



**HAL**  
open science

# A nonlinear semantic-preserving projection approach to visualize multivariate periodical time series

Pierre Blanchart, Marine Depecker

► **To cite this version:**

Pierre Blanchart, Marine Depecker. A nonlinear semantic-preserving projection approach to visualize multivariate periodical time series. *IEEE Transactions on Neural Networks and Learning Systems*, 2023, 25 (6), pp.1053 - 1070. 10.1109/TNNLS.2013.2285928 . cea-01828347

**HAL Id: cea-01828347**

**<https://cea.hal.science/cea-01828347>**

Submitted on 23 Jun 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A Nonlinear Semantic-Preserving Projection Approach to Visualize Multivariate Periodical Time Series

Pierre Blanchart and Marine Depecker

**Abstract**—A major drawback of nonlinear dimensionality reduction (DR) techniques is their inability to preserve some authentic information from the source domain, leading to projections that are often hard to interpret when it comes to observing anything other than the topological structure of the data. In this paper, we propose a nonlinear DR approach enforcing projection constraints resulting from an *a priori* knowledge about the structure of the data in multivariate periodical time series. We then propose several ways of exploiting this constrained projection to extract user-relevant information, such as the nominal behavior of a periodical dynamical system or the deviant behaviors which may occur at different time scales. The techniques are demonstrated on both a synthetic dataset composed of simulated multivariate data exhibiting a periodical behavior, and a real dataset corresponding to six months of sensor data acquisitions and recordings inside experimental buildings.<sup>1</sup>

**Index Terms**—Data mining, deviant behaviors identification, high-dimensional, information visualization, monitoring, nonlinear dimensionality reduction (DR), pseudoperiodical time series, visual analytics.

## I. INTRODUCTION

TO our knowledge, there have been few works in the literature combining data mining techniques with information visualization tools for mining multidimensional time series. Although both domains tend toward the same goal, namely extracting meaningful information from the data, and time series have been rarely envisaged from both points of view simultaneously. We argue in this article that it would be highly beneficial to use the power of data mining techniques to achieve convenient representations of time series data, while preserving interpretability and facilitating interactions with the user. The latter are indeed two fundamental aspects to support the exploitation of such representations based on information visualization, in particular when the latter is integrated into a more global processing loop, e.g., such as visual analytics.

Generally speaking, data mining is rather employed in a modeling fashion, where an assumption is made about the

data model/structure, the obtained model/structure being used to extract information from the data. Information visualization envisage the problem of mining data from a more representational point of view, the main goal being to find data representations making sense with respect to the information sought by the user. The main disadvantage of data mining techniques is that they generally do not offer the user the possibility to orient the mining process toward the information he deems relevant, providing instead arbitrary representations of the data, which may or may not contain the looked for information. But, contrary to information visualization methods, they can handle a very large dimensionality and as such, may lead much faster to the result when the dimensionality is high.

This paper is an attempt to combine approaches from both data mining and information visualization communities, which, so far, have remained quite clustered apart in the literature when facing the problem of mining multidimensional time series. The data we are dealing with are multidimensional temporal series exhibiting a pseudoperiodic structure in the signal form. These signals are usually termed as cyclostationary signals or periodically correlated (PC) random processes in the literature [1], [2]. They are encountered very frequently when observing natural phenomena such as tide amplitude, sea temperature, average solar radiation, which tend to vary periodically with respect to time (though the pseudoperiod may be very different depending on the observed phenomenon: for instance, sea temperatures are subject to seasonal variations whereas solar radiations rather exhibit a periodicity of one day). See for instance [3] or [4]. In this paper, we considered multidimensional time series data gathered by a set of sensors placed inside experimental buildings. These data consist of physical measurements of quantities, such as temperature, humidity, and so on, inside different rooms of several buildings. Based on the available information, we aim at developing a user-friendly data mining tool, so as to seek relevant information regarding the insulating properties of those experimental buildings, such as regular trends or deviant behaviors through time. The option considered in this paper aims at facilitating the information mining process by providing adapted information visualizations taking into account the specificities of the data. Hence, we propose a system dedicated to the observation of pseudoperiodic phenomena, with the purpose of quickly identifying nominal/deviant behaviors inside the data through the use of a simple and easy-to-use interactive information visualization interface. To do so, we

The authors are with the CEA, LIST, Gif-sur-Yvette Cedex 91191, France (e-mail: pierre.blanchart@gmail.com; marine.depecker@cea.fr).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TNNLS.2013.2285928

<sup>1</sup>We would like to thank the Institut National de l'Énergie Solaire (INES) and the CEA, LITEN, Laboratoire Énergétique du Bâtiment for providing the data.

introduce a constrained nonlinear multidimensional scaling (CMDS) approach. This novel dimensionality reduction (DR) method exploits the hypothesis of pseudoperiodicity of the data and leads to a trajectory-based representation of the measured information. Indeed in the projection space obtained, the successive pseudoperiods of the signals are represented as piecewise linear curves summarizing the daily behaviors of the different buildings' rooms, termed as (daily-)behaviors or (daily-)profiles in the following part of this article. The experimental data mining process is facilitated by adding simple user controls to the representation, for example allowing the user to easily and quickly make appear nominal/deviant trajectories by restraining the interval of observation inside a pseudoperiod or by choosing to represent only trajectories corresponding to a restrained number of pseudoperiods.

The structure of this paper is as follows: we first propose an overview of existing works related to time series mining from a classical data mining point of view, but also from the perspective of information visualization and visual analytics; then Section III-A introduces a general model of pseudoperiodic time sequence and explains the inspiration behind our CMDS approach starting from this general model. In Section III-B, we derive a formalization of the generic CMDS problem along with an optimization procedure relying on an alternating optimization scheme. The energy management issue originally motivating this paper is presented in Section IV. Based on both synthetic and experimental data, we present a way to postprocess and exploit the 2-D projection provided by the CMDS algorithm by describing potential user-system interactions through the help of a simple graphical interface. Section V provides an assessment of our CMDS approach both from a data mining and an information visualization point of view. Section VI eventually concludes this paper with a discussion and some perspectives to this paper.

## II. BACKGROUND AND RELATED WORKS

In the following, we make a quick tour of the previous work on pseudoperiodical time series and more especially of the most common modelings of this kind of data. We then detail the attempts which have been made from a data mining point of view to extract information from times series, not necessarily pseudoperiodical ones. We do the same from the point of view of information visualization. Finally, we make a quick tour of previous attempts in the visual analytics domain to link DR techniques with information visualization tools.

The periodic autoregressive moving average (PARMA) process models [5]–[9] are the most common modeling approach of PC time series. They start from the definition of PC random processes which are defined as random processes exhibiting a second order periodical stationarity. The PARMA models are mostly used for prediction purposes, like making meteorological predictions, and, as such, are not well suited for our purpose which is to perform monitoring of multidimensional periodical time series. Moreover, PARMA modeling is very heavy computationally speaking, and literally explodes when the dimensionality is high. There are other approaches

which try to cope with this problem of PARMA processes like dynamical principal component analysis [10] or dynamic factor models [11]. These methods extend classical principal component analysis (PCA) and factor analysis for dynamical systems by taking into account the autoregressive nature of the data. But these techniques are mainly oriented toward performing dynamical multivariate statistical process control (for instance disturbance detection), and, as such, do not lead to any practical way of representing the data to perform visual monitoring.

More recently, a lot of work has been done regarding DR, possibly for visualization purposes. Some of these theoretical results have been applied to multidimensional time series by extending classical low-dimensional embedding methods to spatiotemporal data. We can mention for instance the work of Jenkins *et al.* [12] which proposes an extension of ISOMAP [13] for data with both spatial and temporal relationships. In [14], a spatiotemporal extension of Laplacian eigenmaps [15] is proposed with the same purpose of producing a low-dimensional description of data with temporal coherence between data points. In [16], Bishop *et al.* introduce an extension to the generative topographic model (GTM) [17] for time series data. In the original GTM model, the authors model the  $d$ -dimensional data distribution in terms of a few number of latent variables and assume a nonlinear mapping between the latent and data space. To keep the model tractable, the prior distribution of latent variables is constrained to be a discrete distribution defined on a grid, where the node of the grids are termed latent points. DR is performed by computing the parameters of this nonlinear mapping model through the optimization of a likelihood function defined using the underlying probabilistic latent model. The likelihood function is defined by making the assumption that the data are independent and identically distributed (i.i.d.), which is clearly no longer the case when considering time series data. To account for the existing correlations between nearby data points in time, the authors propose to introduce a hidden Markov model where the hidden states are the latent points of the GTM model. Transition probabilities between hidden states are introduced which are used in the global likelihood function to make the inference of the model parameters.

The main issue of the above-mentioned techniques is that they are rather oriented toward obtaining reduced representations of the data while keeping most of the initial information. They have not been designed with the precise goal of representing the data, and, as such, do not necessarily provide the user with meaningful/useful representations for the task at hand.

To cope with this, methods in the field of information visualization have emerged with new ideas about how to perform data mining starting from graphical representations. This community has become quite large over the past few years and numerous works have been issued concerning visual mining and monitoring of time series. In [18], a tool named Viztree is proposed to perform monitoring of streaming telemetry data in the hours preceding the launch of a space vehicle. The system works by encoding the time sequence data into a suffix tree where the frequency of patterns and

other properties are mapped into colors and other visual items. This system may have limited capacities though when using high-dimensional data since it relies on an encoding of the data with a number of symbols which is necessarily limited by the display possibilities of the suffix tree. In addition, the number of symbols in the alphabet is not known and can prove to be a very sensitive parameter regarding the performance of the system. This system moreover does not allow a multiscale analysis since a discretization step is fixed in advance, which will determine once for all the time scale at which the analysis is performed.

In [19], the authors introduce a system named TimeSearcher which is a time series exploration and visualization tool that allows users to retrieve similar patterns inside a single time sequence or across several ones by creating queries. The latter are achieved by using rectangular query locators which allow the user to specify the regions which are of interest to him. TimeSearcher obeys a query-by-example paradigm which implies some preliminary knowledge about what is interesting in the dataset at hand and as such cannot be used as monitoring system. Notwithstanding the fact that multidimensional patterns are most of the time very difficult to identify visually because of overplotting and also because of the limited capacity of the human eye to perform such a task.

Other systems, such as [20] or [21], have been proposed specifically to visualize time series data exhibiting periodical patterns. In [20], data are chunked into time patterns which are then reorganized into clusters. A calendar is then shown in which each time pattern appears with the color of its associated cluster. As previously, this system does not allow for multiscale analysis due to the fact that the size of the time pattern is fixed in advance. High-dimensional data may also enforce the number of clusters to be very high which may render difficult the visual interpretation of the obtained calendar due to the presence of too numerous colors on the representation. The system proposed in [21] provides visualizations of periodical time series data under the form of spiral graphs. In this kind of graph, one cycle of the spiral corresponds to one period of the signal and the thickness of the line used to draw the spiral represents the value of the signal. Though very effective for periodic time signals, this kind of representation is limited to 1-D time series.

More recently, Javed *et al.* [22] have compared different existing line graphs techniques and quantified to which extent these methods allow the user to perform basic visual analysis tasks such as discrimination of patterns between multiple time series. Though these techniques may prove interesting to perform monitoring of multidimensional time series whose dimensions have the same physical meaning (and which can thus be studied from the point of view of multiple time series), they are rather unadapted to our case where the studied time series result from the concatenation of various types of sensor acquisition data.

We conclude this review by mentioning previous work in the field of visual analytics trying to combine data mining and information visualization techniques for time series and other data types. We can first mention the works of Venna *et al.* [23] and Lespinats *et al.* [24]: although unrelated

with time series data, these two articles describe visualization-oriented DR techniques, and, as such, are worth mentioning. In [23], the user can control the mapping through the help of a parameter fixing a tradeoff between precision and recall (the method purpose being to perform information retrieval from the visualization). In [24], the user can control the mapping through the help of a parameter which determines the compromise between local neighborhood preservation and global mapping error, the final purpose being to obtain a representation which makes sense visually.

There also have been recently innovative works which set the accent on the use of DR techniques as a preprocessing step to perform visualization and interactive search of time series. Ward *et al.* [25] propose a high-level comprehensive tool to identify characteristics in univariate time series such as cyclic behavior, outliers, general trends, and so on. The proposed method consists in extracting sub-sequences in the data through the use of a sliding window of size  $N$  and then, mapping each subsequence to a 2-D representation using a classical DR method performed in the  $N$ -dimensional subsequence space. Each subsequence is then represented using a profile or star glyph, suggesting limits on the length of subsequence which can be used in addition to the obvious limit set by the method being only able to handle univariate time series. The authors suggest in the conclusion an extension able to operate on multivariate time series, but it may not be thoroughly adapted to our case of study where we are handling high-dimensional time series exhibiting very long temporal patterns which may not lend themselves to glyph-based representations. In [26], the authors propose a system for the analysis of large multivariate time series data. The data points are projected onto a 2-D space using PCA and then linked together by chronological order to form a path. The time series analysis can then be performed by looking for similar sequences of paths. The user can also rely on several interactive controls such a time slider or a sketch tool to select arbitrary polygonal shapes on the screen, helping him to obtain deeper insights into the data. The system also provides a user-guided aggregation functionality which allows to form data clusters, leaving to the user the choice of the attribute(s) to cluster. The cluster information is shown both on the 2-D data mapping and on cluster glyphs showing the time distribution of the data inside a cluster (to make appear periodical patterns for instance) and the data distribution per dimension (still inside a cluster).

The purpose of this section not being to be exhaustive about visualization techniques for time oriented data, a more complete overview is given in [27]: this book provides a summary of analytical/data mining, interaction, visualization, and display methods, and shows how these aspects can be combined efficiently to perform exploration and analysis of time-oriented data. The authors stress the fact that the previous aspects should not be envisaged separately but instead, should be incorporated and combined into the workflow of real data exploration systems, the purpose being to compensate the weak spots of one method by the utilization of another and also to combine their respective strengths. This is also the claim of the method we develop in this paper, combining analytical/data mining, interactive, and visual aspects.

While containing very interesting general ideas, there remains still that none of the preceding works is particularly dedicated to visual exploration of multivariate (pseudo) periodical time series, even if they propose systems that may be in some cases exploited to do so. Instead, they introduce methods covering a broader range of time series data types and as such, do not implement in the visualization and data mining process the hypotheses relative to pseudoperiodical time series.

In this paper, we address within a single method all the aspects envisaged so far separately in the literature to perform efficient monitoring of high-dimensional pseudoperiodical time series. We focus on classical monitoring tasks such as the identification of nominal or deviant behaviors at different time scales, diagnosis of sensor failure, and classical visualization tasks of multidimensional data. We introduce a generic visualization-oriented DR approach which takes into account the pseudoperiodic structure of the data. Pseudoperiodic time series are projected into a 2-D space in a constrained manner: all the points corresponding to a same recording time index inside a pseudoperiod—i.e., to what we term a point collection in the following—are enforced to be projected onto a line, the lines associated to different point collections being parallel in the projection space. The intuition behind this constraint comes from the pseudoperiodic character of the data, allowing to make the assumption that sensor data recorded at the same period index will lie on a low-dimensional linear manifold. We provide a theoretical justification of this intuition in Section III-A of this paper by using a generic model of multidimensional pseudoperiodical time series, and, we prove that a 1-D linear manifold (i.e., a line) is a justified hypothesis regarding points inside a given collection. Based on a real-life application related to energy management for buildings, we then propose simple ways of interacting with the resulting representation to obtain a system allowing for multiscale monitoring of pseudoperiodical time series. That is, the user can define a customized time interval to monitor inside a pseudoperiod and watch the evolution of the observed phenomenon inside this interval over the different periods. To summarize, our contributions are the following.

- 1) We introduce a CMDS approach for high-dimensional pseudoperiodical time series which preserves the interpretability of the projection axis.
- 2) We propose a set of information visualization tools which allow the user to interact with the representation yielded by the CMDS approach and to perform multiscale monitoring.

### III. DESCRIPTION OF THE CMDS METHOD

#### A. Topological Structure of Periodical Time Series Data

In what follows, we introduce a simple model of periodical time series which allows to account for the specific organization of datapoints after projection. This model is inspired from the structure of pseudoperiodical multidimensional time series real data. Though the introduced model does not belong to any class of known statistical models, the experiments we make on real data in Section V-A4 confirm its appropriateness

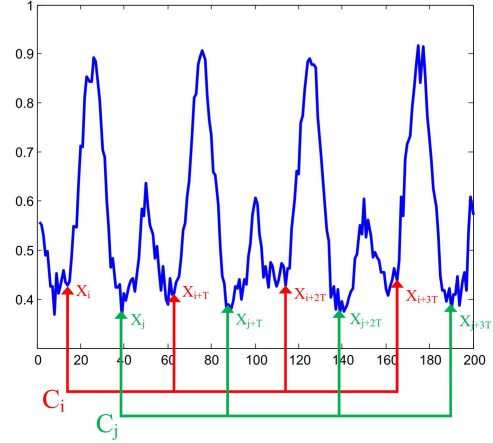


Fig. 1. Illustration of the notion of point collection on a 1-D signal. The green and red rakes show how point collections are built, by grouping together points belonging to identical period locations over several signal periods.

for the problem at hand. We do not use it in a statistical modeling fashion with the purpose of inferring model parameters afterwards, but rather to highlight interesting properties of the signal which will be exploited in the next section to derive our constrained projection approach.

The following notations will be used:  $T$  is the pseudoperiod,  $n$  is the number of observed pseudoperiods, and  $X_t \in \mathcal{M}(1, d)$  is the vector value of the time sequence  $X \in \mathcal{M}(N, d)$  at time  $t$ , i.e.,  $X_t = [x_t^1, x_t^2, \dots, x_t^d]$  where  $d$  is the number of dimensions. Throughout this paper, we will denote as  $N = n \times T$  the total sample size.

We model a periodical time sequence as an amplitude-modulated periodic signal with additive Gaussian noise. Denoting by  $S \in \mathcal{M}(N, 1)$  a 1-D modulating signal and by  $A = [a_1, \dots, a_d] \in \mathcal{M}(1, d)$  a vector of amplitude coefficients associated with each dimension, we can write

$$X = (S \times A) \odot (P \otimes \mathbb{1}_n) + \varepsilon \quad (1)$$

where  $\odot$  is the Hadamard product of matrices (element wise product) and  $\otimes$  the Kronecker product of matrices. The  $T \times d$  matrix  $P$  contains the periodic pattern of each dimension. These patterns are replicated  $n$  times to form the unmodulated periodic signal. The matrix  $\mathbb{1}_n$  is a  $n \times 1$  matrix of 1.

Then, we study the properties of separate ensembles of points, which are formed by grouping together the points with identical (time) locations inside a period, i.e., the collections  $C_i = [X_i^T, X_{i+T}^T, \dots, X_{i+(n-1)T}^T]^T \in \mathcal{M}(n, d)$ , for  $i = 1, \dots, T$ . Fig. 1 gives an illustration on a 1-D signal of how the points inside a collection are extracted.

Using our data model, the collections will be expressed in matrix form as

$$C_i = (S_i \times A) \odot (\mathbb{1}_n \times P_i) + \varepsilon_i$$

where  $P_i$  is the  $i$ th line of  $P$  and  $S_i = [S(i), S(i+T), \dots, S(i+(n-1)T)]^T$ . It is easy to see that the matrix  $M_i = (S_i \times A) \odot (\mathbb{1}_n \times P_i)$  is of rank one, since its generic term is  $m_{kl}^i = a_l P_i(l) S_i(k)$ . Thus, there exists a  $d \times 1$  vector  $\mu_i$  and a  $1 \times n$  vector  $y_i$  such as  $C_i = (\mu_i y_i)^T + \varepsilon_i$ .

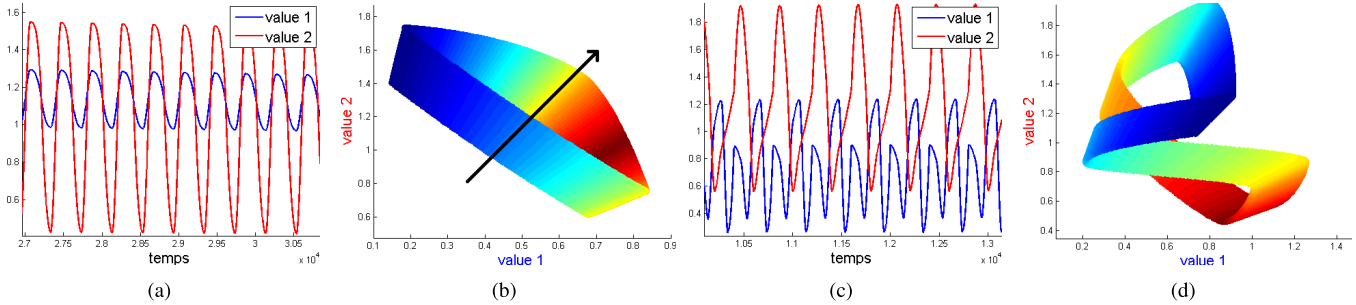


Fig. 2. (a) 2-D pseudoperiodic signal with approximately similar periodic patterns. (b) Representation of the point collections extracted from the 2-D signal of (a). (c) 2-D pseudoperiodic signal with different periodic patterns. (d) Projection of the point collections extracted from the 2-D signal of (c). The vector values of the 2-D signal at each time instant [graphs (a) and (c)] are represented as 2-D points in the diagram of graphs (b) and (d), the value 1 (blue signal) being represented on the  $x$ -axis and the value 2 (red signal) on the  $y$ -axis. In (b) and (d), one color corresponds to one point collection. The vector being situated on 1-D linear manifolds which orientations are close to the orientation of the amplitude vector (dark arrow). The color map is chosen so that consecutive point collections will have similar colors. We thus observe that consecutive collections lead to projections which have similar angular orientation and which are close in the projection space.

Under this model,  $\mu_i$  is then the principal direction of the ensemble of  $d$ -dimensional points  $C_i$ .

Thus, a natural idea would be to perform separate projections of the point collections  $C_i$  and then to perform alignment of these independent projections to obtain a global representation. This is the idea adopted in the local tangent space alignment (LTSA) model introduced in [28]. The main problem of this method with respect to our goal of preserving interpretability, is that it does not necessarily ends up with a relative positioning of the point collections which makes sense visually. It indeed embeds local projections into a global coordinate system by minimizing a global quadratic constraint on the projection error to retrieve the global coordinates from the local ones. Although it preserves the local geometry inside point collections, it leads to arbitrary rotations of the initial groupings with no immediate visual meaning associated with such transforms.

In our method, we propose to enforce a parallelism constraint on the projections of point collections when computing the global coordinates, allowing that way direct intercollection visual comparisons. This idea is again inspired by the structure of periodical time series data. Indeed, under mild conditions of similarity between different signal dimensions, we can show that the projections of point collections have globally the same direction which can be proved to be that of the vector  $A = [a_1, \dots, a_d]$ . An example is shown in Fig. 2(a) and (b), where we project a 2-D pseudoperiodic signal with a different color for each point collection. The vector  $A$  is plotted as a green arrow and we can clearly see on the graph of Fig. 2(b) that the point collections are globally oriented in the same direction as this vector. This parallelism property is not verified for multivariate signals composed of 1-D signals with very dissimilar periodical patterns. An example is given in Fig. 2(d). But in any case, we keep the very interesting property that consecutive collections yield projections with close angular orientations and also with close locations in the projection space.

Considering these properties of pseudoperiodic multidimensional time series, we propose a projection algorithm which projects point collections onto parallel lines in a 2-D space.

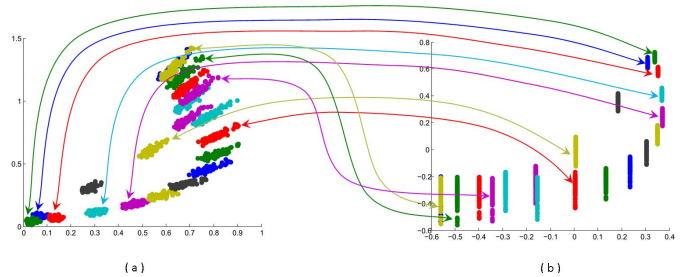


Fig. 3. (b) Constrained projection of the points in (a). The projected points are constrained to lie on parallel lines, each line being representative of one point collection. Corresponding groupings of points appear with the same color in the two diagrams (the colored arrows show some of the correspondences). The points displayed in (a) are obtained from a 2-D synthetic signal such as the one represented in Fig. 2(a).

The local geometry as well as the relative positioning of point collections in the projection space are both preserved using a constrained quadratic criterion, the constraints being that points of a same collection should be lined up and the corresponding lines should be parallel in the 2-D projection space. Fig. 3 illustrates this idea: the left figure represents the point collections obtained from a 2-D synthetic pseudoperiodic signal and the right figure the corresponding constrained projection.

In Section III-B, we successively introduce the functional to optimize in order to obtain the constrained 2-D projection described above and an adaptive step gradient descent optimization algorithm.

### B. CMDS: Formulation and Optimization

First, we briefly review the multidimensional scaling (MDS) formulation which presents many similarities with the constrained projection approach we introduce later in this section. The unconstrained version of our projection algorithm indeed boils down exactly to the principle of multidimensional scaling. It should be noted here that multidimensional scaling refers to a very generic class of DR methods rather than to a specific technique, due to the fact that it starts from the

definition of a Gram matrix containing the scalar products between datapoints. For instance, MDS is strictly equivalent to a linear PCA if the considered scalar product is the standard Euclidean one. The use of kernel scalar products then leads to the definition of a wide range of DR methods. To remove ambiguities, we use the term of kernelized MDS to refer to MDS with a Gram matrix defined from a kernel dot product. The term MDS is indeed sometimes used in the literature to refer to the dual formulation of linear PCA. (Kernelized) MDS thus constitutes a very general framework to introduce both linear and kernel-PCA, as well as all the DR techniques which amount to performing kernel-PCA with a specific Gram matrix such as ISOMAP [13], LLE [29], and Laplacian eigenmaps [15]. The strict equivalence of these methods to a kernelized MDS is explained for instance in [30]. The latter will be exploited in this paper to derive several constrained projection methods by reformulating the above-mentioned classical DR techniques within our CMDS approach. In this sense, the CMDS framework refers to a generic class a DR methods rather than to a specific technique, in the same way as MDS.

1) *Multidimensional Scaling*: The general idea behind (kernelized) multidimensional scaling is to reproduce in a  $r$ -dimensional space an initial configuration of points contained in a  $d$ -dimensional space (with  $d \gg r$ ), while preserving the interpoint metric distances of the original  $d$ -dimensional space. The embedding is performed by preserving the pairwise scalar products rather than the pairwise distances, which is strictly equivalent if we derive the relation between the distance matrix  $D$  and the Gram matrix of scalar products  $K$ . This relation can be shown to be the following:

$$K = -\frac{1}{2} \left( D - \frac{1}{N} D \mathbb{1}_N \mathbb{1}_N^T - \frac{1}{N} \mathbb{1}_N \mathbb{1}_N^T D + \frac{1}{N^2} \mathbb{1}_N \mathbb{1}_N^T D \mathbb{1}_N \mathbb{1}_N^T \right) \quad (2)$$

that is

$$K = -\frac{1}{2} H D H \quad (3)$$

where  $H = \mathbb{I}_N - C_N$ ,  $\mathbb{I}_N$  is the identity matrix of size  $N$  and  $C_N$  is the symmetric centering matrix  $C_N = \frac{1}{N} \mathbb{1}_N \mathbb{1}_N^T$ . This relation is generally known in the literature as double-centering since it amounts to performing a centering on both the lines and the columns of the matrix  $K$ . It also shows that preserving the pairwise scalar products preserves the pairwise distances and vice versa.

Kernelized MDS algorithms perform first a centering in the feature space. Taking the matrix  $K$  as such indeed amounts to considering that the mapped points  $\phi(X_i)$  are centered, which is usually not the case. Thus, we use instead the matrix  $\tilde{K}$ , which contains the pairwise scalar products between the centered  $\phi(X_i)$ , i.e., the matrix

$$\begin{aligned} \tilde{K} &= \langle Y - \mu, Y - \mu \rangle = \langle Y, Y \rangle - \langle \mu, Y \rangle - \langle Y, \mu \rangle + \langle \mu, \mu \rangle \\ &= K - \langle C_N Y, Y \rangle - \langle Y, C_N Y \rangle + \langle C_N Y, C_N Y \rangle \end{aligned} \quad (4)$$

where the rows  $Y_i$  of  $Y$  are the mapped  $X_i$  (i.e.,  $Y_i = \phi(X_i)$ ). The operator  $\langle \cdot, \cdot \rangle$  refers to the pairwise scalar product operator. In the above expression, scalar products between

columns of  $Y$  are Euclidean ones, so we can rewrite this expression in a matrix form as

$$\begin{aligned} \tilde{K} &= K - C_N Y Y^T - Y (C_N Y)^T + (C_N Y) (C_N Y)^T \\ &= K - C_N K - K C_N + C_N K C_N. \end{aligned}$$

The functional minimized by MDS is then

$$E_{\text{MDS}} = \sum_{i,j=1}^n \left\| \tilde{K}_{ij} - U_i U_j^T \right\|_2^2 = \left\| \tilde{K} - U U^T \right\|_F^2 \quad (5)$$

where  $U \in \mathcal{M}(N, d)$ . We thus have  $U U^T = \tilde{K}_r$ , where  $\tilde{K}_r$  is the low-rank approximation of order  $r$  of  $\tilde{K}$ . In the following, we denote by  $\mu_i \in \mathbb{R}$  and  $v_i \in \mathcal{M}(N, 1)$ , respectively, the  $i$ th largest eigenvalue of  $K$  and its associated normalized eigenvector. The elements of  $\sqrt{\mu_i} v_i$  are the coordinates of the embedded points along the  $i$ th coordinate of the projection space. Another interesting property is that  $\tilde{K}_r$  can be decomposed into the sum  $\sum_{i=1}^r \mu_i v_i v_i^T$ . Thus, solving for  $\min_{Z_1, Z_2} \left\| \tilde{K} - (Z_1 Z_1^T + Z_2 Z_2^T) \right\|_F^2$ , where  $Z_1$  and  $Z_2 \in \mathcal{M}(N, 1)$ , yields  $Z_1 = \sqrt{\mu_1} v_1$  and  $Z_2 = \sqrt{\mu_2} v_2$ .

2) *Constrained Multidimensional Scaling*: Constraining the projections of the points belonging to the same collection to lie on a line is equivalent to enforcing the projections of these points to have the same value in  $Z_1$ . By considering a permutation of the time indexes in the time sequence  $X$  such as the elements corresponding to the same collection are grouped together, i.e., the permutation  $X \leftarrow M X$  specified by the  $N \times N$  matrix  $M$  where  $M(i, j) = 1$  iff  $i = (j \bmod T - 1) * n + \lceil \frac{j}{T} \rceil$ , and 0 otherwise, we can enforce the above constraint by rewriting  $Z_1$  as  $M Z_1 = Z = z \otimes \mathbb{1}_n$ . The  $T \times 1$  vector  $z$  contains the first coordinates of the 2-D projections of the point collections in the 2-D constrained projection space, i.e.,  $z(i)$  is the first coordinate of the projections of the points belonging to the collection  $C_i$ .

Then, using the above notations, the constrained optimization problem writes itself down as

$$\min_{z, W} \left\| \tilde{K} - \left( (z \otimes \mathbb{1}_n)(z \otimes \mathbb{1}_n)^T + W W^T \right) \right\|_F^2 \quad (6)$$

where  $z \in \mathcal{M}(T, 1)$  and  $W \in \mathcal{M}(N, 1)$ . In the following, we denote as  $F(z, W)$  the functional to optimize:

$$\begin{aligned} F(z, W) &= \left\| \tilde{K} - \left( (z \otimes \mathbb{1}_n)(z \otimes \mathbb{1}_n)^T + W W^T \right) \right\|_F^2 \\ &= \left\| \tilde{K} - \left( Z Z^T + W W^T \right) \right\|_F^2. \end{aligned} \quad (7)$$

Using a property of the Frobenius norm, we can rewrite the (7) as

$$\text{Tr} \left( \left( \tilde{K} - \left( Z Z^T + W W^T \right) \right)^T \left( \tilde{K} - \left( Z Z^T + W W^T \right) \right) \right)$$

which in turn yields

$$F(z, W) = \text{Tr} \left( \tilde{A}_W \tilde{A}_W^T \right) - 2 \text{Tr} \left( \tilde{A}_W Z Z^T \right) + \left( Z^T Z \right)^2$$

where  $\tilde{A}_W = \tilde{K} - W W^T$ .

The functional  $F$  is not convex with respect to  $z$  and  $W$  taken separately, and, as such, is not convex with respect

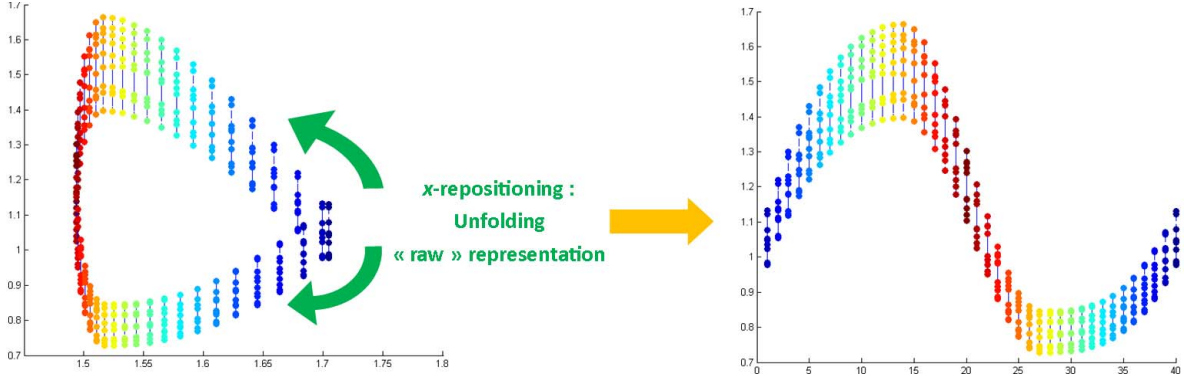


Fig. 4. Left: diagram of the raw representation obtained by projecting data with our constrained projection method. Right: Diagram of the equivalent unfolded representation obtained by performing  $x$ -repositioning of projected point collections. As previously, the different colors indicate the various indexes of the considered point collections. The color map is chosen so that consecutive point collections (time indexes) have similar colors.

to  $[z, W]$ . A sketch of proof is given in Appendix A by considering the Hessian matrix.

To perform the optimization, we use an alternating optimization scheme. Optimizing with respect to  $W$  is easy since it amounts to computing the eigenvector associated with the largest eigenvalue of  $\tilde{A}_Z = K - ZZ^T$ . The most difficult part is to optimize  $F$  with respect to  $z$ . We make use of a gradient descent algorithm with an adaptive step. The first derivative of  $F$  with respect to  $z$  can be shown to be (see details of the computations in Appendix A)

$$\frac{\partial F(\bullet, Z)}{\partial z} = \left[ -2\text{vec}(\tilde{A}_W)^T (Z \otimes \mathbb{I}_N + \mathbb{I}_N \otimes Z) + 4Z^T ZZ^T \right] (\mathbb{I}_T \otimes \mathbb{I}_n). \quad (8)$$

In a practical implementation, (8) can be simplified into

$$\nabla_z F_W = \frac{\partial F(\bullet, W)}{\partial z} = -4Z^T \tilde{B}_Z + 4n^2 z^T z z^T$$

where  $\tilde{B}_Z$  is an  $N \times n$  matrix such as  $\tilde{B}_Z(i, j) = \tilde{V}_Z((j-1) * N * n + i, 1)$  and  $\tilde{V}_Z = \text{vec}(\tilde{A}_Z)$ .

There remains next the problem of initializing the algorithm and of computing the step of the gradient in the gradient descent algorithm.

*a) Initialization:* A natural way of initializing the gradient descent is to compute the couple  $(z, W)$  which yields the closest  $(Z, W)$  (in a quadratic error sense) to the global optimum of the unconstrained optimization problem. We thus take  $W_{\text{init}} = \sqrt{\mu_2} v_2$  where  $v_2$  is the normalized eigenvector associated with the second largest eigenvalue  $\mu_2$  of the matrix  $\tilde{K}$  and  $z_{\text{init}} = \arg \min_z \|z \otimes \mathbb{I}_n - \sqrt{\mu_1} v_1\|^2$  where  $v_1$  is the normalized eigenvector associated with the largest eigenvalue  $\mu_1$  of the matrix  $\tilde{K}$ .

To solve for  $z_{\text{init}}$ , we again consider a gradient descent where the direction of the gradient is given by

$$\frac{\partial f_{\text{init}}}{\partial z} = 2 \left[ Z^T - \sqrt{\mu_1} v_1 \right] (\mathbb{I}_T \otimes \mathbb{I}_n). \quad (9)$$

*b) Adaptive gradient step:* The step of the gradient descent in the main optimization procedure can be made adaptive by computing at each iteration  $t$  the step  $\gamma_t^*$

such as

$$\gamma_t^* = \arg \min_{\gamma} F(z_t - \gamma \nabla_z F_{W_t}, W_t). \quad (10)$$

In [31], a suboptimal solution to (10) is shown to be given by:  $\gamma_t^* = \arg \min \|\Delta_t z - \gamma \Delta_t g\|^2$  where  $\Delta_t z = z_t - z_{t-1}$  and  $\Delta_t g = \nabla_z F_{W_t} - \nabla_z F_{W_{t-1}}$ , which yields

$$\gamma_t^* = \frac{(\Delta_t z)^T \Delta_t z}{(\Delta_t z)^T \Delta_t g}.$$

The whole optimization procedure is summarized in Algorithm 1 presented in Appendix B.

We now have at our disposal a constrained projection enforcing certain properties on the projected data. The following section is dedicated to the manipulation of the obtained 2-D projection and shows how the imposed constraints allow user-oriented information to be very easily extracted from it. In particular, we introduce several controls which allow the user to manipulate the representation, by choosing for instance which period of time to represent, or by isolating specific point collections whose evolution over time is of interest to him.

#### IV. VISUAL INFORMATION EXTRACTION

To provide an insight into the implementation of our approach, we consider the topical issue of energy management within buildings. The CMDS algorithm will be run on both experimental and synthetic data (inspired on the original ones), describing the (mainly thermal) dynamics inside experimental houses. These data consist of temporal series; the postprocessing step further described helps to make their visualization and analysis through the CMDS algorithm easier.

##### A. Postprocessing of the Raw Constrained 2-D Projection

We first transform the raw projection given by our CMDS algorithm by repositioning the projections of the point collections along the  $x$ -axis. The original representation may indeed be quite difficult to interpret visually due to numerous superimpositions. The  $x$ -position of a point collection projection is moreover not very informative since, it mainly betrays the fact that two point collections which correspond to



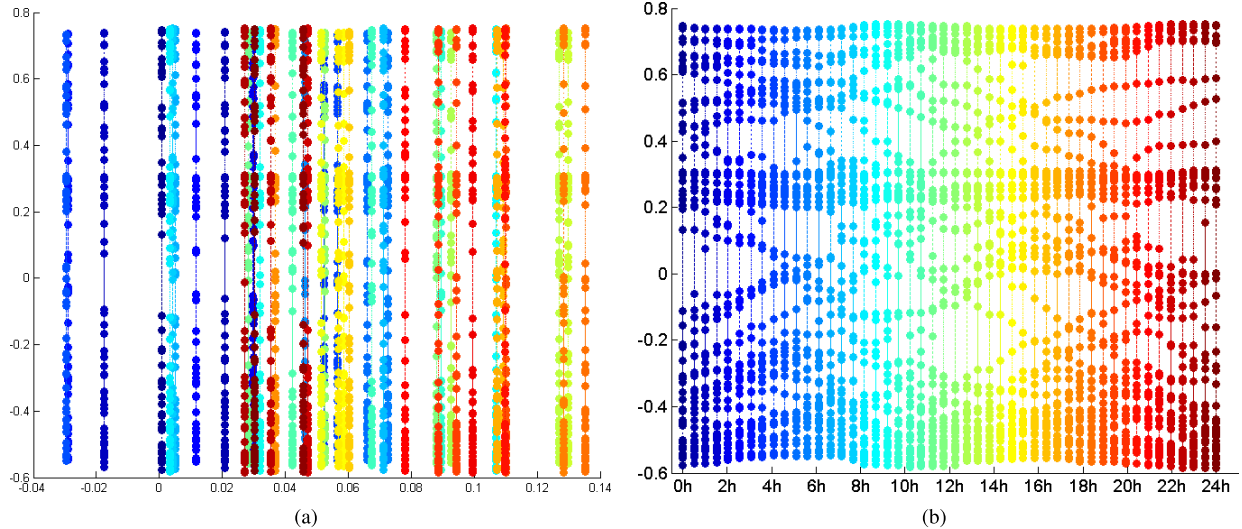


Fig. 5. (b) Graph of the  $x$ -repositioning of the projections of the point collections obtained using our CMDS algorithm. (a) Graph of the corresponding raw output of the CMDS algorithm. The  $x$ -repositioning clarifies the representation by eliminating the superimpositions of point collection projections. Each point collections are displayed in a different color; the color map being chosen so that consecutive point collections (time indexes) have similar colors. Due to the periodic nature of the signal, the point collections corresponding to the end of the pseudoperiod are very similar to the one corresponding to the beginning of the pseudoperiod, creating many superimpositions in the projection space and adding visual confusion in (b) by mixing together visually different colors with the continuous color map used. The vertical axis is a discriminative axis which allows to assess the distance between two trajectories.

two consecutive time indexes will yield projections with very close  $x$ -coordinates. So, we can reposition the projections of the point collections  $C_1, \dots, C_T$  regularly along the  $x$ -axis, and order them according to the index of the corresponding point collection (which indicates the order of appearance in the signal period). An illustration is shown in Fig. 5. This repositioning preserves the important information which resides in the relative positioning of the projections of the point collections along the  $y$ -axis, and, more importantly, in the relative positioning of point projections inside a collection. The reader could argue that the above section concerning the optimization of  $Z$  is useless since we discard the  $Z$ -coordinates obtained by the CMDS algorithm in the representation we present to the user. But imposing from the beginning arbitrary  $x$ -positions (i.e., an arbitrary  $Z$  vector) for the projections of the point collections introduces a great amount of distortion that would introduce a large error in the computation of  $W$  which is the informative vector. Thus, the pertinence of the obtained  $W$  vector depends largely on the correct computation of the  $Z$  vector. Moreover,  $x$ -repositioning is not meaningless as it corresponds to unfolding the cyclical representation of the projected data, the cyclical nature of the projection being the consequence of the pseudoperiodicity of the original data (see Fig. 4). We plainly justify the  $x$ -repositioning in Section IV-C.1 by arguing that the important information which will be used in the information visualization step is the distance between behaviors and that this distance is not impacted by the  $x$ -repositioning whose role is only to make the representation clearer to the user.

Finally, we obtain a 2-D projection where the  $x$ -axis represents the (time) index inside the system pseudoperiod and the  $y$ -axis the variability inside measurements at this index of the pseudoperiod. The  $x$ -axis is discrete with  $T$  possible positions on it. The  $y$ -axis is continuous and a point projection can

take any possible values. This axis does not bear any semantic related to the physical quantities that are measured by sensors. This is a discriminative axis which allows to compare profiles between each other at any point in time or in their globality: it provides the user with the visual hint he needs to isolate nominal or deviant behaviors.

### B. Multiscale Analysis of the System Behavior

Users are generally interested in identifying nominal and deviant behaviors at different time scales. In the following, we define precisely a behavior over an interval of time as the profile/trajectory induced by joining together the projections of points corresponding to consecutive time indexes contained in this time interval and belonging to a same pseudoperiod of the signal. A behavior is thus identified uniquely by a given time interval and the index of a given pseudoperiod. Fig. 6 shows an illustration of this definition on time series data with a pseudoperiod of one day. A behavior here simply represents a sub-interval of time inside a particular day, for instance, we wish to visualize the sub-interval [7H30, 17H50] of day/(period) 16, which is represented by a yellow line in Fig. 6.

The notion of behavior is directly related to the application which will be presented in Section V-B1, whose purpose is to allow the user to isolate behaviors which are of interest to him: if we consider a system exhibiting a pseudoperiod of a day, a typical user need may be for instance to characterize the system nominal behavior inside the time interval 10–11H and to isolate the top 10% behaviors which differ the most from the nominal behavior. The user may also want to display only a reduced interval of the total data acquisition time. To account for these specific user needs, we make available to him a graphical interface and several user controls. A complete overview of the interface is shown on Fig. 12.

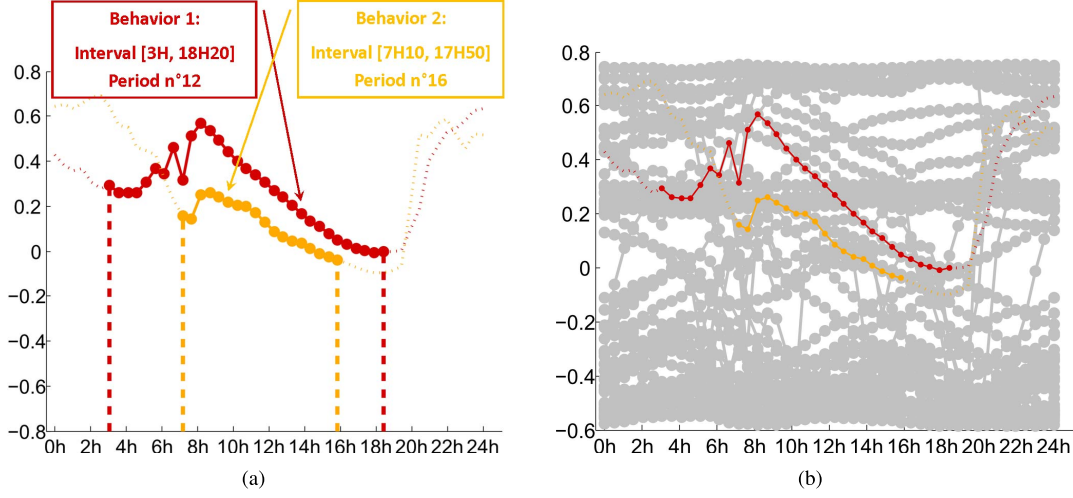


Fig. 6. (a) Two behaviors defined over different sub-intervals of the signal pseudoperiod (which is taken equal to a day in this example). The vertical dotted lines show the limits of the time sub-interval associated with each behavior. Thin dotted lines are used to represent the corresponding complete behaviors defined over the whole pseudoperiod. (b) Graph that replaces the extracted behaviors within the total constrained projection in which all days(/periods) are represented. In this graph, times indexes (points) belonging to a same pseudoperiod (day in this example) are displayed with the same color.

In the following section, we introduce the way to compute the distances between two profiles and the way we can determine the nominal behavior over an interval of time.

### C. Computational Considerations

A behavior/profile is entirely defined by the set of points composing its trajectory. It is possible to consider both the set of points after or before postprocessing of the result of the CMDS algorithm. The tests we made reveal that the  $x$ -repositioning of the projections of the point collections does not impact significantly the computation of the degrees of outlieriness.

1) *Distance Computation*: Graphically, behaviors/profiles can take the form of piecewise linear curves defined over the same domain and at the same positions on the  $x$ -axis. Under these hypotheses, it is natural to compute an interbehavior distance by considering only the  $y$  coordinates of the points defining the curve. A behavior is thus defined by a vector of  $y$  coordinates and the distance between two behaviors can be simply computed as the L1 distance between two vectors, which is equivalent to computing the sum of L1 distances between points with the same  $x$  coordinates. Denoting by  $b_l = \{p_l^1, \dots, p_l^k\}$  a behavior consisting of  $k$  points and by  $\{x_l^j, y_l^j\}$  the coordinates of the point  $p_l^j$ , the distance between two behaviors  $b_i$  and  $b_j$  is defined as

$$d(b_l, b_m) = \sum_{j=1}^k \left| y_l^j - y_m^j \right|. \quad (11)$$

Using this definition of distance between behaviors, we see that the  $x$ -repositioning operation described in Section IV-A does not impact the distance between behaviors, which is an important consideration since the distance between behaviors is the information of interest to make appear nominal or abnormal behaviors in the representation.

2) *Nominal Behavior Computation*: The nominal behavior  $b_{\text{nom}}$  associated with an interval of time is defined as the robust mean of the  $n$  existing behaviors over that interval. We iterate the following two steps until convergence to determine it

$$w_l^t \leftarrow \exp\left(-\frac{1}{2\sigma^2}d\left(b_l, b_{\text{nom}}^{t-1}\right)^2\right)$$

$$b_{\text{nom}}^t \leftarrow \frac{\sum_{l=1}^n w_l^t b_l}{\sum_{l=1}^n w_l^t}$$

using  $b_{\text{nom}}^0 = \frac{1}{n} \sum_{l=1}^n b_l$  to initialize the iterative process. This scheme has the advantage of preserving the computation of the nominal behavior from the influence of deviant behaviors by down-weighting the latter in the averaging process.

3) *Outlieriness Degree Computation*: We then define the degree of outlieriness  $\tilde{o}(b_i)$  of a behavior  $b_i$  as a normalized robust Z-score

$$o(b_i) = \frac{d(b_i, b_{\text{nom}})}{\sigma_o} \quad (12)$$

$$\tilde{o}(b_i) = \frac{o(b_i)}{\max_{j \in \{1, \dots, n\}} o(b_j)} \quad (13)$$

where

$$\sigma_o = \sqrt{\frac{\sum_{i=1}^n w_i d^2(b_i, b_{\text{nom}})}{\sum_{l=1}^n w_l}}$$

The outlieriness degree is normalized in (13) by its maximum value over the considered time interval to obtain a quantity comprised between 0 and 1, 1 being the highest possible degree of outlieriness.

4) *Multimodality*: If the observed period of time is too long, the behavior distribution can be multimodal due, for instance, to seasonal variations if a natural phenomena such as temperatures evolution is observed. In this case, it is not possible to isolate a single nominal behavior. To handle this situation, we suggest that the user restrains the interval of time displayed on the screen until the behavior distribution becomes monomodal.

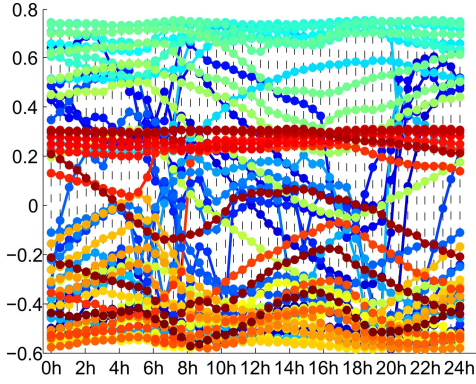


Fig. 7. Result obtained by projecting a three months interval of time with the CMDS algorithm. The different colors attributed to the profiles correspond to the date of acquisition (from dark blue to dark red), so that visual groupings appear among daily behaviors with close acquisition dates.

An example of multimodality is shown in Fig. 7. On the plot of this figure, we can clearly identify several nominal behaviors corresponding each to a specific meteorological regime, and transitory regimes in between corresponding to intervals of time with unstable meteorological conditions. As another example, the plots of Fig. 8 represent isolated monomodal groups of behaviors. This example illustrates the effectiveness of our tool to perform visual monitoring of the deviant behaviors.

## V. RESULTS

This section is divided into two parts. In the first part, we evaluate the performance of our CMDS algorithm, comparing it for instance to standard DR methods. In the second part, we assess the usefulness of our algorithm when integrating it into a complete system for monitoring pseudoperiodical time series. This second part is rather oriented toward testing the information visualization potential of our method through the available user–system interactions described in Section IV-B.

### A. Performance of the CMDS Algorithm

An important advantage of our algorithm is that it is formulated as a constrained kernelized MDS problem, and, as such, it allows the definition of a large variety of constrained nonlinear DR methods. As mentioned in Section III-B, a large number of DR algorithms amounts to performing a kernelized MDS, the main differences between these algorithms residing in the way the Gram matrix containing the kernel dot products is defined. We can thus perform a systematic empirical evaluation by comparing the constrained and the unconstrained versions of several popular DR techniques. In the experiments below, we choose four of the most standard DR techniques: kernel-PCA with RBF kernel, ISOMAP, LLE, and Laplacian eigenmaps. The comparison is based on three criteria commonly used in the literature to assess the performance of DR techniques: embedding error, trustworthiness, and continuity. A detailed explanation of how they are computed is given in the following section.

It should be noted here that the goal of these experiments is not to prove that our method performs better in terms of embedding error, trustworthiness, and continuity compared with the unconstrained versions of the algorithms considered in the tests. These experiments are rather conducted with the goal of demonstrating that our CMDS framework sustains the comparison in a reasonable way with the state-of-the-art methods with respect to these criteria, while possessing a major advantage in term of visual interpretability and in term of exploitation from the point of view of information visualization.

In Section V-A.1–4, we first give a formal definition of the three criteria used to compare the constrained and unconstrained versions of the four state-of-the-art methods used for the tests. We introduce next the kernelized MDS-like formulations of these techniques and we conclude this part by presenting an experimental evaluation on both a synthetic and a real dataset.

#### 1) Evaluation Criteria:

a) *Embedding error*: The embedding error is a manifold learning notion which is generally computed as the squared error between the recovered low-dimensional embedding in the latent space and the nonreduced data transposed into the latent space (this measure is used for instance in [32]). If we consider the example of linear PCA, the latent space is defined by the (normalized) eigenvectors of the covariance matrix of the data. The embedding error will thus be the distortion (in the quadratic error sense) produced by keeping only a subset of principal axes with dominant eigenvalues. The same definition applies for kernelized MDS in the feature space induced by the kernel function and it can be shown that the embedding error is as follows:

$$E = \sqrt{\sum_{i=1}^d \|u_i - u_i^*\|_2^2} = \sqrt{\sum_{i=r+1}^d \|x_i^*\|_2^2} = \sqrt{\sum_{i=r+1}^d \lambda_i}$$

where  $r$  is the reduced dimensionality,  $d$  is the dimension of the original space,  $u_i$  is the recovered  $i$ th coordinate of the low-dimensional embedding in the latent space ( $u_i \in \mathcal{M}(N, 1)$  since there are  $N$  datapoints), and  $u_i^*$  are the coordinates of the nonreduced data in the latent space.

In the CMDS projection, the definition above has to be slightly altered: we take  $u_i^* = \sqrt{\lambda_i^{\text{res}}} \mu_i^{\text{res}}$  where  $\mu_i^{\text{res}}$  and  $\lambda_i^{\text{res}}$  are the eigenvectors and the associated eigenvalues of the matrix  $R = \tilde{K} - ((z \otimes \mathbb{1}_n)(z \otimes \mathbb{1}_n)^T + WW^T)$  which is the Gram matrix of the residual in the latent space after the constrained projection. The distortion can then be expressed in a similar way as above

$$E = \sqrt{\sum_{i=1}^{d-r} \|u_i^*\|_2^2} = \sqrt{\sum_{i=1}^{d-r} \lambda_i^{\text{res}}}$$

with  $r = 2$ .

Hence, the embedding error is rather a global measure assessing the distortion caused by the DR process. On the other hand, trustworthiness and continuity [33], [34] are rather employed to quantify the amount of local distortions, and measure, respectively, the amount of false neighborhoods and tears.

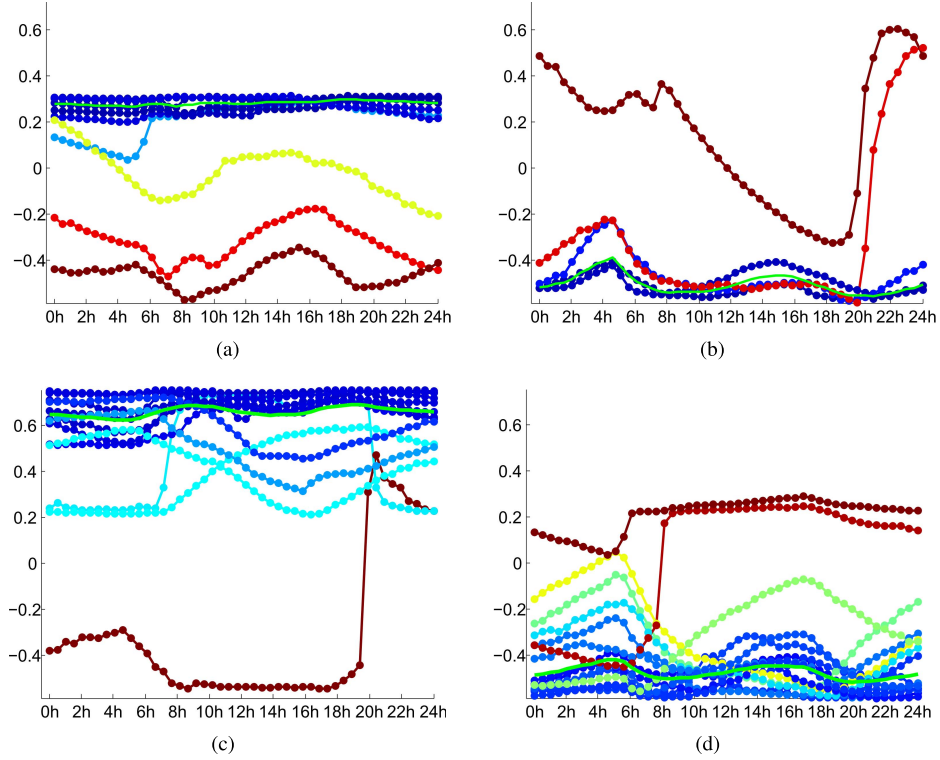


Fig. 8. (a)–(d) Monomodal groups of behaviors extracted from the Fig. 7. Each graph represents a user-restrained interval of time over the total acquisition time, so that the behavior distribution becomes monomodal with a clearly identifiable nominal behavior. The latter is represented on each graph with a green continuous line. The color associated with a behavior represents its degree of outlieriness in the displayed interval of time. The color map ranges from blue (low outlieriness) to dark red (high outlieriness). Note that the colors are not related to the ones used in Fig. 7 and that there are no color-based correspondences between the curves of the four graphs presented in this figure. The color only indicates the distance to the nominal behavior (outlieriness) within a graph and thus, each graph has to be taken independently.

These are very popular measures used in numerous works, including recent ones, to assess the quality of an embedding method, though it may also be interesting to use other criteria such as the ones recently proposed in [35] and [36].

*b) Trustworthiness:* This criteria measures the proportion of points which are close to each other in the low-dimensional embedded space but not in the original high-dimensional space. Thus, the lower the trustworthiness, the better. This quality measure is defined as

$$T(k) = 100 \times \frac{2}{Nk(2N - 3k - 1)} \sum_{i=1}^N \sum_{j \in \mathcal{M}^k(x_i)} (r^*(i, j) - k) \quad (14)$$

where  $k$  is the neighborhood size,  $r^*(i, j)$  is the rank of the  $j$ th point when taking as ranking criterion the distance to the  $i$ th point in the high-dimensional space. The ensemble  $\mathcal{M}^k(x_i)$  is the set of points which are among the  $k$ -NN neighbors of  $x_i$  in the low-dimensional embedded space but not among the  $k$ -NN neighbors of the corresponding point  $x_i^*$  in the high-dimensional space.

*c) Continuity:* In contrast to trustworthiness, continuity measures the proportion of points which are far from each other in the low-dimensional embedded space but not in the original high-dimensional space. Thus, the lower the continuity, the better. This measure is defined in an analogous way

as follows:

$$C(k) = 100 \times \frac{2}{Nk(2N - 3k - 1)} \sum_{i=1}^N \sum_{j \in \mathcal{N}^k(x_i)} (r(i, j) - k) \quad (15)$$

where  $r(i, j)$  is the rank of the  $j$ th point when taking as ranking criterion the distance to the  $i$ th point in the low-dimensional embedded space. The ensemble  $\mathcal{N}^k(x_i)$  is the set of points which are among the  $k$ -NN neighbors of  $x_i^*$  in the high-dimensional original space but not among the  $k$ -NN neighbors of the corresponding point  $x_i$  in the low-dimensional embedded space.

*2) Kernelized MDS-Like Formulations of ISOMAP, LLE, and Laplacian Eigenmaps:* The following properties are given without any justification. Details can be found in [30] or [37].

*a) ISOMAP:* ISOMAP is equivalent to kernelized MDS if  $K_{\text{ISOMAP}} = \mathcal{A}_N^{\mathcal{G}}$  where  $\mathcal{A}_N^{\mathcal{G}} = -\frac{1}{2}H\mathcal{S}^{\mathcal{G}}H$  and  $\mathcal{S}^{\mathcal{G}} = ([d_{ij}^{\mathcal{G}}]^2)$  is the matrix of squared geodesic distances computed on a connected neighborhood graph of the data. The double centering matrix  $H$  is defined by (3).

*b) LLE:* LLE is equivalent to kernelized MDS if  $K_{\text{LLE}} = \lambda_1 \mu_1 \mu_1^T - M$  where  $\lambda_1$  and  $\mu_1$  are, respectively, the largest eigenvalue of  $M$  and its associated normalized eigenvector and  $M = (\mathbb{I}_N - Q)^T(\mathbb{I}_N - Q)$ . The matrix  $Q = (q_{ij})$  explains the local linear relationships between a point and its neighbors

and is computed as

$$Q = \arg \min_Q \sum_{i=1}^N \left\| x_i - \sum_{j \in \mathcal{N}^k(x_i)} q_{ij} x_j \right\|^2. \quad (16)$$

The coefficient  $q_{il}$  is set to 0 if  $x_l$  is not among the  $k$  nearest neighbors of  $x_i$ , i.e., if  $x_l \notin \mathcal{N}^k(x_i)$ , otherwise, it is computed using the above formula.

*c) Laplacian Eigenmaps:* Laplacian eigenmaps are equivalent to kernelized MDS if  $K_{LE} = L^-$  where  $L^-$  is the Moore–Penrose pseudoinverse of the Laplacian matrix  $L = D - Q$ . The matrix  $Q$  is a weighted adjacency matrix whose general term is

$$q_{ij} = \begin{cases} \exp \left\{ -\frac{\|x_i - x_j\|^2}{2\sigma^2} \right\}, & \text{if } x_j \in \mathcal{N}_i \\ 0, & \text{otherwise} \end{cases} \quad (17)$$

and  $D$  is a diagonal matrix whose  $i$ th diagonal element is the sum of the  $i$ th line of  $Q$ .

*3) Experiments on a Synthetic Dataset:* We first illustrate the performance of our CMDS algorithm on a synthetic dataset of multidimensional pseudoperiodic data. Therefore, we generate a 50-dimensional pseudoperiodic signal consisting of  $n = 300$  pseudoperiods of size  $T = 100$ , according to the general data model (1) introduced in Section III-A. The first eight dimensions are represented on the graphs of Fig. 9.

We choose the following parameters: the neighborhood size is set to  $k = n = 300$  in the three methods ISOMAP, LLE, and Laplacian eigenmaps. This is a natural choice since our method relies on the fact that points with the same time index inside a pseudoperiod lie close to a linear manifold of dimension one. In ISOMAP, it means that the pairwise geodesic distances between these points will be exactly the Euclidean distance, and as such, that each point inside this group should be connected to all the others (or in other terms, that there should be a complete interconnectivity between points which possess the same time index inside a given period). This justifies in the case of ISOMAP the choice of  $k = n$  for the connectivity parameter  $k$  in the computation of the geodesic distance matrix. In LLE, the fact that points having the same temporal index inside a pseudoperiod lie on linear manifold of dimension one means that each of these points can be expressed as a linear combination of the others. This justifies in the case of LLE the choice of  $k = n$  for the connectivity parameter  $k$  in the computation of the weight matrix expressing the local linear relationships between points [see (16)]. In Laplacian eigenmaps, the choice of  $k = n$  for the connectivity parameter  $k$  simply expresses the fact that points having the same time index inside a pseudoperiod will have high-similarity degrees in the similarity matrix  $W$  defined in (17) and, conversely, points with different time indexes inside a pseudoperiod will have low-similarity degrees in  $W$ . The  $\sigma$  parameter in the kernel-PCA and the Laplacian eigenmaps algorithms is set to 1 (we performed a grid search on this parameter to determine its best value).

The results are shown in the plots of Fig. 10 and summarized in the tables of Fig. 11. For each DR method, we display two plots which correspond to the 2-D projection obtained by using

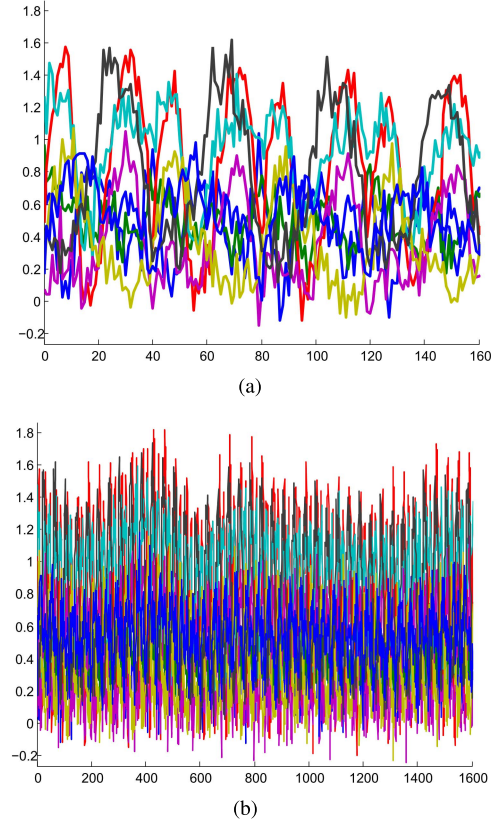


Fig. 9. Synthetic pseudoperiodic signal used in our experiments. Only the first eight dimensions are represented for clarity reasons, each dimension being displayed with a different color. (a) Zoomed-in view over four pseudoperiods. (b) Representation over 40 pseudoperiods. We can see the amplitude modulation appear.

successively the unconstrained and the constrained version of the considered method. We notice that in each case the constrained projection is more distorted than the unconstrained one. But, it should be pointed here that the constrained projections in their raw form [Fig. 10(b), (d), (f), (h), and (j)] would not be displayed as such to the user: we derive instead a line-based representation from the raw representation (see Section IV-A) that provides the user with the insights he wish to get about the data and facilitates its interaction with the representation—which is not possible in the raw representation. The purpose of the diagrams previously cited is just to show how our constrained projection technique behaves compared with the corresponding unconstrained state-of-the-art methods from a DR point of view. Relying solely on the visual aspects and data mining quality measures, our method is not better than the unconstrained DR techniques and even performing a little worse, though, in an acceptable extent as the numerical results of this section prove it [see Fig. 11(a) and 11(b)]. Our claim is to provide a better representation in terms of the potential insights it brings about the data once the postprocessing on the constrained raw representation is performed and the final representation is fed to the user.

The tables of Fig. 11 summarize the numerical results in terms of embedding error, trustworthiness and continuity. We

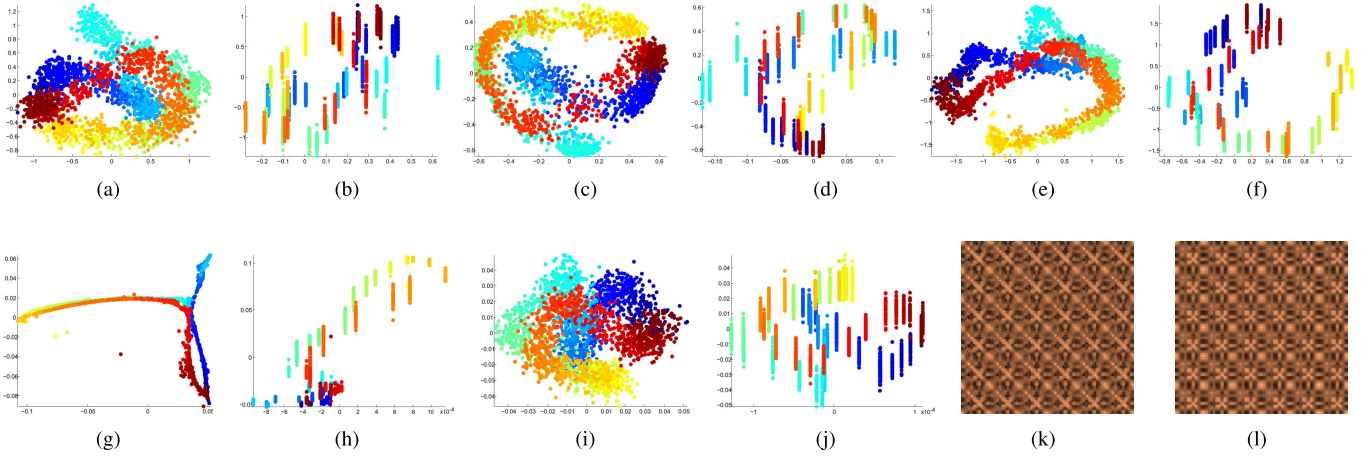


Fig. 10. (a)–(j) Representations of the 2-D projections obtained by using unconstrained DR techniques and their associated constrained (cstr.) version. For each DR technique, the point collections are attributed the same color in the unconstrained and constrained projections, allowing each time to establish a visual correspondence between the two projections. Similarity matrices computed from the (k) data in the true latent space and from the (l) result of the CMDS algorithm using ISOMAP as DR technique.

	Emb. err.			Ad. dist.	Trust.			Cont.		
	ncstr.	cstr.	Rd		ncstr.	cstr.	Rd	ncstr.	cstr.	Rd
PCA	27.1	41.9	64.5	0.35	6.3	14.6	50	3.1	4.5	50
KPCA	22.1	28.2	47.1	0.27	4.5	9.1	50	2.8	4.4	50
ISOMAP	30.8	45.1	104.8	<b>0.2</b>	3.2	<b>6.3</b>	50	3.0	<b>4.4</b>	50
Laplacian	2.7	3.7	5.8	0.34	7.1	11.8	50	5.1	7.3	50
LLE	5.4e-5	9.8e-5	2e-4	0.28	5.5	7.5	50	4.2	6.1	50

(a)

	Emb. err.			Ad. dist.	Trust.			Cont.		
	ncstr.	cstr.	Rd		ncstr.	cstr.	Rd	ncstr.	cstr.	Rd
PCA	25.3	34.6	61.8	0.28	5.3	12.6	50	3.0	3.5	50
KPCA	18.7	23.2	53.2	0.22	4.1	7.3	50	1.9	4.1	50
ISOMAP	27.1	31.3	84.9	<b>0.09</b>	3.1	<b>5.6</b>	50	1.8	<b>2.7</b>	50
Laplacian	2.2	3.1	6.2	0.27	7.0	10.2	50	4.9	5.3	50
LLE	3.0e-5	5.9e-5	1.9e-4	0.19	5.2	7.1	50	3.7	5.4	50

(b)

Fig. 11. Performances of the constrained versus unconstrained projection in terms of embedding error (Emb. err.), trustworthiness (Trust.), and continuity (Cont.). (a) Results on the synthetic dataset described in Section V-A3. (b) Results on the real dataset described in Section V-A4. For each criterion and DR technique, three values are given corresponding, respectively, (from left to right) to: ncstr. the criterion value associated with the unconstrained DR technique, cstr. the criterion value associated with the constrained DR technique (CMDS approach), and Rd the criterion value averaged over several norm-constrained random projections. The additional field Ad. dist. contains the percentage of supplementary distortion incurred by constraining the original DR problem. The added distortion is computed from the first two values contained in the Emb. err. and the total inertia associated with the DR method according to (18).

compare each time the performance of the nonconstrained DR technique with the performance of its constrained counterpart. Since the eigenvalues obtained using different DR methods will be of different magnitudes from one method to the other, the induced projection errors will not be comparable. Thus, rather than comparing directly the projection errors between different methods, we chose to quantify for each method the supplementary amount of distortion incurred by constraining the projection problem. To do so, we rescale the value of the embedding error associated with both the constrained and the unconstrained projections on a  $0 \rightarrow 1$  scale by using the total inertia in the latent space induced by the kernel function as a normalization factor. Thus, it is possible to compare constrained DR methods between each other in terms of an added distortion factor comprised between 0 and 1. Denoting by  $E_u$  and  $E_c$  the projection error, respectively, in the unconstrained and the constrained case and by  $\delta E$  the added distortion, we write the following definition

$$\delta E = \frac{E_c - E_u}{I_{\text{tot}}} \geq 0 \quad (18)$$

where  $I_{\text{tot}}$  is the total inertia in the latent space induced by the kernel function:  $I_{\text{tot}} = \text{Tr}(\tilde{K})$ . The added distortion can be reinterpreted as the percentage of the total inertia which would not be explained by the constrained projection and

thus quantifies directly the loss incurred by constraining the problem.

For each DR method, we also make a comparison with a baseline given by random projections. More precisely, we evaluate the average embedding error obtained by a norm-constrained random projection, i.e., we impose the norm of the first and second coordinate vectors to be, respectively,  $\sqrt{\lambda_1}$  and  $\sqrt{\lambda_2}$  where  $\lambda_i$  is the  $i$ th largest magnitude eigenvalue of the similarity matrix associated with the considered DR method.

Among all the tested DR techniques, we see that ISOMAP is the one which performs best in terms of the three envisaged criteria in the constrained projection case. This could be explained by the fact that the data lie on a piecewise linear manifold with a low curvature (point collections can be approximated by 1-D linear manifolds whose orientations remain very similar between consecutive point collections). This may be the reason why ISOMAP, which works very well for easy-to-embed developable manifolds, performs so well on the considered data. The experiments on real data confirm this claim.

The second conclusion we can draw from the numerical results summarized in Fig. 11 is that our constrained projection method does not add much supplementary distortion to the one introduced by applying the unconstrained 2-D embedding. Our CMDS algorithm performs also very favorably in terms

of trustworthiness and continuity when compared with the corresponding unconstrained DR technique. This confirms that the proposed constrained projection can be used as a sound basis to perform higher level information visualization tasks and extract user-oriented information.

In addition, a visual assessment can be made by looking at the similarity matrices computed from the data in the true latent space induced by the Gram matrix associated with the ISOMAP technique and from the result of the CMDS algorithm. We can see on Fig. 10(l) and (j) that both matrices are very similar visually speaking.

4) *Experiments on a Real Dataset:* The database of real data we consider is composed of sensor data acquisitions over an interval of six months with an acquisition every minute. Hundreds of sensors are installed inside the experimental houses of the INCAS platform, comprising five experimental buildings (with different construction types) dedicated to testing various approaches regarding systems, control, and energy-saving policies. The physical quantities measured include the temperature, the pressure, the relative and absolute humidity, the solar and the infrared level of radiation among others. Sensors are distributed in all the rooms of the houses with several sensors of each category per room, yielding a signal with a very high dimensionality (250 in our case). We perform basic treatments on the data such as prefiltering with a median filter and missing data completion using a standard robust PCA with missing value model (see [38]).

The numerical results are summarized in the table of Fig. 11(b). The conclusions we draw are very similar to the ones we obtain on synthetic data: ISOMAP also performs better than other DR techniques for the three criteria in the constrained projection case and the constrained problem does not induce an important deterioration in terms of embedded error, trustworthiness, or continuity in comparison with the unconstrained problem.

The experiments conducted on this real dataset also allow us to confirm the validity of model (1) in Section III-A regarding the hypotheses it induced on the data structure.

### B. Performance from Information Visualization Point of View

The goal of this subsection is to illustrate the information visualization capabilities of our system. We first give a brief description of the user interface and principal related user-system controls. Then, we detail several use cases for this multidimensional pseudoperiodical time series monitoring system. In first case, we consider a scenario where the user seeks to identify nominal behaviors over different time sub-intervals of the total acquisition time. In second case, we consider a scenario where the user looks for deviant behaviors and tries to identify whether they are caused by a sensor dysfunction or by the observed phenomenon's taking abnormal values.

1) *Description of the Graphical User Interface:* To assess the efficiency of the proposed controls inside the information visualization part of this tool, we implemented a simple graphical user interface (GUI) using SDL/OpenGL (compare Fig. 12). Two controls allow, respectively, to select

the time interval the user wants to display inside a signal pseudoperiod and to tune the wished-for degree of outlieriness. For clarity reasons, the user may also choose to display only a reduced interval of the total data acquisition time, which leads to the creation of a third control (for instance, if we dispose of six months sensor data acquisition, it may be clearer to display only one month of data acquisition to the user, though the CMDS projection algorithm makes a batch computation on the whole data acquisition time).

The proposed interface consists of two parts.

1) The upper part is split between two diagrams. The one on the left hand side represents the CMDS projection over the whole time interval (hence the impression of overplotting). On the one hand, two cursors on the blue horizontal scrollbar allow to restrain the displayed interval within a pseudoperiod (equal to a day in our example). On the other hand, a magenta vertical scrollbar allows to display a user-restrained interval of time over the total acquisition time. The result of using both scrollbars on the left diagram is shown on the right one. The user can click on any of the trajectories in this second diagram and the corresponding sensor signal will be shown in the bottom part of the interface. Finally, the yellow vertical scrollbar allows to control the degree of outlieriness of the plotted trajectories.

2) The bottom part is particularly useful when trying to identify potential sensor dysfunction. Therefore, sensors are sorted by categories, each category being displayed in a separate tab. The display is entirely controlled by the right hand side diagram of the upper part of the interface in the way mentioned above. Hence, when the user clicks on a deviant behavior on the top right hand side graph, he can identify on the bottom left hand side diagram which sensor(s) is/are responsible for the observed deviance and analyze whether this is due to a sensor dysfunction or an abnormal behavior of the observed phenomena.

2) *Use Case Examples:*

a) *First use case:* This use case illustrates the effect of the scrollbars on restraining the displayed time interval and the number of plotted trajectories, so as to make a nominal behavior appear. Fig. 13 shows how using the scrollbars influences the display on the upper right hand side diagram: on the left hand side diagram, the displayed interval of time is restrained to 20 days, which is not restrictive enough to identify clearly a nominal behavior. On the right diagram, we restrain the displayed time interval to 10 days and the maximum degree of outlieriness to 0.8. In this case, the nominal behavior symbolized by the green line on the diagram appears visually consistent with most of the represented behaviors.

The choice of an appropriate interval of time is left to the user who is supposed to have an *a priori* knowledge of the data. As a general rule, the longer is the selected interval of time to be displayed, the less robust the identified nominal behavior will be. Here, the notion of robustness is subjective

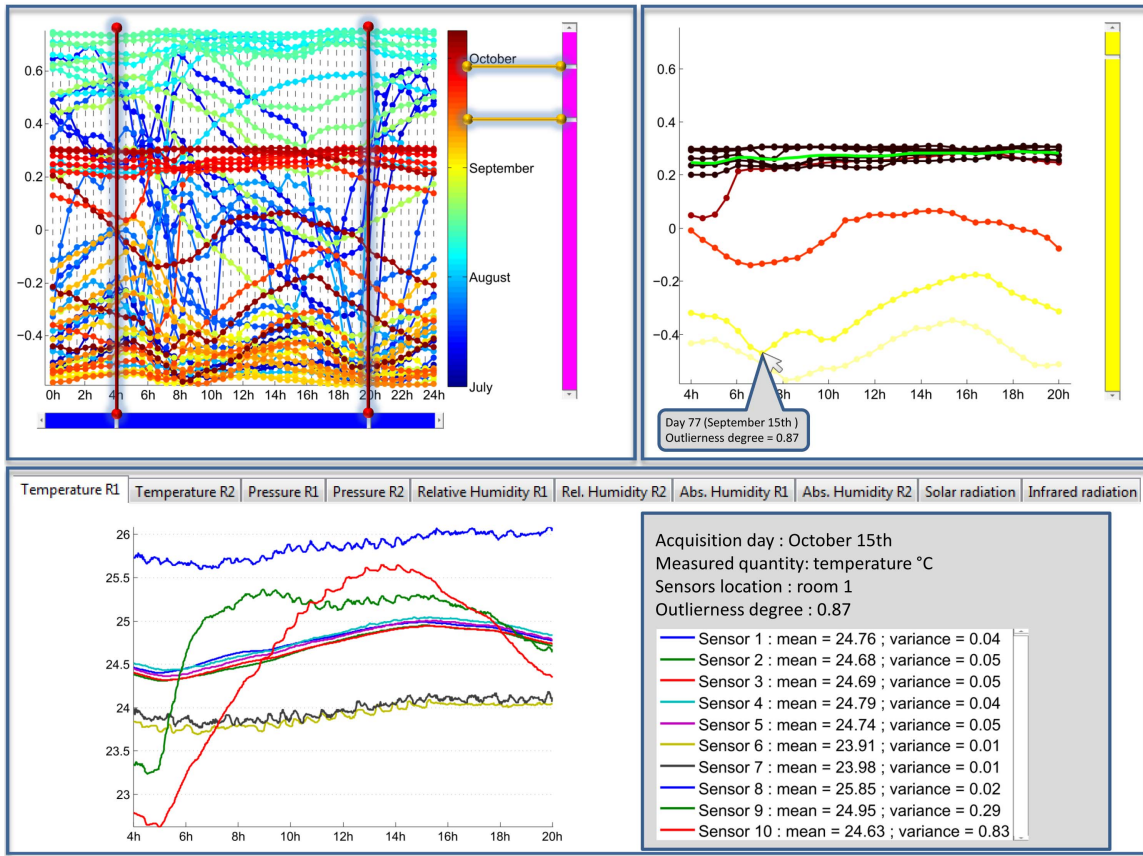


Fig. 12. Screenshot of the proposed monitoring system for multidimensional pseudoperiodical time series. A complete description of the system is given in Section V-B1. This screenshot also illustrates the use case number two, where the goal is to identify the reason why a behavior deviates from the nominal one. With the mouse cursor, the user points a deviant profile in the top right hand side figure and the corresponding original sensor signal is displayed in the bottom part of the interface for the selected time interval (4–20H in this example). The user can then select a category of sensors by using the tabbed window mechanism. Here, the values represented in the bottom part of the interface correspond to the temperature sensors in room 1.

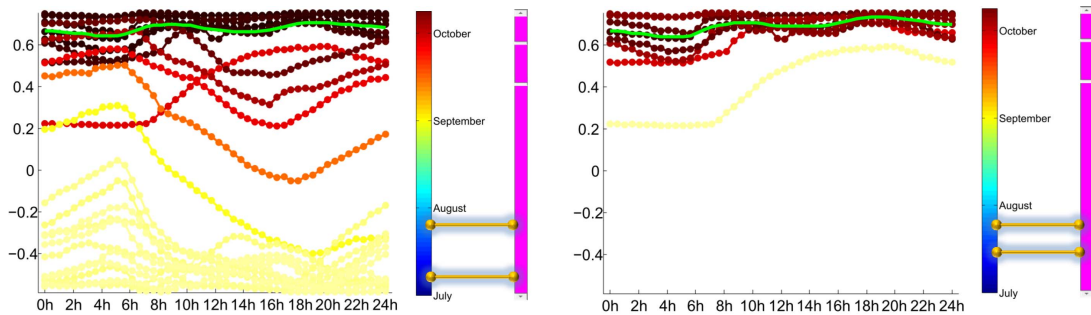


Fig. 13. Impact of using the magenta and the yellow scrollbars to make a nominal behavior appear. On the left, the selected interval of time is not restrictive enough to clearly identify a nominal behavior, whereas on the right, a meaningful nominal behavior can be extracted (displayed in light green).

and left to the visual appreciation of the user who will adjust the selected time interval until he judges significant the obtained nominal behavior (significant is used in the meaning of visually consistent with all the other behaviors plotted on the upper right diagram of the graphical interface). By clicking on a trajectory close to the identified nominal behavior in the top right window, it is possible to observe the sensor values characterizing this nominal behavior in the bottom windows of the interface and to obtain global statistics on these values.

This is important information to the analyst who will be able for instance to define a local heating policy from these observations, or to analyze the thermal behavior of the house—which may be useful for instance to characterize the properties of the insulating materials. For the latter, a very simple user analytical process would be to compare the nominal behaviors during precise periods of the year between different houses equipped with different insulating means. If the heating policies (and of course the location/exposure) are the same



for all the houses as well as the type of measurements taken inside them, the nominal behaviors will directly characterize the properties of the materials used. This supposes of course to run several instances of the system (one for each time series/house). By correlating these data with meteorological data, it is possible to determine the performance of insulating materials for different thermal regimes/seasons, and also, to make a global assessment of the energetic performance of the house.

*b) Second use case:* In this use case, the user selects a trajectory with a high-outlierness degree and analyzes the corresponding sensor signal. The actions to be taken are the following: the user selects first an interval of time over the total acquisition time with the magenta scrollbar and an interval to be displayed inside the time pseudoperiod with the blue scrollbar. On the screenshot shown in Fig. 12, we chose to display the last fifteen days of October and to visualize an interval of time comprised between 4 and 20H. The user then adjusts the wished-for degree of outlierness with the help of the yellow scrollbar (fixed to 0.9 in the example of Fig. 12) and clicks on a behavior in the upper right hand side diagram of the graphical interface. The corresponding sensor data over the selected time interval (4–20H in the example) are then displayed in the bottom part of the interface. The tabs allow to select a specific category of sensors (temperature in room 1 in the example). The user can then analyze the possible causes for the deviance of the behavior he clicked on. In example Fig. 12, we can see very clearly that the observed deviance is due to temperature sensors yielding abnormal values in room number 1.

## VI. CONCLUSION

In this paper, we define a general constrained projection technique, which can enter into the framework of any of the existing DR technique insofar as it can be reformulated under the form of a kernelized MDS. The loss in terms of embedding error, trustworthiness, and continuity incurred by adding the projection constraints is quite limited on both the synthetic and real data tested for each of the considered DR techniques. Contrary to their unconstrained counterparts, the constrained projections obtained with the CMDS approach open the way to the use of information visualization techniques, decoupling the interest of the obtained projections in terms of the amount of information the user can extract from the associated graphical representations presented to him in the display. These representations indeed allow the superimposition of simple controls that can be used to obtain a very specific piece of information as well as to perform standard global monitoring. Moreover, all the operations effected by using these controls are costless from a computational point of view, since they simply imply the reformatting of the visual content of a graph displayed on the screen. This confirms the initial claim of the paper which was to combine powerful data mining techniques with more user-compliant information visualization approaches.

In terms of advantages, the proposed system provides the user with an easy-to-use self-explanatory graphical interface

which does not necessitate any previous user training. The implemented graphical controls are intuitive and are voluntarily restrained to a few number to maximize the efficiency of the search process. The main disadvantage may reside in the lack of generalizability of the system to explore other types of time-oriented data (i.e., data which do not exhibit any pseudoperiodicity). This system has indeed been developed in a well-defined applicative framework in response to real needs in terms of monitoring environmental sensor data in experimental houses—but, it could be used on all kinds of pseudoperiodical time series data, which still opens a broad range of potential applications. Another limitation of the system may lie in the fact that it is designed to treat only one source of data at a time, i.e., the system only allows the processing of a single (multidimensional) time series resulting from the synchronized observation of the same phenomenon by different sensors. In some cases, it may be useful to consider simultaneously several sources of data corresponding to the observation of different (but correlated) phenomena, with measurements not necessarily synchronized in time and/or of different nature, and envisage a cross-domain treatment of these data. In the practical use case we considered, it may be interesting for instance to consider simultaneously meteorological data, which we expect to be correlated with the sensor measurements inside the house. This is an interesting perspective of improvement of the system, which will surely raise new challenges from a data mining and information visualization point of view.

To conclude on the pro and cons, we provided a system dedicated to the analysis of pseudoperiodical time series which has the advantage of simplicity and efficiency, stressing to the user the relevant content and leading him in a few interactions (less than 3) to the targeted piece of information. This simplicity may be a disadvantage if we want to perform more complex tasks (such as cross-domain analysis) or treat more general data, but more complex tasks are often implemented at the expense of simplicity of use, and higher generalizability (regarding data that can be treated) often impacts the efficiency of the search process.

In addition, in a future part of this paper, we plan to extend our CMDS algorithm to accept online streaming data to perform online monitoring. Indeed, although data can be treated in an online fashion by recomputing the CMDS projection after each new acquisition, time series with a high-frequency acquisition rate will necessitate a real online approach updating the constrained projection in real time. This problem is very challenging since it requires a visual continuity in the successive projection updates. Indeed, if we present the user with too much varying projections from one update to the other, he will lose track of what he was observing before the display got updated. And reversely, the update step may prove unnecessary if no significant changes in the observed phenomenon occurred within the update step. Thus, the update interval has to be controlled dynamically to provide the user with reasonable updates, making appear an understandable and continuous evolution of the system.

## COMPUTATION OF THE FIRST AND SECOND ORDER DERIVATIVES OF THE CMDS OBJECTIVE

The notations used in this appendix are the same as those introduced in this paper.

We remind that the CMDS objective is given by

$$F(z, W) = \left\| \tilde{K} - \left( (z \otimes \mathbb{1}_n)(z \otimes \mathbb{1}_n)^T + WW^T \right) \right\|_F^2 \\ \text{Tr} \left( \tilde{A}_W \tilde{A}_W^T \right) - 2\text{Tr} \left( \tilde{A}_W ZZ^T \right) + \left( Z^T Z \right)^2$$

where  $\tilde{A}_W = \tilde{K} - WW^T$  and  $Z = z \otimes \mathbb{1}_n$ . We have  $\partial F(\bullet, W)/\partial z = (\partial F(\bullet, W)/\partial Z)\partial Z/\partial z$  where

$$\frac{\partial F(\bullet, W)}{\partial Z} = -2 \frac{\partial \text{Tr} \left( \tilde{A}_W ZZ^T \right)}{\partial Z} + \frac{\partial \left( Z^T Z \right)^2}{\partial Z} \\ = -2 \left[ \text{vec} \left( \tilde{A}_W \right)^T \left( Z \otimes \mathbb{1}_N + \mathbb{1}_N \otimes Z \right) \right] \\ + \left[ 4 \left( Z^T Z \right) Z^T \right]$$

and  $\partial Z/\partial z = \mathbb{I}_T \otimes \mathbb{1}_n$ , which yields the first derivative. It can be shown using numerical examples that the functional  $F$  is not convex with respect to  $Z$  and, as such, is not convex with respect to  $[z, W]$ . The second derivative with respect to  $Z$  can be expressed in the following form:

$$\frac{\partial^2 F(z, \bullet)}{\partial W^2} = -2 \left[ \text{vec} \left( \tilde{A}_Z \right)^T \otimes \mathbb{I}_N \right] \\ \times \left[ \left( \mathbb{T}_{N,N} \otimes \mathbb{I}_N \right) \left( \mathbb{I}_N \otimes \text{vec} \left( \mathbb{I}_N \right) \right) \right. \\ \left. + \left( \text{vec} \left( \mathbb{I}_N \right) \otimes \mathbb{I}_N \right) \right] \\ + 8WW^T + 4W^T W \quad (19)$$

where  $T_{N,N}$  is such that  $T_{N,N} \text{vec} (A) = \text{vec} (A^T)$ . By noticing that the matrix

$$C = -2 \left[ \text{vec} \left( \tilde{A}_Z \right)^T \otimes \mathbb{I}_N \right] \\ \times \left[ \left( \mathbb{T}_{N,N} \otimes \mathbb{I}_N \right) \left( \mathbb{I}_N \otimes \text{vec} \left( \mathbb{I}_N \right) \right) + \left( \text{vec} \left( \mathbb{I}_N \right) \otimes \mathbb{I}_N \right) \right]$$

is semi-definite negative when  $A_Z$  is semi-definite positive and that the additive term  $+8WW^T + 4W^T W$  can only modify the sign of the first two eigenvalues of the Hessian matrix  $\partial^2 F(z, \bullet)/\partial W^2$ , it is quite easy to find a vector  $W$  such that the resulting Hessian possesses both positive and negative eigenvalues. Thus, the Hessian is not positive definite which proves the functional we optimize is not convex in  $W$  and, as such, is not convex with respect to  $[z, W]$ .

## SUMMARY OF THE CMDS ALGORITHM

**Algorithm 1** Constrained Multidimensional Scaling Algorithm

**Require:** Pseudo-periodic time sequence  $X \in \mathcal{M}(N, d)$  and associated pseudo-period  $T$ , tolerance thresholds  $\epsilon_{\text{tol}}$  and  $\delta_{\text{tol}}$ , minimum gradient step  $\gamma_{\text{min}}$  and initialization gradient step  $\gamma_{\text{init}}$ .

**Ensure:**  $N = n \times T$  with  $n \in \mathbb{N}^*$ .

$\tilde{K} \leftarrow$  Gram matrix associated with  $X$

$\tilde{K} \leftarrow K - CK - KC + CKC$

$Z_{\text{target}} \leftarrow \sqrt{\mu_K^1} v_K^1$

$a \leftarrow \text{RAND}(T, 1)$ ,  $a \leftarrow \frac{a}{\|a\|}$

$z_0 \leftarrow \sqrt{\mu_K^1} a$ ,  $Z_0 \leftarrow z_0 \otimes \mathbb{1}_n$

$\Delta E_{\text{init}}, E_{\text{init}}^0 \leftarrow \infty$

$r \leftarrow 1$

**while**  $\Delta E_{\text{init}} < \delta_{\text{tol}}$  **do**

$\nabla_z f_{\text{init}} \leftarrow 2 \left[ Z_{r-1}^T - \sqrt{\mu_1} v_1 \right] \left( \mathbb{I}_T \otimes \mathbb{1}_n \right)$

$z_r \leftarrow z_{r-1} - \gamma_{\text{init}} \nabla_z f_{\text{init}}$

$E_{\text{init}}^r \leftarrow \|Z_r - Z_{\text{target}}\|^2$

$\Delta E_{\text{init}} \leftarrow E_{\text{init}}^{r-1} - E_{\text{init}}^r$

$r \leftarrow r + 1$

**end while**

$W_0 \leftarrow \sqrt{\mu_K^2} v_K^2$

$\Delta E_{\text{CMDS}}, E_{\text{CMDS}}^0 \leftarrow \infty$

$t \leftarrow 1$

**while**  $\Delta E_{\text{CMDS}} < \epsilon_{\text{tol}}$  **do**

$\tilde{A}_{W_t} \leftarrow \tilde{K} - W_{t-1} W_{t-1}^T$

$\tilde{V}_{W_t} \leftarrow \text{vec} \left( \tilde{A}_{W_t} \right)$

**for**  $i = 1 \rightarrow N$  **do**

**for**  $j = 1 \rightarrow n$  **do**

$\tilde{B}_{W_t}(i, j) \leftarrow \tilde{V}_{W_t}((j-1) * N * n + i, 1)$

**end for**

**end for**

$s_0 \leftarrow z_t$ ,  $S_0 \leftarrow Z_t$

$\gamma_0^* \leftarrow \gamma_{\text{min}}$

$\Delta E_{\text{CGD}} \leftarrow \infty$ ,  $E_{\text{CGD}}^0 \leftarrow E_{\text{CMDS}}^t$

$r \leftarrow 1$

**while**  $\Delta E_{\text{CGD}} < \delta_{\text{tol}}$  **do**

$\nabla_z F_{W_t} \leftarrow -4S_{r-1}^T \tilde{B}_{W_t} + 4n^2 s_{r-1}^T s_{r-1} S_{r-1}^T$

$s_r \leftarrow s_{r-1} - \gamma_{r-1}^* \nabla_z F_{W_t}$

$\Delta_{r,s} \leftarrow s_r - s_{r-1}$

$\Delta_r g = \nabla_s F_{W_t} - \nabla_s F_{W_{t-1}}$

$\gamma_r^* \leftarrow \frac{(\Delta_{r,s})^T \Delta_r g}{(\Delta_{r,s})^T \Delta_r g}$

$S_r \leftarrow s_r \otimes \mathbb{1}_n$

$E_{\text{CGD}}^r \leftarrow \text{Tr} \left( \tilde{A}_{W_t} \tilde{A}_{W_t}^T \right) - 2\text{Tr} \left( \tilde{A}_{W_t} S_r S_r^T \right) + \left( S_r^T S_r \right)^2$

$\Delta E_{\text{CGD}} \leftarrow E_{\text{CGD}}^{r-1} - E_{\text{CGD}}^r$

$r \leftarrow r + 1$

**end while**

$z_t \leftarrow s_{r-1}$ ,  $Z_t \leftarrow S_{r-1}$

$\tilde{A}_{Z_t} \leftarrow \tilde{K} - Z_t Z_t^T$

$W_t \leftarrow \sqrt{\mu_{\tilde{A}_{Z_t}}^1} v_{\tilde{A}_{Z_t}}^1$

$E_{\text{CMDS}}^t \leftarrow \text{Tr} \left( \tilde{A}_{W_t} \tilde{A}_{W_t}^T \right) - 2\text{Tr} \left( \tilde{A}_{W_t} Z_t Z_t^T \right) + \left( Z_t^T Z_t \right)^2$

$\Delta E_{\text{CMDS}} \leftarrow E_{\text{CMDS}}^{t-1} - E_{\text{CMDS}}^t$

$t \leftarrow t + 1$

**end while**

**return**  $(z_t, W_t)$

## REFERENCES

- [1] H. Hurd and A. Miamee, *Periodically Correlated Random Sequences: Spectral Theory and Practice*, vol. 355. New York, NY, USA: Wiley, 2007.
- [2] S. Bittanti and P. Colaneri, *Periodic Systems: Filtering and Control*. New York, NY, USA: Springer-Verlag, 2008.
- [3] A. Vecchia and R. Ballerini, "Testing for periodic autocorrelations in seasonal time series data," *Biometrika*, vol. 78, no. 1, pp. 53–63, Jan. 1991.
- [4] Y. Tesfaye, M. Meerschaert, and P. Anderson, "Identification of periodic autoregressive moving average models and their application to the modeling of river flows," *Water Resour. Res.*, vol. 42, no. 1, Jan. 2006.
- [5] R. Jones and W. Brelsford, "Time series with periodic structure," *Biometrika*, vol. 54, nos. 3–4, pp. 403–408, Dec. 1967.
- [6] M. Pagano, "On periodic and multiple autoregressions," *Ann. Statist.*, vol. 6, no. 6, pp. 1310–1317, Nov. 1978.
- [7] B. Troutman, "Some results in periodic autoregression," *Biometrika*, vol. 66, no. 2, pp. 219–228, 1979.
- [8] G. Adams and G. Goodwin, "Parameter estimation for periodic arma models," *J. Time Ser. Anal.*, vol. 16, no. 2, pp. 127–145, 1995.
- [9] P. Anderson and A. Vecchia, "Asymptotic results for periodic autoregressive moving-average processes," *J. Time Ser. Anal.*, vol. 14, no. 1, pp. 1–18, 1993.
- [10] W. Ku, R. Storer, and C. Georgakis, "Disturbance detection and isolation by dynamic principal component analysis," *Chemometrics Intell. Lab. Syst.*, vol. 30, no. 1, pp. 179–196, 1995.
- [11] P. Molenaar, "A dynamic factor model for the analysis of multivariate time series," *Psychometrika*, vol. 50, no. 2, pp. 181–202, 1985.
- [12] O. Jenkins and M. Mataric, "A spatio-temporal extension to isomap nonlinear dimension reduction," in *Proc. 21st ICML*, Jul. 2004, pp. 56–63.
- [13] J. Tenenbaum, "Mapping a manifold of perceptual observations," in *Proc. NIPS*, 1998, pp. 682–688.
- [14] M. Lewandowski, J. Martinez-del Rincon, D. Makris, and J. Nebel, "Temporal extension of Laplacian eigenmaps for unsupervised dimensionality reduction of time series," in *Proc. 20th ICPR*, Aug. 2010, pp. 161–164.
- [15] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Proc. NIPS*, vol. 14. 2001, pp. 585–591.
- [16] C. M. Bishop, G. E. Hinton, and I. G. Strachan, "GTM through time," in *Proc. IEE 5th Int. Conf. Artif. Neural Netw.*, 1997, pp. 111–116.
- [17] C. M. Bishop, M. Svensen, and C. K. Williams, "GTM: The generative topographic mapping," *Neural Comput.*, vol. 10, no. 1, pp. 215–234, 1998.
- [18] J. Lin, E. Keogh, S. Lonardi, J. Lankford, and D. Nystrom, "Visually mining and monitoring massive time series," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 460–469.
- [19] H. Hochheiser and B. Shneiderman, "Interactive exploration of time series data," in *Discovery Science*. New York, NY, USA: Springer-Verlag, 2001, pp. 441–446.
- [20] J. Van Wijk and E. Van Selow, "Cluster and calendar based visualization of time series data," in *Proc. IEEE Symp. Inf. Visualizat.*, Oct. 1999, pp. 4–9.
- [21] M. Weber, M. Alexa, and W. Müller, "Visualizing time-series on spirals," in *Proc. IEEE Symp. Inf. Visualizat.*, Jan. 2001, pp. 7–13.
- [22] W. Javed, B. McDonnell, and N. Elmqvist, "Graphical perception of multiple time series," *IEEE Trans. Visualizat. Comput. Graph.*, vol. 16, no. 6, pp. 927–934, Nov./Dec. 2010.
- [23] J. Venna, J. Peltonen, K. Nybo, H. Aidos, and S. Kaski, "Information retrieval perspective to nonlinear dimensionality reduction for data visualization," *J. Mach. Learn. Res.*, vol. 11, pp. 451–490, Feb. 2010.
- [24] S. Lespinats, M. Verleysen, A. Giron, and G. Fertil, "DD-HDS: A method for visualization and exploration of high-dimensional data," *IEEE Trans. Neural Netw.*, vol. 18, no. 5, pp. 1265–1279, Sep. 2007.
- [25] M. O. Ward and Z. Guo, "Visual exploration of time-series data with shape space projections," in *Computer Graphics Forum*, vol. 30. New York, NY, USA: Wiley, 2011, pp. 701–710.
- [26] J. Bernard, N. Wilhelm, M. Scherer, T. May, R. K. dos Anjos, J. M. Pereira, et al., "TimeSeriesPaths: Projection-based explorative analysis of multivariate time series data," *J. WSCG*, vol. 20, no. 2, pp. 97–106, 2012.
- [27] W. Aigner, *Visualization of Time-Oriented Data*. New York, NY, USA: Springer-Verlag, 2011.
- [28] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *J. Shanghai Univ.*, vol. 8, no. 4, pp. 406–424, 2004.
- [29] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [30] J. Lee and M. Verleysen, *Nonlinear Dimensionality Reduction*. New York, NY, USA: Springer-Verlag, 2007.
- [31] J. Barzilai and J. Borwein, "Two-point step size gradient methods," *IMA J. Numer. Anal.*, vol. 8, no. 1, pp. 141–148, 1988.
- [32] J. Verbeek, "Learning nonlinear image manifolds by global alignment of local linear models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 8, pp. 1236–1250, Aug. 2006.
- [33] J. Venna and S. Kaski, "Neighborhood preservation in nonlinear projection methods: An experimental study," in *Artificial Neural Networks*. New York, NY, USA: Springer-Verlag, 2001, pp. 485–491.
- [34] L. Van Der Maaten, E. Postma, and H. Van Den Herik, "Dimensionality reduction: A comparative review," *J. Mach. Learn. Res.*, vol. 10, no. 2, pp. 1–35, Feb. 2007.
- [35] J. A. Lee and M. Verleysen, "Quality assessment of nonlinear dimensionality reduction based on K-ary neighborhoods," in *Proc. JMLR: Workshop Conf.*, vol. 4. 2008, pp. 21–35.
- [36] J. A. Lee and M. Verleysen, "Quality assessment of dimensionality reduction: Rank-based criteria," *Neurocomputing*, vol. 72, no. 7, pp. 1431–1443, 2009.
- [37] A. Izenman, *Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning*. New York, NY, USA: Springer-Verlag, 2008.
- [38] S. Roweis, "EM algorithms for PCA and SPCA," in *Proc. Adv. NIPS*, 1998, pp. 626–632.



**Pierre Blanchart** received the Engineering degree in telecommunications from École Nationale Supérieure des Télécommunications de Bretagne, Brest, France, in 2008, the M.S. degree in signal and image processing from Université de Rennes 1, Rennes, France, in 2008, and the Ph.D. degree in machine learning and intelligent systems from Télécom ParisTech, Paris, France, in 2011.

He has been a Post-Doctoral Researcher with CEA, LIST, France, since 2011. His current research interests include data mining/visual data mining, dimensionality reduction/information visualization, content-based image retrieval systems and more generally machine learning and image retrieval, data clustering and classification, object recognition, relevance feedback/active learning and mining large multimedia collections, automatic scheduling/resource allocation problems, (stochastic) dynamic programming, and their application to determine optimal policies in energy management problems such as those encountered in smart grids and eco-neighborhoods.



**Marine Depecker** received the Engineering degree in statistics from École Nationale de la Statistique et de l'Analyse de l'Information, Rennes, France, in 2007, and the Ph.D. degree in machine learning and signal processing from Télécom Paris-Tech, Paris, France, in 2010. The Ph.D. degree was a three-year collaborative work with Renault S.A. Research and Development Department DREAM, Guyancourt, France.

She has been a Research-Engineer with CEA, LIST, France, since 2010. Her main applicative fields are the health care domain and (smart) energy management. Her current research interests include data mining and information visualization, signal and image processing, constrained optimization, statistical modeling.