fig. 1) and not all of them provide interpretable results. (NB: in fact, for the PCA purposes, there are only really three combinations, because half of them are simply the other half transposed.) In batch process monitoring (Nomikos and MacGregor, 1994) variable-wise unfolding ($I \times J \times K \rightarrow I K \times J$, observations in rows are all the samples acquired during the execution of batches) and the batch-wise unfolding ($I \times J \times K \rightarrow K \times IJ$, where observations in rows represent completed batches and number of columns extends to the variables at every time instant, IJ, during the execution of a batch) are commonly used. In other domains, such as monitoring energy in housing buildings for example, time-wise unfolding can also be meaningful (see, for instance, Burgas et al., 2015) to identify singularities in the power consumption of dwellings.

However, there are situations where 3D arrays are not suitable for organizing historic observations and higher dimensional data arrays, or hypercubes, need to be used instead. The need to analyse and model this complex data as a whole, requires developing of a clear methodology to manage the folding/unfolding procedures (as well as other preprocessing measures) for N-dimensional arrays to make them suitable for building interpretable and exploitable PCA models. This occurs, for example, when observations contain not only information from continuous sensors, but also images or spectroscopic information evolving through time where tensor-based dimension reduction techniques are used (Lu et al., 2008; Chen and Shapiro, 2009). A similar situation transpires when considering processes, or systems in a general way, with multiple replicated structures being monitored with the same set of nominal variables (e.g. solar fields and wind farms, injection and assembly lines, cavities in a mould, inkwells in offset industrial printers, power consumers in a grid, or monitoring stores in a mall or rooms in a hotel, etc.). A new challenge appears, one that consists of monitoring not only every subsystem, but also the interactions between them and through time.

Consequently, this requires monitoring tools to be developed that are not only capable of automatically detecting the significantly differently operating elements in any subsystem (e.g. sensor faults, faulty components, performance reduction, misbehaviour detection, etc.) but that also monitor the interactions between these elements and detect any emergent behaviours. By considering modular replication as a new dimension in the data structure this analysis can be carried out, but first requires the adequate pre-treatment and management of the data. Similarly, when an operating continuous system presents a repetitive or periodic behaviour through time, this introduces a degree of redundancy that can be exploited when monitoring. This happens, for instance, in many systems that operate 24/7, but accommodate this operation accordingly due to, for example, shifts, power prices, seasons, solar illumination, etc. Examples of systems with this kind of pseudo-periodic temporal pattern (daily, weekly, seasonally, etc.) are, again, solar fields and wind farms, process industries, or hotels and tertiary buildings affected by daily variations. Such repetitive operations allow models to be built that can then be used as references for monitoring on different time scales or granularity (hourly, daily, weekly, etc.). An example of a multivariate analysis considering this temporal pattern in academic buildings is presented by the authors in Burgas et al. (2014).

Thus, organizing data into multi-dimensional arrays (usually dimension higher than four) is required for data from large systems built on the principle of repetitive modularity and periodic behaviour. This paper aims to provide a method for constructing multivariate models that will monitor such systems as a whole and allow MPCA methodology to deal with N-dimensional arrays. Because the methodology proposed is focused on a previous stage of the PCA modelling itself, then it can be useful not only for PCA modelling and monitoring, but also for other Data Mining tools, such as PLS (Partial least squares). Therefore, this work focuses on the pre-processing stage and, in particular, analysing the significance of the models obtained once specific unfolding strategies have been applied.

This introduction is followed by a background section that includes related work. Following on form that, the methodology to deal with N-dimensional arrays is introduced and the procedure to follow before applying PCA is explained step-by-step. The paper then describes an example of the application and a complete, real exploitation use case is depicted to illustrate the different models that can be obtained from an initial data set and their interpretation and use for monitoring purposes. The paper ends with a section devoted to conclusions and future work.

2. Background and related work

PCA is a method that allows linear dependencies between the variables of a system to be modelled (Russell et al., 2000; Edward Jackson and Mudholkar, 1979). Data gathered during normal operating conditions (NOC) is usually used to obtain a reference model in a new space of lower dimensionality (for instance, waste-water treatment plants as in Aguado and Rosen, 2008). Once the system has been modelled, the new observations projected onto the model's subspace can be used to verify its consistency. Usually two statistics, Hotelling's T^2 and SPE (Square Prediction Error), both defined in the model subspace, are used as the bounds of the model to check if any new observations fall inside or outside the model's thresholds. Hotelling's T^2 indicates how far an observation is from the centre of the model and SPE specifies to what extent the correlations mismatch the ones modelled. Those falling outside the model are considered faulty. Optionally, by using a contribution analysis it is possible to isolate the variables responsible for the deviation outside the statistical thresholds (Kourti, 2005). Currently, there are variations of PCA such as R-PCA (Recursive principal component analysis) in Yu et al. (2017) for sensor outlier detection or monitoring (Peter He et al., 2017) in an IoT scope, that meet the challenges that real-time presents. A complete comparison and study of PCA and its variations can be found in Camacho et al. (2008a, 2008b) and González-Martínez et al. (2014).

However, PCA itself, as with many other data modelling and mining techniques, operates over two-dimensional data matrices organized as $observations(rows) \times variables(columns)$. Some extensions of PCA (for

batch process monitoring for example), known as Multiway PCA (MPCA) described in Nomikos and MacGregor (1994), were defined to deal with three dimensional arrays. Multiway PCA can deal with batch processes (e.g. sequencing batch reactors (SBRs) in waste-water treatment facilities Haimi et al., 2016), thus allowing redundant information stored in the historic data bases of the batch process containing cyclical executions to be exploited. Each complete execution constitutes an observation and supposes adding a new dimension into the input data to be used for modelling and monitoring. Thus, a single observation becomes a 2D matrix containing a set of time series describing the evolution of every variable during the execution of the batch, instead of a single vector containing the samples of variables at a single time instant.

This temporal repetitiveness can be found in other domains. For instance, the power demands of a building present repetitive daily patterns affected by occupancy and weather conditions (Burgas et al., 2014). In Burgas et al. (2015), the same authors extended this approach to deal with multi-entity systems such as buildings (e.g. malls, hotels, housing buildings, offices, etc.) or communities (e.g. neighbourhoods, residential districts, industrial or business parks, etc.), dealing with up to 4D arrays and offering a multi-view monitoring approach for housing buildings when applying different unfolding processes.

However, PCA is not the only methodology available to deal with multivariate data. Other multi-way decomposition approaches that have been conceived for batch processes have their origins in PARAFAC (Harshman, 1970; Chang and Carroll, 1970), Tucker (1966). A survey of previous multi-way decompositions including PARAFAC (or CANDECOMP), Tucker and two-way PCA, is reviewed in Bro (1997). The survey discusses the similarities, constraints and links between them and notes that while a data-set that can be modelled adequately with PARAFAC can also be modelled by Tucker3 or two-way PCA, PARAFAC requires fewer degrees of freedom. On the other hand, Kiers (1991) says that two-way PCA will always fit better than a PARAFAC or Tucker3 model, except in extreme cases where they may all fit equally well. The suitability of the three methods for batch processes is analysed in Westerhuis et al. (1999). None of the studies, however, propose a method to systematically organize and unfold data.

In the following sections the authors formalize and extend the unfolding methodology (Nomikos and MacGregor, 1994) to deal with N-dimensional arrays, taking into account the repeatability and granularity (formal definition in Bettini et al., 1998) of modular systems. Working with folded N-dimensional data-sets allows for all the characteristics of the data to be preserved and for new modelling opportunities to be derived from the redundancy of data.

3. Methodology

3.1. Granular monitoring of multi-entity systems

This work focuses on pre-treating and organizing multidimensional data for monitoring, especially in the case of systems that present repetitive behaviour and/or structures. The method is applied to multi-entity or modular systems (e.g. housing buildings) where every single entity (e.g. a dwelling) is being monitored by the same nominal set of variables (e.g. power consumption, interior temperature, water consumption, occupancy, etc.). To exploit the method's potential, it is expected that there is some kind of interactions between these units (e.g. heat transfer through walls, shared areas, central heating, etc.). The method is general enough to consider multiple levels of modularity in a way that, for a given level, the monitoring variables in a module contain repetitions of those in the level immediately inferior. Thus, in the previous example of a dwelling, this can be defined as the lower level of modularity where five variables are being monitored. A second level could be a floor divided into four dwellings (20 sensors) and a third level could be defined by the whole six-storey building with four dwellings on each floor (i.e. 120 variables in total). Thus, in the initial set of variables, the dimension J is 120 variables long, although this can be split into three levels of modularity, resulting in a 3D array ($J_1 \times J_2 \times J_3$) of $5 \times 4 \times 6$.

The term granular monitoring refers to the possibility of organizing observations on different levels of temporal detail and performing monitoring accordingly. Thus, in batch process monitoring it is easy to distinguish the minimum two levels of granularity (or multi-trajectories), i.e. sampling time and batch (time series acquired during the execution of the batch). Some continuous systems also present this kind of repetitive behaviour. For example, fed-batch reactors or any other calendar operated system that has repetitive behaviour on daily, weekly or yearly time scales. In all of these systems, the sampling time defines the lowest level of granularity and the longest repetition periods define the highest. For a given level, the information contained in a single observation (a granule) does not overlap with any of the other observations on that same level. However, it does, of course, contain multiple observations from an inferior level (for a formal definition of the time granularity concept, the interested reader is referred to Bettini et al., 1998).

Imagine in the previous housing building example, that data sensors gathered data hourly (sampling time) for three years. This will result in a total of $I = 26\,208$ observations (samples acquired every hour). An accurate observation of daily and weekly shapes should show that they present repetitive behaviour that can be analysed on the following four granularity levels: hour, day, week and year. Thus, the initial set of hourly observations ($I = 26\,208$) can now be reorganized into four levels of granularity: I_1, hours a day; I_2, days a week; I_3, weeks a year; I_4, available years of historic data. The initial data-set defined by the 2D matrix ($I \times J$), can in fact be organized into an N dimensional array ($I_1 \times I_2 \times I_3 \times I_4 \times J_1 \times J_2 \times J_3$ with $N = 4 + 3 = 7$), resulting in an array size of $24 \times 7 \times 52 \times 3 \times 5 \times 4 \times 6$.

The next section details the correspondence between elements in both 2D and N-dimensional arrays and shows different ways to unfold this into a new 2D matrix with a different data distribution suitable for monitoring.

3.2. Basic pre-processing operations: folding, standardization, merging and unfolding

Acquisition systems usually gather information sequentially, resulting into long 2D matrices where columns represent every sensor installed and rows contain dated values acquired at every time-stamp. For this work, the initial 2D matrix is called X and is assumed to contain I observations (rows) of J variables (columns). The objective is to transform this matrix into a new 2D matrix, X', with dimensions $J' \times I'$ ($J \neq J'$ and $I \neq I'$), suitable for PCA. This PCA suitable matrix is obtained after reordering observations and variables conveniently to observe the system at the convenient granularity and modularity level defined by the monitoring goals.

Folding is the procedure that will be used to reorganize the data into this N-dimensional *folded* array, $\underline{X}$, by considering system granularity and modularity. Specific dimensions in the N-folded array will correspond to different granularity levels, allowing the data acquired at different sampling times to be merged by simply appending matrices in the correct dimension (same granularity). Additionally, a standardization procedure, one which avoids variables with larger magnitudes and variation range dominating, must be applied to make the data suitable for PCA.

Thus, to perform this transformation of X into X' there are four basic operations to carry out: folding, unfolding, standardization and merging.

1. **Folding**. This is the procedure that allows the original 2D matrix to be transformed into an N-dimensional folded array, $\underline{X}$, in such a way that granularity and modularity are consistently represented.

2. **Standardization**. This is data centring (zero mean) and equalization in terms of variance (unit variance in all the columns). The purpose is to avoid variables with large variances and bias dominating.
3. **Merging** (Optional). This is only required when the original data is split into several arrays with sampling times on different time scales or distinct modularity. It consists of appending two distinct X' matrices (when possible) to add more information to the models at certain levels of modularity/granularity.
4. **Unfolding**. This is the procedure that reshapes the folded $\underline{X}$ array into the best bi-dimensional matrix X', according to the monitoring goals.

These operations are analysed in detail in the following subsections:

3.3. Folding

Folding is the transformation of the original 2D matrix (I observations × J variables) into an N-dimensional folded array, $\underline{X}$, in such a way that granularity and modularity are consistently represented. At this point, that there are other arrays to be merged is not considered (this issue will be discussed further) and it is assumed that the original X matrix contains equally sampled data that has been aligned without blanks.

If the system presents M levels of modularity and L levels of granularity, then it is possible to fold it into an N dimensional, $\underline{X}$ array, with $N = L + M$. Since granularity and modularity are defined in a context of repeatability, the length of the observations and the grouping of the variables at a given level will be fixed and define a dimension of $\underline{X}$. These dimensions are labelled as I_l, with $l = 1 \dots L$ and J_m, with $m = 1 \dots M$, respectively, and the product of their sizes equals the size of the original 2D matrix: $\prod I_l = I$ and $\prod J_m = J$. Observe that if only the lowest levels of granularity and modularity are considered ($L = M = 1$), this will result in the original 2D matrix ($I_1 \times J_1 = I \times J$).

The main problem when performing the folding procedure, is to have control over how the samples are reorganized to facilitate applying the pre-possessing algorithms to the most convenient data organization. To establish a clear correspondence between elements in X and $\underline{X}$ Eqs. (1) and (2) have been established, where any observation in the X matrix ($x_{i,j}$) is mapped to an observation in the folded array ($\underline{x}_{i_1 \dots i_L, j_1 \dots j_M}$), where i_l (with $l = 1 \dots L$) and j_m (with $m = 1 \dots M$) represent the coordinates of the sample in $\underline{X}$.

$$i_l = \left\lfloor \frac{i-1}{\prod_{p=0}^{p=l-1} I_p} \right\rfloor \% \, I_l + 1 \tag{1}$$

$$j_m = \left\lfloor \frac{j-1}{\prod_{p=0}^{p=m-1} J_p} \right\rfloor \% \, J_m + 1. \tag{2}$$

The symbol for the reminder operator of the division performed between the left and right arguments is %, and the square brackets with missing upper bars is the symbol to represent the integer part of the division inside. Where $i = 1 \dots I$ and $j = 1 \dots J$ are the coordinates of the observation ($x_{i,j}$) in the original X matrix; i_l and j_m are the coordinates of the element in the I_l or J_m dimension of the folded array $\underline{X}$; I_p and J_p are the length, or number of elements, in the I_p and J_p dimension of the folded matrix. $I_0 = 1$, $J_0 = 1$ and non-existent dimensions (due to nomenclature when distinct X are folded for merging later) must be considered to be 1.

To exemplify this relationship, suppose an X matrix with $I = 100\,800$ observations and $J = 110$ variables where three levels of granularity are identified in time ($L = 3$) and two levels of modularity in variables ($M = 2$) where $I_1 = 60$ (min), $I_2 = 24$ (h), $I_3 = 70$ (days), $J_1 = 11$ and $J_2 = 10$. The X matrix can be folded into an $\underline{X}$ array with $N = L + M = 3 + 2 = 5$ dimensions. Eqs. (1) and (2) have been used to find the correspondence between any $x_{i,j}$ and the corresponding ($\underline{x}_{i_1,i_2,i_3,j_1,j_2}$) resulting, for this particular case, in the following five corresponding Eqs. (3)–(7).

$$i_1 = \left\lfloor \frac{i-1}{1} \right\rfloor \%60 + 1 \tag{3}$$

$$i_2 = \left\lfloor \frac{i-1}{60 * 1} \right\rfloor \%24 + 1 \tag{4}$$

$$i_3 = \left\lfloor \frac{i-1}{24 * 60 * 1} \right\rfloor \%70 + 1 \tag{5}$$

$$j_1 = \left\lfloor \frac{j-1}{1} \right\rfloor \%11 + 1 \tag{6}$$

$$j_2 = \left\lfloor \frac{j-1}{11 * 1} \right\rfloor \%10 + 1 \tag{7}$$

where i_1, i_2, i_3, j_1 and j_2 are the corresponding indices of the element $x_{i,j}$ in the 5-dimensional matrix $\underline{X}$.

3.4. Unfolding

The unfolding procedure consists of reshaping a folded N-dimensional array, $\underline{X}$, into a bi-dimensional one, X', adequate for PCA modelling purposes. Depending on the unfolding process chosen, distinct X' matrices can be obtained. Observe that for an N-dimensional data matrix, the number of unfolding possibilities doubles according to the following expression:

$$\sum_{k=1}^{k=N-1} \binom{N}{k} = \sum_{k=1}^{k=N-1} \frac{N!}{k!(N-k)!}. \tag{8}$$

Thus, for $N = 3, 4, 5, 6$, and 7, the unfolding possibilities are 6, 14, 30, 62 and 126, respectively. The unfolding possibilities double (plus two) each time N increases a unit. Notice that half of the unfolding possibilities is the transposition of the other half, so the progression is divided by two. However, many combinations appear for large N. For example, Fig. 2 represents the 14 unfolding possibilities for a 4D matrix of lengths I, J, K, L. However, not all these unfoldings make sense in monitoring applications, and so the most appropriate ones must be chosen according to the monitoring goals that have been set. The possible X' matrices that are obtained after unfolding $\underline{X}$ have different correlation structure and consequently the meaning of the PCA analysis changes. Being able to choose the appropriate unfolding process for each monitoring purposes is a critical point. Some indications to help decide which dimensions in $\underline{X}$ will be unfolded as columns or rows are the following:

- The dimension associated to the original variables (corresponding to the lower level of modularity, J_1), should always be placed in the columns' group (always part of J'). Unfolding results where J_1 are considered part of the set of rows to be analysed (part of I') make no sense from a monitoring point of view. Therefore, half of the unfolding possibilities (those with J_1 placed in the rows' group) should be discarded.
- PCA will find linear correlations between the J' variables, explaining the variations in the I' observations. So, dimensions susceptible to holding correlations of interest in our system should be considered for being placed in the variables set (part of J'), i.e. in the classical batch approach, where I, J and K dimensions are defined, only the Batch-Wise $I' \times J' = I \times (JK)$ and Variable-Wise $I' \times J' = (IK) \times J$ unfolding make sense.
- The order (position) of the row and column elements is not relevant in terms of modelling, as the results will be the same, but it is highly recommendable to choose a meaningful organization to easily visualize and understand the model results. This is especially important for the composition of J' dimension (variables).

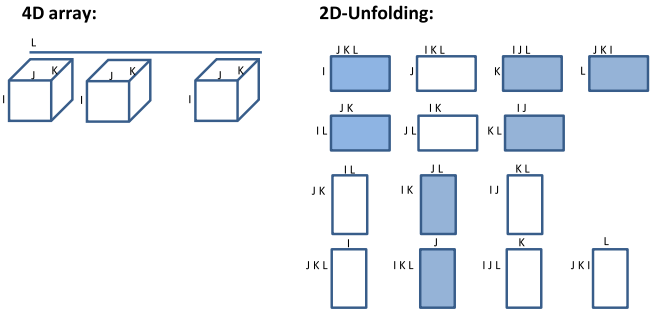

Fig. 2. Graphic representation of all the unfolding possibilities of a 4D matrix.

- Consider re-sampling or data aggregation, when the lower level of granularity is not required, thus reduces the computational cost of creating models and reduces the influence of noise has.
- As PCA studies the correlations between variables, uncorrelated variables can be avoided and computational costs reduced.

Considering this hints the user should be able to choose the best unfolding for his problem and formulating the correspondence between elements in both matrices $\underline{X}$ and X'.

Thus, given an N-dimensional array $\underline{X}$ and a desired unfolding structure $I' \times J'$ ($Rows \times Columns$), the correspondence between an element $(x'_{i',j'})$ in the X' matrix with the corresponding element $(\underline{x}_{i_1,..i_L,j_1...j_M})$ in $\underline{X}$ is given by the mapping Eqs. (9) and (10). Where i'_l (with $l = 1 \dots L'$) and j'_m (with $m = 1 \dots M'$) represent the coordinates of the sample in the N-dimensional folded array $\underline{X}$ reordered according to the final organization of data in rows and columns of X' required. Thus, the notation $(\underline{x}_{i'_1...i'_{L'},j'_1...j'_{M'}})$ represents the same element $(\underline{x}_{i_1,..i_L,j_1...j_M})$ once this reordering of the coordinates has taken place in such a way that the first L' coordinates will be unfolded as rows describing observations and the last M' will be unfolded as columns in the final matrix X'.

$$i' = i'_1 + \sum_{l=2}^{l=L'} \left((i'_l - 1) \prod_{p=1}^{p=l-1} I'_p \right) \tag{9}$$

$$j' = j'_1 + \sum_{m=2}^{m=M'} \left((j'_m - 1) \prod_{p=1}^{p=m-1} J'_p \right) \tag{10}$$

where i'_l (j'_m) is the index of the element in the lth (mth) dimension assigned to the rows' (columns) group, I'_p (J'_m) is the length, or number of elements in that dimension, L' (M') is the number of dimensions in the rows' (columns) group and i' (j') the corresponding index in the final unfolded matrix X'.

To exemplify the correspondence given by the previous equations, suppose that the desired transformation is from a 5-dimensional array $(I_1 \times I_2 \times I_3 \times J_1 \times J_2)$ into a 2D matrix distributed as $(J_2 I_1) \times (J_1 I_2 I_3) = (I'_1 I'_2) \times (J'_1 J'_2 J'_3)$ with sizes $(I_1 = 60, I_2 = 24, I_3 = 70, J_1 = 11$ and $J_2 = 10)$. The correspondence equations that links any element $\underline{x}_{i_1,i_2,i_3,j_1,j_2}$ to the corresponding $x_{i',j'}$ are given by the Eqs. (11) and (12).

$$\begin{aligned} i' &= i'_1 + (i'_2 - 1) * I'_1 \\ i' &= j_2 + (i_1 - 1) * J_2 \\ i' &= j_2 + (i_1 - 1) * 10 \end{aligned} \tag{11}$$

$$\begin{aligned} j' &= j'_1 + (j'_2 - 1) * J'_1 + (j'_3 - 1) * J'_2 * J'_1 \\ j' &= j_1 + (i_2 - 1) * J_1 + (i_3 - 1) * I_2 * J_1 \\ j' &= j_1 + (i_2 - 1) * 11 + (i_3 - 1) * 264. \end{aligned} \tag{12}$$

Thus, a given element in $\underline{X}$, represented by $\underline{x}_{10,20,30,4,5}$ the i' and j' indices of the corresponding $x_{i',j'}$, will be computed with Eqs. (13) and (14).

$$i' = 5 + (10 - 1) * 10 = 95 \tag{13}$$

$$j' = 4 + (20 - 1) * 11 + (30 - 1) * 264 = 7869. \tag{14}$$

Therefore, the element $\underline{x}_{10,20,30,4,5}$ in the 5D array will be reallocated as the element $x'_{95,7869}$ in the unfolded 2D matrix X'.

3.5. Standardization

PCA requires variables being centred and with similar variance. To guarantee this, a standardization procedure should be applied. Standardization will consist of obtaining data with zero mean and unit variance. The procedure is simple: for each variable, its mean (μ) and standard deviation (σ) are obtained, once every sample has been standardized by subtracting μ and dividing by σ, as in expression (15). For the sake of simplicity, x is used for the standardized value and x^o for the original data.

$$x = \frac{x^o - \mu}{\sigma}. \tag{15}$$

In classical 3D unfold-PCA, depending on how μ and σ are obtained (which dimension is considered as the sample), the literature purposes four main standardization procedures known as Continuous Scaling (CS), Auto-Scaling (AS), and Group-Scaling (GS) and Block Scaling. In Continuous Scaling (Esbensen et al., 1987), μ and σ are obtained for each variable during all the time instants (observations). Then, according to the methodology proposed, this is equivalent to performing it at the initial step, that is from X:

$$x_{ij} = \frac{x^o_{ij} - \mu_j}{\sigma_j} \tag{16}$$

with

$$\mu_j = \frac{\sum_{i=1}^{i=I} x^o_{ij}}{I} \tag{17}$$

$$\sigma_j = \sqrt{\frac{\sum_{i=1}^{i=I} (x^o_{ij} - \mu_j)^2}{I - 1}}. \tag{18}$$

When the initial data-set presents distinct granularity (different sampling times, for example) or not all the modules have the same degree of replication (for instance, the existence of common or global variables) the initial data-set must be divided into homogeneous subsets. The resulting subsets have to be able to be represented consistently as the initial matrix X. After performing the previously described folding/unfolding procedures, obtaining a set of matrices X' with the same granularity will then be possible. The following must be considered:

- The same nomenclature must be followed in the unfolded matrices (e.g. if I_1 = seconds in one folding then it cannot be I_1 = hours in an other)
- Unfolded dimensions that result in rows in X' must be consistent in granularity (sampling), units and order.

So, an X' matrix coming from a $(I_1 J_2) \times (AnyGroup)$ unfolding procedure can be added to any X' matrix coming from a $(I_1 J_2) \times (AnyotherGroup)$ unfolding procedure if I_1 represents the same time instants and the same frequency in both matrices and J_2 represents the same variables or entities.

A detailed explanation of the merging procedure can be found in Camacho et al. (2008b). Since the merging is performed with unfolded matrices, it is the same, independently of the dimension of the folded matrix.

In Auto-Scaling (Westerhuis et al., 1999), μ and σ are obtained for each variable at each time instant of the batch (observations are now the time series during the batch). Thus, according to the proposed

methodology, this will be equivalent to performing it after unfolding, in the matrix X':

$$x'_{i'j'} = \frac{x'^{o}_{i'j'} - \mu_{j'}}{\sigma_{j'}} \tag{19}$$

with

$$\mu_{j'} = \frac{\sum_{i'=1}^{i'=I'} x'^{o}_{i'j'}}{I'} \tag{20}$$

$$\sigma_{j'} = \sqrt{\frac{\sum_{i'=1}^{i'=I'} (x'^{o}_{i'j'} - \mu_{j'})^2}{I' - 1}}. \tag{21}$$

Finally, Group Scaling and Block Scaling are used when data consist of several groups or blocks of variables with some given uniform feature (i.e. unit of measure). Different groups have different features. Group and Block Scaling are performed by scaling each group or block by the same standard deviation (i.e. the grand mean of their standard deviations). Following the methodology proposed, an extension of Group or Block Scaling can be defined by allowing for the possibility of obtaining the standard deviation from different unfold matrices (X'') and, once standardized, going back to the initial data format.

3.6. Merging

When the initial data-set presents distinct granularity (different sampling times, for example) or not all the modules have the same degree of replication (for instance, common or global variables exist), it must be divided into homogeneous subsets and the resulting subsets must be able to be represented consistently as the initial matrix X. After performing the folding/unfolding procedures previously described, it will now be possible to obtain a set of matrices X' with the same granularity. The following considerations should be taken into account:

- The same nomenclature must be followed in the unfolded matrices (e.g. if I_1 = seconds in one folding then cannot be I_1 = hours in an other)
- Unfolded dimensions that result rows in X' must be consistent in granularity (sampling), units and order.

So, an X' matrix coming from a $(I_1 J_2) \times (AnyGroup)$ unfolding can be added to any X' matrix coming from a $(I_1 J_2) \times (AnyotherGroup)$ unfolding if I_1 represents the same time instants and the same frequency in both matrices and J_2 represents the same variables or entities.

A detailed explanation about the merging procedure can be found in Camacho et al. (2008b). Since this is performed with unfolded matrices, it is the same, independently of the dimension of the folded matrix.

4. Application example

To illustrate the methodology, a case study monitoring a parabolic trough solar power plant is presented. In this case, the granularity in both the monitored variables and time can be used to reach new modelling options. In the following sections these options are introduced and the proposed methodology is followed.

4.1. Data information

On the one hand, the plant being monitored (Fig. 3) consists of four identical solar fields, each with 50 parallel loops composed of four solar collector assemblies. To generate electricity, the collectors capture the solar radiation by heating a fluid to drive a turbine connected to an electrical generator. In each collector assembly, three variables are measured at the same frequency (transfer fluid temperature, volumetric flow rate, and solar irradiation). Moreover, three production plant variables are provided (power, transfer fluid temperature and volumetric flow rate).

Table 1
Dimensions summary for field, production and meteorological data.

Dimension		Field	Production	Weather
I_1	Hours	24	24	24
I_2	Days	360	360	360
J_1	Variables	3	3	3
J_2	Collectors	4	(none)	(none)
J_3	Parallel loop	50	(none)	(none)
J_4	Solar field	4	(none)	4

On the other hand, as solar plant generation is highly correlated with weather, four weather stations,(one for each solar field), provide three weather variables (temperature, wind, humidity) at an hourly rate.

To summarize, the system has three variables that are replicated at every collector assembly, three global variables from the plant and three global weather variables.

4.2. Data folding

Since three different data sources are available, and to later merge the unfolded matrices, a common nomenclature must be established. For time dimensions, I_1 is used for hours and I_2 for days in the three data-sets. J_1 is always used for the measured variables that are different for each data-set. Then, the J_2 dimension will be used for the collector assemblies of each parallel loop, J_3 for parallel loops of each solar field and J_4 for the solar fields of the power plant. The sizes of each dimension for each data source are indicated in Table 1. Note that the size of J_1 is, by coincidence, the same for the three data sources, but this condition is not really needed to later merge the unfolded matrices.

Eq. (1) has been applied to the three data-sets known as $X1$, $X2$ and $X3$ matrices, to obtain three folded arrays $\underline{X1}$, $\underline{X2}$ and $\underline{X3}$. Thus, $\underline{X1}$ results in a 6D $(I_1 \times I_2 \times J_1 \times J_2 \times J_3 \times J_4)$ array of $24 \times 360 \times 3 \times 4 \times 50 \times 4$ for the collected power plant data, $\underline{X2}$ a 3D $(I_1 \times I_2 \times J_1)$ array of $24 \times 360 \times 3$ for the production data and $\underline{X3}$ a 4D $(I_1 \times I_2 \times J_1 \times J_4)$ array of $24 \times 360 \times 3 \times 4$ for the weather data. According to the proposed methodology, these three N-dimensional arrays are suitable to be unfolded and then used for data-based modelling of the power plant.

4.3. Unfolding

Depending on the objective,the three $\underline{X}$ matrices can be unfolded in several ways following the indications in Section 3.4. Since the high dimensional matrix is $\underline{X1}$, which contains the largest amount of data, this will be used as the basis for the unfolding. Next, $\underline{X2}$ and $\underline{X3}$ will be optionally added by using the merging procedure described in Section 3.6. Considering that $\underline{X1}$ is a 6D matrix, according to Eq. (8) and the associated constraints, there will be up to 31 meaningful unfolding options. To show the value of some of these possibilities, two different modelling objectives are defined: monitoring and benchmarking.

4.3.1. Unfolding for monitoring

The most common modelling objective is to monitor the whole system to detect faults and for diagnostic purposes. This corresponds to a classical data-based monitoring and is achieved by placing I_1 and I_2 in the Rows' group and the rest of dimensions in the Columns' group. In this way, the unfolded $X1'$ $(I_1 I_2) \times (J_1 J_2 J_3 J_4)$ matrix is the same as the original one, $X1$. All the variables measured at each time instant (in this case hourly) are continuously monitored for fault detection and diagnosis tasks.

In addition, daily monitoring can be reached by placing only I_2 in the Rows' group and the rest of dimensions in the Columns' group $(I_2) \times (I_1 J_1 J_2 J_3 J_4)$ for modelling. In this way, when monitoring, all the measurements obtained during a day are used as inputs. This allows, as in batch processes, the repetitiveness (in this case daily) of the data to be considered to perform more accurate fault detection and diagnosis tasks, albeit only once a day. In both monitoring versions, weather and

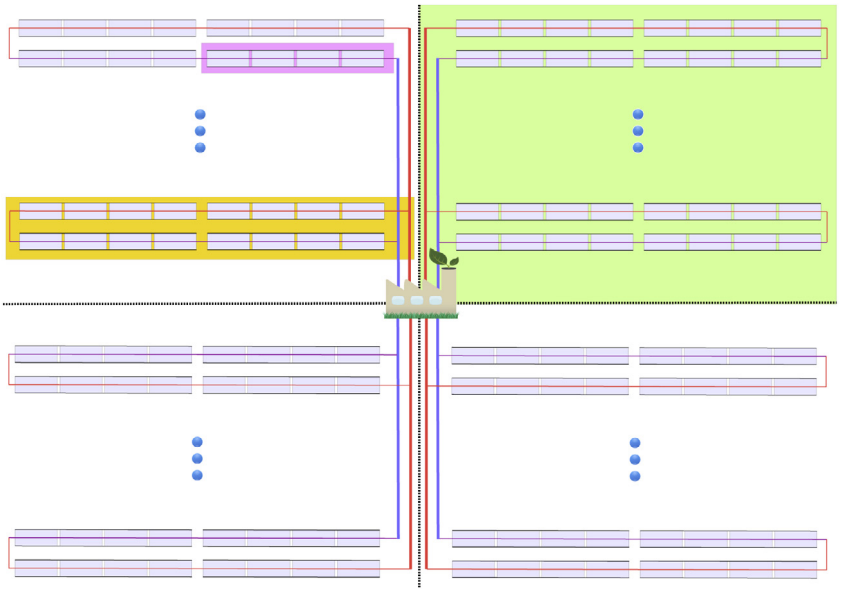

Fig. 3. Schema of monitored parabolic trough solar power plants. One solar field is marked in green, one loop in yellow and one solar collector assembly in pink. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

production variables can be added by using the merging procedure. Weather data can be unfolded $(I_2) \times (I_1 J_1 J_4)$ or $(I_1 I_2) \times (J_1 J_4)$ for daily and hourly monitoring, respectively. In this way, the correlation between the measured process variables and the weather variables is also modelled and then used for monitoring. Likewise, the production data matrix can be unfolded $(I_2) \times (I_1 J_1)$ or $(I_1 I_2) \times (J_1)$ and merged for modelling. This will allow production plant variables to be used for monitoring.

As a numerical example, consider that a daily monitoring of the whole plant, including weather and production data, is going to be carried out. According to Table 1, the historical data matrix used for modelling will be $360 \times (24 * 3 * 4 * 50 * 4 + 24 * 3 + 24 * 3 * 4) = 360 \times 57\,960$. Then, with the monitoring system running on-line, the 57 960 measurements obtained during a day, will be the input to obtain the daily diagnostic of the plant. If the goal is an hourly on-line monitoring using only plant and production data, the matrix with the historical data used for modelling will be $(360 * 24) \times (3 * 4 * 50 * 4 + 3) = 8640 \times 2403$. Then, the 2403 measurements obtained hourly will be the input of the on-line monitoring system.

Moreover, individual monitoring can be done for each specific part of the plant (i.e. each solar field, loop or collector assembly). In this case, models can be built by either taking advantage of the information gathered from the whole plant or from only a specific part. For example, for online daily monitoring of a loop, it is clear that the $3 * 4 * 24 = 288$ measurements obtained each day in the loop should be used. However, at the modelling stage there are several possibilities. Directly, the X' matrix can be built from the dimension $(I_2) \times (I_1 J_1 J_2)$ and the size 360×244 in such a way that only the historical data of that loop is used for modelling. However, other options are to build X' from the dimension $(I_2 J_3) \times (I_1 J_1 J_2)$ and the size $360 * 50 \times 244 = 18\,000 \times 244$, or from $(I_2 J_3 J_4) \times (I_1 J_1 J_2)$ and $360 * 50 * 4 \times 244 = 72\,000 \times 244$, thus obtaining a unique model for all the loops in the same solar field or for the whole plant, respectively. In a similar way, specific models can be built for hourly monitoring, and for collector assembly or solar field monitoring. In all the models proposed in this paragraph, both the weather and the production data-sets can be merged. Then, when monitoring, the daily weather and/or production data should be used, and will likely obtain better results.

4.3.2. Unfolding for benchmarking

In the same way that the granularity of the process behaviour can be useful for monitoring, the modularity of the process structure, based on historical data, could be used for benchmarking. In this case, time dimensions I_1 and I_2 are not in the rows of the unfolded matrix but in the columns. Depending on the dimensions put in the rows of the unfolded matrix, solar fields, loops or collector assemblies can be compared.

In the case of solar fields, X' will have the dimension $(J_4) \times (I_1 I_2 J_1 J_2 J_3)$ and a size of $4 \times (360 * 24 * 3 * 4 * 50) = 4 * 5\,184\,000$, meaning that the 4 solar fields will be compared according to the 5184000 measurements obtained for each one during the year. Since weather data is different for each solar field this can be merged, resulting a merged unfolded matrix of size $4 \times (360 * 24 * 3 * 4 * 50 + 24 * 360 * 3) = 4 * 5\,261\,760$.

In the case of parallel loops, X' will have the dimension $(J_3 J_4) \times (I_1 I_2 J_1 J_2)$ and a size of $(4 * 50) \times (360 * 24 * 3 * 4) = 200 \times 103\,380$. this means that the 200 parallel loops of the plant will be compared depending on the 103 380 measurements collected at each one during the year. Moreover, in this case, the four solar fields can be considered as independent, so four unfolded matrices with the dimension $(J_3) \times (I_1 I_2 J_1 J_2)$ and size $50 \times 103\,380$ can be built to obtain four different models that compare only the parallel loops in each solar field.

In the case of collector assemblies, X' will have the dimension $(J_2 J_3 J_4) \times (I_1 I_2 J_1)$ and a size of $(4 * 50 * 4) \times (360 * 24 * 3) = 800 \times 25\,920$. This means that, the 800 collector assemblies of the plant will be compared depending on the 25 920 measurements obtained at each one during the year. As in the previous case, the four solar fields can be considered as independent, so four unfolded matrices of the dimension $(J_2 J_3) \times (I_1 I_2 J_1)$ and the size $200 \times 25\,920$ can be built to obtain four different models to compare only the collector assemblies of each solar field.

In previous benchmarking examples, the production data cannot be merged since it is common to all the elements compared. This means that, if it were to be used, a number of identical measurements would have to be added at the end of each row. Something which makes no sense for benchmarking tasks. For the same reason, weather data should only be merged in the case of solar fields.

Finally, thanks to the folding and unfolding methodology proposed, some more sophisticated benchmarking possibilities can be analysed for better understand the plant. For example, consider that the structure of the four solar fields is identical and the unfolding is done to obtain an X' of the dimension $(J_3 J_2) \times (J_1 J_4 I_1 I_2)$. In this case, the matrix of the size $(50 * 4) \times (360 * 24 * 3 * 4) = 200 \times 103\,380$ will be useful to analyse the influence the location of the collector assemblies within the solar field

Table 2
Dwelling variables.

Sani (kWh)	Heating energy for Hot water for sanitary use
Heat (kWh)	Heating energy
Cold (kWh)	Cooling energy

Table 3
Production plant generation variables.

Glsa (kWh)	Energy for heating water for sanitary use in dwellings
HRF (kWh)	Energy for radiant floor heating in dwellings
HFan (kWh)	Energy for fan-coils heating in common areas
CRF (kWh)	Energy for radiant floor cooling in dwellings
CFan (kWh)	Energy for fan-coils cooling in common areas
Gas (kWh)	Gas consumption
Ele (kWh)	Electric consumption
Slr (kWh)	Solar generation

has. Similar models can be built for collector assemblies with respect to the parallel loop and/or for parallel loops with respect the solar field.

5. Exploitation example

To better illustrate the benefits of the proposed methodology, another case study using real data from a social building is presented. The building is located in downtown Barcelona (Catalonia) and consists of 32 separated dwellings, common areas and a common generation plant which is used for heating and cooling. Three modelling options derived from the different unfolding strategies from the same initial data-set and defined according to the monitoring goals will be illustrated. PCA has been used as the statistical monitoring strategy following the same principles as in Tucker (1966) (see the reference for further details on applying PCA for multi-housing building monitoring). In the next subsection the data structure will be introduced. Then, following the proposed methodology, several models built from the same initial data will be shown.

5.1. Data information

The social building consists of 32 dwellings. Dwellings are small apartments between 35.58 and 41.24 m^2 each, and each dwelling has its own kitchen, bathroom and one bedroom. Radiant floors heat and cool the apartments and as each dwelling has their own thermostat, the occupants can set the temperature according to their needs. The variables monitored in each dwelling are summarized in Table 2. All the variables being monitored are sampled hourly. The building has a single generation plant that serves the whole building and includes a solar field to generate hot water and three 110 kW Brotje Heizung Ecotherm Plus WGB condensation boilers. The generation plant provides hourly data on consumption and generation. Table 3 summarizes the variables monitored in the production plant.

Weather information during the period is also available through the Catalan public weather agency MeteoCat, and consist of the 11 variables summarized in Table 4. These variables present a sample time of 1 day.

Therefore, the system has three variables that are replicated in every dwelling, eight common energy variables from the generation plant and 11 weather variables.

5.2. Data folding

To be able to later merge the unfolded matrices produced any of the three data sets,a common nomenclature must first be established. The dimension I_1 is used for Hours, I_2 for Days, J_1 for Variables and J_2 for Dwellings. The sizes of each dimension for each data-set are indicated in Table 5.

Table 4
Summary of weather variables.

TM (°C)	Mean daily temperature
TX (°C)	Maximum daily temperature
TN (°C)	Minimum daily temperature
*PPT*24 h (mm)	Daily precipitation
HRM (%)	Mean daily humidity
*RS*24 h (MJ/m^2)	Global irradiation
*VVM*10 (m/s)	Mean daily wind velocity
*DVM*10 (°)	Mean daily wind direction
*VVX*10 (m/s)	Maximum daily wind speed
*DVX*10 (°)	Maximum daily wind speed direction
PM (hPa)	Mean daily atmospheric pressure

Table 5
Summary of dimensions for dwelling, generation and meteorological data.

Dimension		Dwelling	Production	Weather
I_1	Hours	24	24	(none)
I_2	Days	621	621	621
J_1	Variables	3	8	11
J_2	Dwellings	32	(none)	(none)

Finally, Eq. (1) has been applied to the three subsets considered as X matrices, to obtain three folded matrices $\underline{X}$. Thus, $\underline{X}$ results in a 4D ($24 \times 621 \times 3 \times 32$) matrix for dwelling data, a 3D ($24 \times 621 \times 8$) matrix for generation data and a 2D (621×11) matrix for weather data.

The three distinct data sources present different granularity and spatial receptivity. Thus, the first and second present granularity on two levels (hour and day), while the weather data-set only has information on a daily level. Similarly, the dimension corresponding to variables in the first data-set has two levels of modularity (sensors or variables and dwellings), whereas the other two data-sets only have one (variables).

Re-sampling or aggregating variables collected at hourly rates to a daily frequency could produce losses of significant information. Instead of this, by applying the proposed methodology all the data sources are retained and used. They have been folded according to previous structures and adequately unfolded further to be merged when possible.

The main information is provided by the data from Dwellings. This data source is then the basis of the proposed models, and the two data-sets will be used as complementary information sources when needed by applying the merging operation.

5.3. Unfolding

The three $\underline{X}$ matrices can be unfolded in several ways, depending on the monitoring goals and by following the indications in Section 3.4. In this application example, the following objectives were defined:

- Daily monitoring of the whole building
- Identify dwellings that behave similarly
- Daily monitoring of individual dwellings

The following subsections introduce three real use cases, where the corresponding unfolding and merging are described, and some results on using the methodology with these three distinct modelling scenarios are presented.

5.3.1. Unfolding for daily monitoring of the building

In the first use case, the aim is to model the building for supervision purposes to find sensor faults, leakages, poor configurations of the system, etc. This model aims to help the building manager easily obtain information about the building by using simple control charts like dashboard, and performing fault detection daily (I_2). The goal of the model is to explain the differences in the building's daily performance. Consequently, the unfolding is done by placing (I_2) in the Rows' group while I_1, J_1 and J_2 are placed in Columns' group. Thus, an initial model with the dwelling data unfolded as $(I_2) \times (I_1 J_1 J_2)$ is obtained and

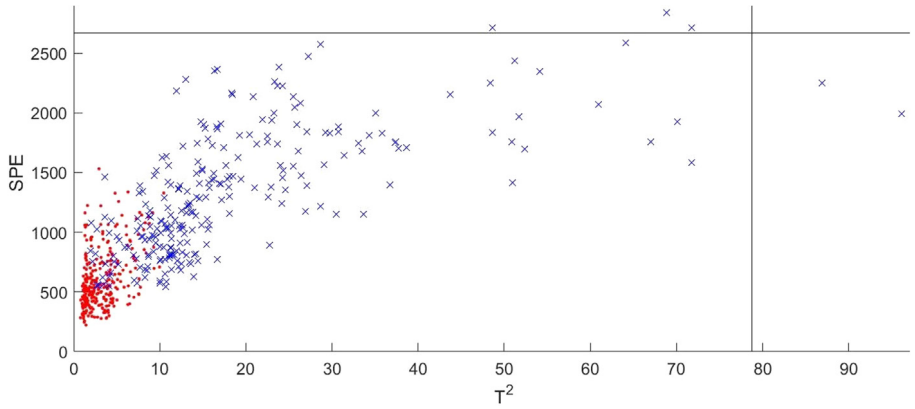

Fig. 4. Hotelling's T^2 index vs. SPE index for daily monitoring using only dwellings data, each red point represents a summer day and each blue cross a winter day.

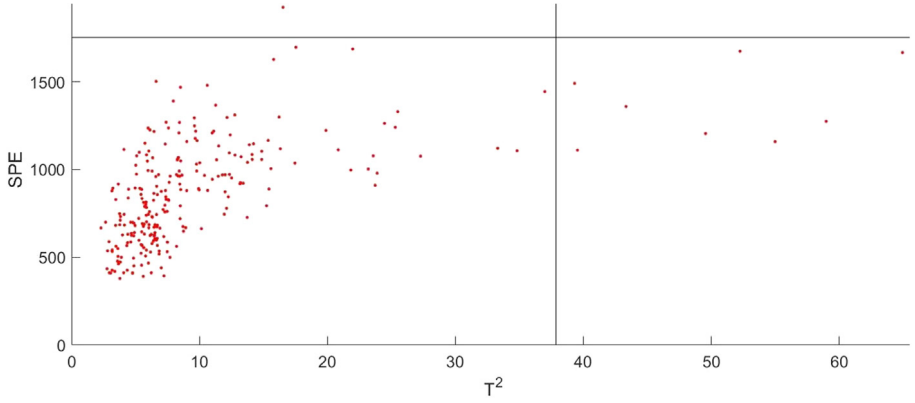

Fig. 5. Hotelling T^2 index vs. SPE index, for daily monitoring using a winter model and only dwellings data, each point represents a day in the system (only dwellings).

studied. Later, production plant data is unfolded as $(I_2) \times (I_1 J_1)$ and merged, and finally weather data is unfolded as $(I_2) \times (J_1)$ and merged to obtain more precise models by advantage of the correlations between all these variables.

5.3.1.1. Modelling only with dwelling data. A first result that can be easily obtained by building the model with the whole data set, is that the winter (blue dots) and summer (red dots) behaviour is totally different (see Fig. 4). This change in the behaviour is obvious, because in winter heating is consumed, whereas in summer this consumption is in cooling. Therefore, winter and summer models must be obtained and analysed separately for more accurate results. In this use case, only the winter models are shown.

Once the winter model has been obtained, classic PCA monitoring charts are then used to detect faulty days (for example, days where the correlations between distinct dwellings change from the ones modelled or days with abnormal magnitudes). Later, when a faulty day is detected, contribution analysis can be used to discriminate the variables causing the fault.

As an example of the monitoring charts provided by PCA, the Hotelling's T^2 vs. SPE graphic is shown in Fig. 5 and Scores in Fig. 6.

Fig. 5 shows some days that surpass the limits. Such days are those that do not follow the normal behaviour modelled by PCA. Generally, days falling over Hotelling's T^2 are magnitude faults and those falling over SPE are correlation faults.

Fig. 6 shows the sore space (grey ellipsoid) and the location of each modelled day (a red point). In the Score space some groups can be found. These groups can usually be associated with distinct consumption patterns. In our case, one group with autumn and spring days can be found (generally located at the positive side of the first score ($T(1)$), grouped and near the centre of the model), whereas winter days are more dispersed.

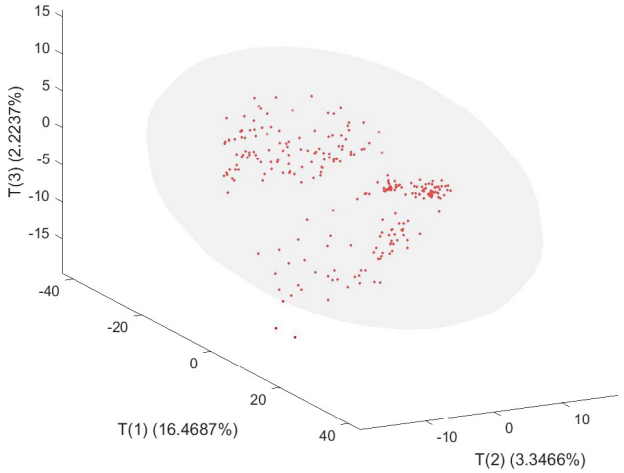

Fig. 6. Three first scores graph marked as T(Number of score)(% of variance), for daily monitoring using a winter model and only dwellings data, each point represents a day in the system (only dwellings).

5.3.1.2. Modelling with dwelling data merged with production plant and weather data. According to the methodology and by merging the data from the production plant using the merging procedure, it is possible to attain the same control charts (Figs. 5 and 6), but these now include

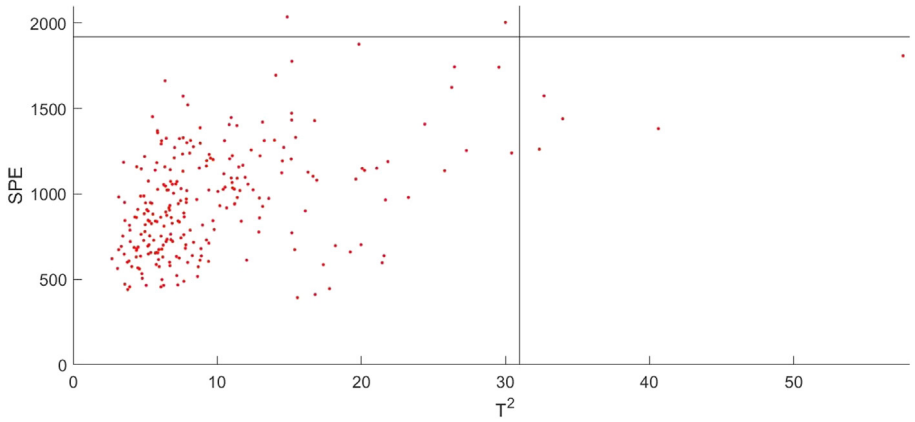

Fig. 7. Hotelling's T^2 index vs. SPE index, for daily monitoring using a winter model and dwellings+production data, each point represents a day in the system (only dwellings and production plant).

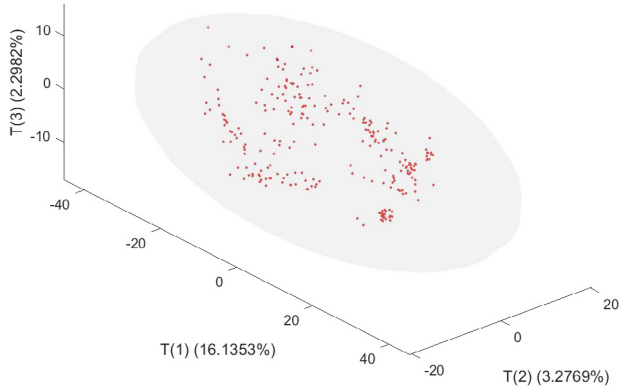

Fig. 8. Three first scores graph marked as T(Number of score)(% of variance), for daily monitoring using a winter model and dwellings+production data, each point represents a day in the system (only dwellings and production plant).

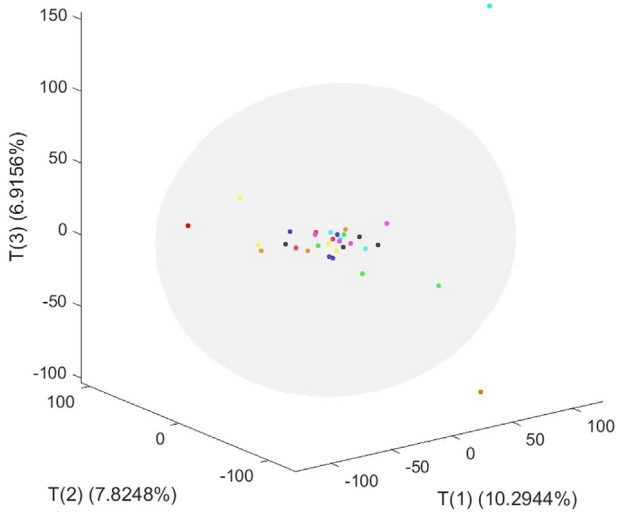

Fig. 9. Three first scores graph marked as T(Number of score)(% of variance), for identify dwellings similarities and using dwellings data, each point represents a dwelling coloured for orientations. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

production plant consumption and generation. The resulting control charts are shown in Fig. 7 for Hotelling's T^2 vs. SPE chart and scores in Fig. 8. The model now includes the data from dwellings and production. By introducing this data, it is now possible to detect poor configurations or errors in the production area.

In Fig. 7, some small changes occur when including the production data, limits are now a little bit lower, some of the previous days near the limits now fall inside the control area while others fall outside. These small changes are due to the information from the production plant calendar and the production settings have been indirectly introduced to the model. Days with errors in the production area or poor configurations are not present during the monitored period, so there are no great changes in the control chart.

On the other hand, in Fig. 8 it is now possible to see distinct groups within the previous autumn and spring group (groups are also located in the positive side of the first score ($T(1)$) axis as in the previous model), these groups are caused by the distinct production configurations.

Finally, weather data is added using the methodology's merge procedure. The model now includes the data from dwellings, production and weather. Consequently, the control charts Hotelling's T^2 vs. SPE and scores will also include the merged data. In this case, weather does not introduce any new detail into the model since the production plant gathers correlated behaviours. However, it can be used to differentiate faults from extreme behaviours caused by weather variations.

5.3.2. Unfolding for identify dwellings similarities

In this second use case, the aim is to benchmark the consumption of the dwellings. Thus, the model does not aim to monitor dwellings on a daily scale, but rather give global information to find the similarities and differences between them. This information can be useful for understanding the system and managing energy more efficiently. Using this model, it is possible to attain information about suspicious or abnormal user behaviours, similarities between users, and also information about the building itself, for instance, finding relationships between the consumption of dwellings that have similar locations (orientation, floor, etc.). Thus, in this case (J_2) is placed in Rows' group and the rest of the dimensions in Columns', resulting in the unfolding structure $(J_2)\times(I_1 I_2 J_1)$. Note that weather and production plant matrices cannot be appended as they do not have the J_2 dimension.

The scores chart obtained from this model is shown in Fig. 9. Apartments on the corners (two external sides) or with poor orientation tend to have behaviours distant from the centre of the model.

In a similar plot, coloured according to floor (Fig. 10), it can be seen how the first floor presents the most distant behaviour to the centre of the model. This is because of the influence of the facilities located on the ground floor. Meanwhile, the second and third floors, except for a few outliers (probably due to the habits of the occupants), present similar and statistically normal behaviours. Finally, the fourth floor also presents a behaviour more distant from the centre of the model. This last

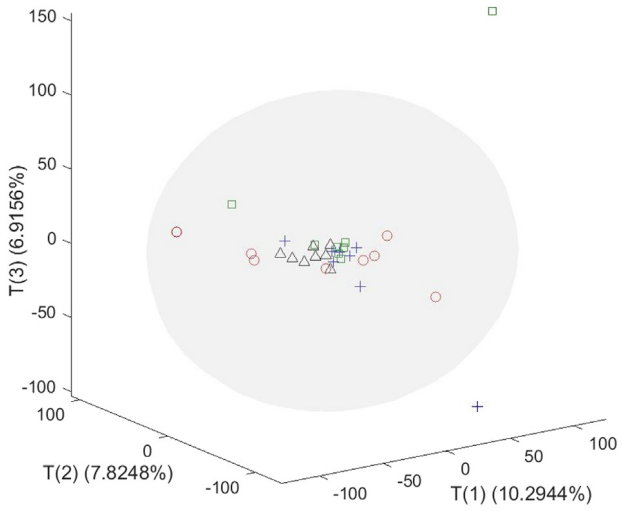

Fig. 10. Three first scores graph marked as T(Number of score)(% of variance), for identify dwellings similarities and using dwellings data, each marker represents a dwelling differentiated by floors. Red circle first floor, blue cross second floor, green square third floor and grey triangle for fourth floor.

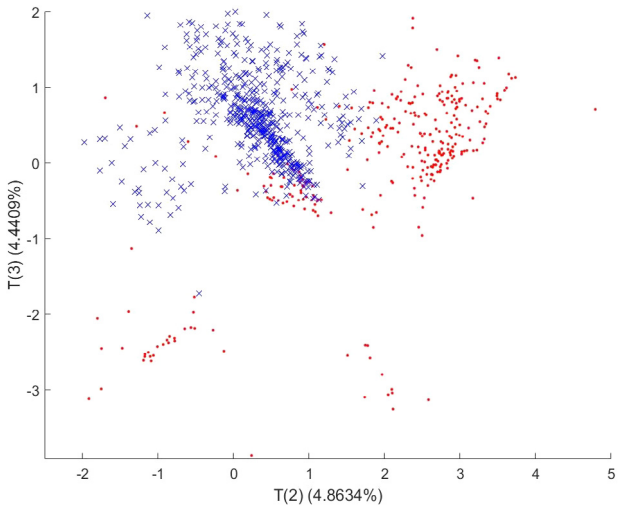

Fig. 11. Two first scores graph marked as T(Number of score)(% of variance), for daily monitoring of each dwelling using a winter model and dwellings data. Each point represents a day in a dwelling. Red dots are from dwelling 1 (first floor corner) and blue crosses are from dwelling 18 (third floor non corner).

behaviour can be explained by the influence the roof isolation has on their consumption.

5.3.3. Unfolding for daily monitoring of dwellings

This third use case, aims to monitor the building, but explains every dwelling separately albeit without losing sight of the whole building. Traditionally, modelling would be done separately for each dwelling, but here the same methodology will be applied without losing the information about the rest of the building. Initial data was divided into distinct X matrices, preserving the singularities described in the previous step (folding). Since the aim is to explain the differences between dwellings, (J_2) is placed in Rows and (I_2) is also placed in Rows to preserve the daily resolution. The others will be reorganized as Columns, thus obtaining $(I_2 J_2)\times(I_1 J_1)$ as the desired unfolding. Note that in this third use case, as in the second one, the unfolding can only be reached using the 4D matrix of the dwellings. Weather and production plant matrices do not have the J_2 dimension so they cannot be merged.

Once unfolded, as in the first example, the model is focused only on winter so cooling production and consumption variables are deleted. Also, the non-winter days are avoided when building the model.

By plotting their scores, this model allows how dwellings behave on a day scale to be compared. See Fig. 11 which shows the behaviour of a first-floor corner apartment (dwelling 1 in red dots) in comparison to a third floor non-corner apartment (dwelling 18 in blue crosses). The corner dwelling presents a larger variability and more outliers than the non-corner one over the winter period observed.

6. Conclusions

In this paper a new methodology to deal with N-dimensional data for monitoring trough PCA models has been presented. First, it is assumed that, from the monitoring point of view, multidimensional is caused by data modularity (repetition of variables) and granularity (periodicity in time). From the point of view of granularity, the method deals with the possibility of organizing detailed observations on different levels and performing monitoring accordingly. In a similar way, the method is general enough to consider multiple levels of modularity in a way that, for a given level, the monitoring variables in a module contain repetitions of those contained in the level immediately inferior. To guide users when choosing their desired unfolding data organization that does not lose any information and respects the original data structure,the methodology provides a step-by-step explanation of the process to be applied before applying PCA. It also includes standardizations and the possibility of merging data-sets with different granularity or modularity.

The application example demonstrates how, by applying the methodology to a single set of data from a parabolic trough solar power plan, many different models can be obtained. These models can have many different purposes, including monitoring or even benchmarking the plant.

The exploitation example, using real data from a social building located in down-town Barcelona (Catalonia), shows the possibilities the proposed methodology has. From same data it is possible to reach distinct unfolding (and then PCA models) that offer different monitoring points of view for the same system (the building). The three different use cases show how different models are obtained and how both classical and new monitoring possibilities are achieved.

Acknowledgements

This work has been carried out by the research group eXIT (http://exit.udg.edu), funded through the following projects: MESC project (Ref. DPI2013-47450-C21-R) and its continuation CROWDSAVING (Ref. TIN2016-79726-C2-2-R), both funded by the Spanish Ministerio de Industria y Competitividad within the Research, Development and Innovation Program oriented towards the Societal Challenges, and also the project Hit2Gap of the Horizon 2020 research and innovation program under grant agreement N680708. The author Llorenç Burgas would also like to thank Girona University for their support through the competitive grant for doctoral formation IFUdG2016.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.engappai.2018.02.013.

References

Aguado, Daniel, Rosen, Christian, 2008. Multivariate statistical monitoring of continuous wastewater treatment plants. Eng. Appl. Artif. Intell. 21 (7), 1080–1091.

Bettini, Claudio, Dyreson, C.E., Evans, W.S., Snodgrass, R.T., Wang, X.S., 1998. A glossary of time granularity concepts. In: Temporal Databases: Research and Practice, Vol. 1399. pp. 406–413.

Bro, Rasmus, 1997. Parafac. tutorial and applications. Chemometr. Intell. Lab. Syst. 38 (2), 149–171.

Burgas, Llorenç, Melendez, Joaquim, Colomer, Joan, 2014. Principal component analysis for monitoring electrical consumption of academic buildings. Energy Procedia 62, 555–564.

Burgas, Llorenç, Melendez, Joaquim, Colomer, Joan, Massana, Joaquim, Pous, Carles, 2015. Multivariate statistical monitoring of buildings. case study: Energy monitoring of a social housing building. Energy Build. 103, 338–351.

Camacho, José, Picó, Jesús, Ferrer, Alberto, 2008a. Bilinear modelling of batch processes. Part I: Theoretical discussion. J. Chemom. 22 (5), 299–308.

Camacho, José, Picó, Jesús, Ferrer, Alberto, 2008b. Bilinear modelling of batch processes. Part II: A comparison of PLS soft-sensors. J. Chemom. 22 (10), 533–547.

Chang, I., Carroll, J.D., 1970. Analysis of individual differences in multidimensional scaling via an n-way generalization of and eckart-young decomposition. Psychometrika (35), 283–319.

Chen, Jiun-Hung, Shapiro, Linda G., 2009. PCA vs. tensor-based dimension reduction methods: An empirical comparison on active shape models of organs. IEEE Eng. Med. Biol. Soc. 2009, 5838–5841.

Edward Jackson, J., Mudholkar, Govind S., 1979. Control procedures for residuals associated with principal component analysis. Technometrics 21 (3), 341–349.

Esbensen, K., Wold, S., Geladi, P.,Öhman, J., 1987. Multi-way principal components-and pls-analysis. Chemometrics 1 (1), 41–56.

González-Martínez, Jose Maria, Camacho, Jose, Ferrer, Alberto, 2014. Bilinear modeling of batch processes. Part III: Parameter stability. J. Chemom. 28 (1), 10–27.

Haimi, Henri, Mulas, Michela, Corona, Francesco, Marsili-Libelli, Stefano, Lindell, Paula, Heinonen, Mari, Vahala, Riku, 2016. Adaptive data-derived anomaly detection in the activated sludge process of a large-scale wastewater treatment plant. Eng. Appl. Artif. Intell. 52, 65–80.

Harshman, Richard A., 1970. Foundations of the PARAFAC procedure: Models and conditions for an explanatory multimodal factor analysis. In: UCLA Working Papers in Phonetics, 16(10): 1–84.

Kiers, Henk A.L., 1991. Hierarchical relations among three-way methods. Psychometrika 56 (3), 449–470.

Kourti, Theodora, 2005. Application of latent variable methods to process control and multivariate statistical process control in industry. Internat. J. Adapt. Control Signal Process. 19 (4), 213–246.

Lu, Haiping, Plataniotis, Konstantinos N., Venetsanopoulos, Anastasios N., 2008. MPCA: Multilinear principal component analysis of tensor objects. IEEE Trans. Neural Netw. 19 (1), 18–39.

Nomikos, P., MacGregor, J.F., 1994. Monitoring batch processes using multiway principal component analysis. AIChE 40 (8), 1361–1375.

Peter He, Q., Wang, Jin, Shah, Devarshi, Vahdat, Nader, 2017. Statistical process monitoring for iot-enabled cybermanufacturing: Opportunities and challenges. IFAC-PapersOnLine 50 (1), 14946–14951. 20th IFAC World Congress.

Russell, E., Chiang, L.H., Braatz, R.D., 2000. Data-Driven Methods for Fault Detection and Diagnosis in Chemical Processes, Vol. 49. Springer, London.

Tucker, LedyardR, 1966. Some mathematical notes on three-mode factor analysis. Psychometrika 31 (3), 279–311.

Westerhuis, Johan A., Kourti, Theodora, MacGregor, John F., 1999. Comparing alternative approaches for multivariate statistical analysis of batch process data. J. Chemom. 13 (3–4), 397–413.

Yu, T., Wang, X., Shami, A., 2017. Recursive principal component analysis based data outlier detection and sensor data aggregation in iot systems. IEEE Internet Things J. PP (99), 1–1.