Application of on-line multivariate time-series segmentation for process monitoring and control

L. Dobos, J. Abonyi

Department of Process Engineering, University of Pannonia, Veszprém, Hungary

Abstract

The future of development of chemical technologies is the analysis of streaming process data. Performing data mining tasks in streaming data is considered a challenging research direction, due to the continuous data evolution. In data streams new values continuously arrive so efficient storage and processing techniques are required to cope with high update rates. In field of process monitoring recursive Principal Component Analysis (PCA) is applied to detect misbehavior of the technology from data streams. In optimization of operating technologies the examinations of transient states needs dynamic PCA for extracting more information to describe the dynamic behavior more accurately. By combining recursive and dynamic PCA and integrating into sliding window time series segmentation technique an efficient on-line multivariate segmentation method could be yielded which is applicable for detecting homogenous operation ranges on-line based on streaming data. The homogeneity of multivariate time-series segments based on the Krzanowsky-similarity factor which compares the hyperplanes determined by the PCA models. With the help of the developed algorithm the on-line separation of operation regimes becomes possible for supporting e.g. parameter estimation for modeling and process control.

I. Introduction

Continuous processes frequently undergo a number of changes from one operating mode to another. The major aims of on-line monitoring plant performance at process transitions are the reduction of off-specification production, the identification of important process disturbances and the early warning of process malfunctions or plant faults. The first step in optimization of transitions is the intelligent analysis of archive and streaming process data [9].

Time series segmentation is often used to extract internally homogeneous segments from a given time series to locate stable periods of time, to identify change points [5]. Although in many real-life applications a lot of variables must be simultaneously tracked and monitored, most of the time series segmentation algorithms are based on only one time-variant variable [9].

The aim of this paper is to develop a new algorithm that is able to handle streaming multivariate data to detect changes in the correlation structure among the variables. Principal Component Analysis (PCA) is the most frequently applied tool to discover such information [14] like in field of fault detection [13]. As linear PCA model defines a hyperplane and the most chemical processes are non-linear it is necessary to eliminate the effect of the older data point which has no information content regarding to the recent PCA model. This goal could be reached with the tools of recursive PCA [6] in which the recent PCA model is yielded by updating the PCA model with the recent measurement. By applying the forgetting factor developed by Fortescue [2] it is possible to determine the number of recently collected streaming data point which is necessary to create the recent and proper PCA model. Combining the method of recursive PCA with the tools of dynamic PCA, presented by Ku[11], a helpful tool is yielded to monitor the behavior of multivariate dynamic systems throughout analysis of streaming data.

To segregate the homogeneous segments from streaming data Keogh [7] presented a segmentation algorithm called sliding window segmentation technique. With the help of it the segmentation of streaming data becomes possible. Since the PCA model defines linear hyperplane, the proposed segmentation algorithm can be considered as the multivariate extension of piecewise linear approximation (PLA) of univariate data sets [7]. Most of data mining algorithms utilize a simple distance measure to compare the segments of different time series. This distance measure is calculated based on the endpoints of the linear lines used to describe the segments [8]. The distance of PCA models could be determined using the PCA similarity factor developed by Krzanowski [10, 12]. By integrating the recursive, dynamic PCA into the sliding window segmentation technique a new, on-line segmentation tool is developed for extracting useful information from streaming multivariate data. The applicability of the proposed algorithm is presented on a simple linear time-variant (LTV) system.

The paper is organized as follows: The aim of time series segmentation is formalized in Section II. Section III. describes the applicability of PCA in multivariate time-series segmentation. After this the new algorithm is presented. Finally Section IV. presents a simple application example. Conclusions are given in Section V.

II. Time Series Segmentation

A time series $T = \{\mathbf{x}_k = [x_{1,k}, x_{2,k}, \ldots, x_{n,k}]^T | 1 \leq k \leq N\}$ is a finite set of N n-dimensional samples labelled by time points $t_1, \ldots, t_N$. A segment of T is a set of consecutive time points $S(a, b) = \{a \leq k \leq b\}$, $\mathbf{x}_a, \mathbf{x}_{a+1}, \ldots, \mathbf{x}_b$. The c-segmentation of time series T is a partition of T to c non - overlapping segments $S_T^c = \{S_i(a_i, b_i) | 1 \leq i \leq c\}$, such that $a_1 = 1, b_c = N$, and $a_i = b_{i-1} + 1$. In other words, an c-segmentation splits T to c disjoint time intervals by segment boundaries $s_1 < s_2 < \ldots < s_c$, where $S_i(s_i, s_{i+1} - 1)$.

The goal of the segmentation procedure is to find internally homogeneous segments from a given time series. To formalize this goal, a cost function $cost(S(a, b))$ describing the internal homogeneity of individual segments should be defined. Usually, this cost function $cost(S(a, b))$ is defined based on the distances between the actual values of the time series and the values given by the a simple function (constant or linear function, or a polynomial of a higher but limited degree) fitted to the data of each segment (the model of the segment). For example in [15, 3] the sum of variances of the variables in the segment was defined as $cost(S(a, b))$:

$$\begin{aligned} cost(S_i(a_i, b_i)) &= \frac{1}{b_i - a_i + 1} \sum_{k=a_i}^{b_i} \parallel \mathbf{x}_k - \mathbf{v}_i \parallel^2, \qquad (1) \\ \mathbf{v}_i &= \frac{1}{b_i - a_i + 1} \sum_{k=a_i}^{b_i} \mathbf{x}_k, \end{aligned}$$

where $\mathbf{v}_i$ the mean of the segment.

The segmentation algorithms simultaneously determine the parameters of the fitted models used to approximate the behavior of the system in the segments, and the a_i, b_i borders of the segments by minimizing the sum of the costs of the individual segments:

$$cost(S_T^c) = \sum_{i=1}^{c} cost(S_i) \, . \qquad (2)$$

This cost function can be minimized by dynamic programming (e.g. [3]), which is unfortunately computationally intractable for many real data sets. Hence, usually heuristic approaches are utilized ([1]).

Since our goal is to develop a multivariate time-series segmentation algorithm which is able to handle the streaming process data, in this paper the sliding window approach is followed. After initialization the algorithm merges the recently collected process data until the stopping criteria is met. The pseudocode for algorithm is shown in Algorithm 1.

Algorithm 1 Sliding window segmentation algorithm

```
Initialize a valid model for segmentation.
while not finished segmenting time series do
    Collect the new process data, and determine the merge cost.
    if $mergecost < maxerror$ then
        Merge the collected data point.
    else
        Start a new segment.
    end if
end while
```

This algorithm is quite powerful since the merging cost evaluations requires simple identifications of PCA models which is easy to implement and computationally cheap to calculate.

III. Application of Principal Component Analysis in Time-Series Segmentation

The aim of this paper is to introduce a time-series segmentation method which is able to handle on-line input-output data sets together in case of exploring homogeneous operation ranges based on collecting process data to support solving engineering problems from e.g. failure diagnosis to even model identification.

1. Classical PCA for multivariate data sets

PCA is an efficient method to handle and explore correlations in multivariable data sets the application of this might be beneficial to reach the goal. Since the aim of this paper is to design an on-line segmentation algorithm that is able to detect changes in the correlation structure among several variables, the cost function of the segmentation is based on the Principal Component Analysis of the $\mathbf{F}_i$ covariance matrices of the segments:

$$\mathbf{F}_i = \frac{1}{b_i - a_i} \sum_{k=a_i}^{b_i} (\mathbf{x}_k - \mathbf{v}_i)(\mathbf{x}_k - \mathbf{v}_i)^T . \tag{3}$$

Principal Component Analysis (PCA) is based on the decomposition of the $\mathbf{F}_i$ covariance matrix $\mathbf{F}_i = \mathbf{U}_i \Lambda_i \mathbf{U}_i^T$ into a Λ_i matrix which includes the eigenvalues of $\mathbf{F}_i$ in its diagonal in decreasing order, and into a $\mathbf{U}_i$ matrix which includes the eigenvectors corresponding to the eigenvalues in its columns. With the use of the first few ($p < n$) nonzero eigenvalues and the corresponding eigenvectors, the PCA model projects the correlated high-dimensional data onto a hyperplane which is useful for the visualization and the analysis of multivariate data.

2. Application PCA for analyzing dynamic systems

The classical PCA is mainly for exploring correlations in data sets without any time dependency. Since in some industrial segments (e.g. in some polymerization processes) the time consumption of grade transitions is in the same order of magnitude with the steady state operation it is crucial to be able to analyze and extract information from data sets collected in transitions. The demand of being able to handle time dependency of the collected data motivates [11] to dynamize the static PCA for the needs of dynamic processes. Consider the following process:

$$Y^n_{k+1} = a_1 Y^n_k + \ldots + a_{ta} Y_{k-ta} + b_1 U^m_k + \ldots + b_{tb} U^m_{k-tb} + c \tag{4}$$

where a_x, b_x and c are constants, t_a and t_b shows the time variance, U^m_k is the k^{th} sample of the (multivariate) input and Y^l_k is the output (product) in the same time. Ku [11] pointed out that performing PCA on the $X_n = [Y_l, U_m]$ data matrix preserves the auto and cross correlations caused by time variance of the time series such as the ones above and it is obviously reduce the performance of the applied algorithm, Thus, he suggested that the X_n data matrix should be formed by considering the process dynamics at every sample points. Generally speaking, every sample points should be completed with the points they are depended on, i.e. the past values:

$$\begin{pmatrix} Y^l_k & U^m_k & Y^l_{k-1} & U^m_{k-1} & \dots & Y^l_{k-n} & U^m_{k-n} \\ Y^l_{k+1} & U^m_{k+1} & Y^l_k & U^m_k & \dots & Y^l_{k-n+1} & U^m_{k-n+1} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ Y^l_{k+m} & U^m_{k+m} & Y^l_{k-1+m} & U^m_{k-1+m} & \dots & Y^l_{k-n+m} & U^m_{k-n+m} \end{pmatrix} \tag{5}$$

Due to the process dynamics, linear relations exist between the inputs and outputs. These relations are preserved under PCA as auto and cross correlations of the scores. Performing PCA on the modified data matrix moves these unwanted correlations to the noise subspace, i.e. the possible combinations of the time dependence are presented in the data matrix and we select the most important combinations of these by using PCA. The first zero (or close to zero) eigenvalue shows the linear relationship between the variables revealed by the eigenvector belongs to this eigenvalue.

3. *Recursive PCA with variable forgetting factor*

In sliding window segmentation there is the demand to create a PCA model in every sample time. To reach this goal the method of recursive PCA is needed to be applied ([4, 6]). The method is based on recursively updating the variance-covariance matrix ($\mathbf{X}^T\mathbf{X}$), where $\mathbf{X}$ is a data matrix comprising p variables and n samples, proposed by Li [6]. In industrial applications data is collected frequently that is why important to calculate the (d)PCA model recursively to avoid the excessive expansion of data matrix. This is a well-known issue in e.g. adaptive control ([2]). The method is based on updating the variance-covariance matrix when a new measurement point is available, applying the equation [6]:

$$(X^TX)_t = \lambda(X^TX)_{t-1} + (1-\lambda)(x^Tx)_t \tag{6}$$

where x_t is the new vector of process measurements, $(\mathbf{X}^T\mathbf{X})_{t-1}$ is the covariance matrix formed from the exponentially weighted observations, λ is the forgetting factor ($0 \leq \lambda \leq 1$) and $(\mathbf{X}^T\mathbf{X})_t$ is the updated variance-covariance matrix. As the forgetting factor decreases the recent observation get more weight in calculation of updated variance-covariance matrix with less weight in being placed on the old data. To calculate the value of the forgetting factor Fortescue [2] proposed an algorithm. The forgetting factor on each observation is dependent on the level of variation of the process measurement. The forgetting factor is calculated as follows:

$$\lambda_t = 1 - \frac{\left[1 - \frac{\mathbf{x}_{t-1}(\mathbf{X}^T\mathbf{X})^{-1}_{t-1}\mathbf{x}^T_{t-1}}{p}\right]\frac{e^2_{t-1}}{p}}{\sqrt{n_{t-1}}} \tag{7}$$

where n_t is the asymptotic memory length at $(t-1)$, $\mathbf{x}_{t-1}(\mathbf{X}^T\mathbf{X})^{-1}_{t-1}\mathbf{x}^T_{t-1}$ is the T^2 metric at sample point $(t-1)$ and the error term e_{t-1} is the Q metric at sample point $(t-1)$ presented in [4].

4. *Distance Measure for PCA Models*

The distances among multivariate PCA models (i.e. hyperplanes) can be evaluated with the PCA similarity factor, S_{PCA}, developed by Krzanowski [10, 12]. It is used to compare multivariate time series segments. Consider two segments, S_i and S_j of a historical data set having the same n variables. Let the PCA models for S_i and S_j consist of p principal components each. The corresponding ($n \times k$) subspaces defined by the eigenvectors of the covariance matrices are denoted by $\mathbf{U}_{i,p}$ and $\mathbf{U}_{j,p}$ respectively. The similarity between these subspaces is defined based on the sum of the squares of the cosines of the angles between each principal component of $\mathbf{U}_{i,p}$ and $\mathbf{U}_{j,p}$:

$$S_{PCA} = \frac{1}{p}\sum_{i=1}^{p}\sum_{j=1}^{p}\cos^2\theta_{i,j} = \frac{1}{p}\text{trace}\left(\mathbf{U}_{i,p}^T\mathbf{U}_{j,p}\mathbf{U}_{j,p}^T\mathbf{U}_{i,p}\right) \tag{8}$$

Because subspaces $\mathbf{U}_{i,p}$ and $\mathbf{U}_{j,p}$ contain the p most important principal components that account for most of the variance in their corresponding data sets, S_{PCA} is also a measure of similarity between the segments S_i and S_j.

5. Recursive dPCA based time-series segmentation

In some cases it may become necessary to separate the operation ranges with different dynamic behavior e.g. fitting a linear model for model predictive control. In the previous section the dynamic principle component analysis was introduced as an approach to reach the goal. Applying the recursive calculation method a new dPCA model becomes accessible in each sample point. By the application of the variable forgetting factor it becomes possible to exclude as much information - contained by the data collected earlier - as included by recent measurements. These dPCA models represented by the variance-covariance matrices become comparable by using the Krzanowski similarity measure $(Eq(8))$. By the help of this similarity measure the application of the segmentation algorithms become available so thus the segments with different dynamic behavior can be segregated.

IV. Case study

As a demonstrating example consider a linear, second order, time variant system represented by cascading the two first order time variant transfer function with the parameters K_1, τ_1 and K_2, τ_2. The time horizon is 1000 sec. The first 50 sec is for creating a proper initial PCA model. The values of K_1 and τ_1 change at $t = 200sec$ from $K_1 = 3$ to $K_1 = 6$ and $\tau_1 = 10$ to $\tau_1 = 20$. The values parameters of the second transfer function change at $t = 666sec$ from $K_2 = 3$ to $K_2 = 5$ and at $t = 800sec$ from $\tau_2 = 20$ to $\tau_2 = 40$. From these changes of the parameters of the transfer functions 4 different segments could be expected. With applying the presented segmentation algorithm (Algorithm 1) our expectation met with the result and the changing points were detected (dotted line with squares in the ends in $Fig(1)$). The applicability of the presented on-line algorithm confirmed by segmenting the time series using the off-line bottom-up segmentation technique (dashed lines in $Fig(1)$). Since the algorithms are convergent thick lines confirm the boarders of the segments, thin lines could be negligible.

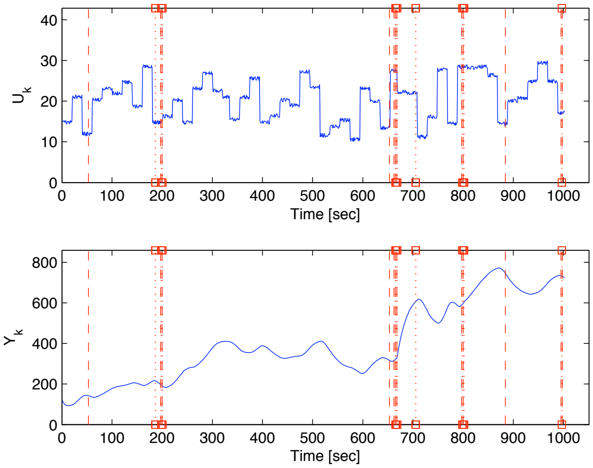

Figure 1: The result of the segmentation

V. Conclusion

In this paper a new multivariate segmentation method was presented based on dynamic PCA. Our main goal was to create an segmentation algorithm which is able to handle multivariate streaming data to segregate the operation regimes with different dynamic character. To reach this goal PCA is necessary to be utilized to extract the hidden information from input-ouput data pairs. To take time-dependency of the collected data into account it is necessary to utilize dynamic PCA. Based on the possibility recursive calculation methodology a new PCA model is yielded in every sample time. This fact is one of the basis factors of creating an on-line segmentation algorithm beside using the Krzanowsky similarity measure. The developed algorithm is tested on a simple example considering a second order, linear, time variant system. Our examination proved that the algorithm could detect the changing points in the parameters of the test system.

Acknowledgement

The financial support from the TAMOP-4.2.2-08/1/2008-0018 (Élhetőbb környezet, egészségesebb ember - Bioinnováció és zöldtechnológiák kutatása a Pannon Egyetemen, MK/2) project is gratefully acknowledged.

References

[1] Feil, B. Abonyi, J. Németh, S. Árva, P. Monitoring process transitions by Kalman filtering and time-series segmentation. *Computers and Chemical Engineering* **2005**, *29*, 1423–1431.

[2] Fortescue, T.L. Kershenbaum, L.S. Ydstie, B.E. Implementation of self-tuning regulators with variable forgetting factors *Automatica* **1981**, *17*, 831–35.

[3] Himberg, J.; Korpiaho, K.; Mannila, H.; Tikanmaki, J.; Toivonen, H.T. Time-series segmentation for context recognition in mobile devices. *IEEE International Conference on Data Mining (ICDM01), San Jose, California* **2001**, 203–210.

[4] Lane, S.; Martin, E.B.; Morris, A.J. Gover, P. Application of exponentially weighted principal component analysis for the monitoring of a polymer film manufacturing process *Transactions of the Institute of Measurement and Control* **2003**, *25*, 17–35.

[5] Last, M. Klein, Y. Kandel, A. Knowledge Discovery in Time Series Databases. *IEEE Transactions on Systems, Man, and Cybernetics* **2000**, *31(1)*, 160–169.

[6] Li, W.; Henry Yue, H.; Valle-Cervantes, S. Joe Qin, S. Recursive PCA for adaptive process monitoring *Journal of Process Control* **2000**, *10*, 471–486.

[7] Keogh, E.; Chu, S. Hart, D.; Pazzani, M. An Online Algorithm for Segmenting Time Series. *IEEE International Conference on Data Mining* **2001**, http://citeseer.nj.nec.com/keogh01online.html.

[8] Keogh, E.; Pazzani, M.J. An Enhanced Representation of Time Series Which allows Fast and Accurate Classification, Clustering and Relevance Feedback. *4th Int. Conf. on KDD.* **1998**, 239–243.

[9] Kivikunnas, S. Overview of Process Trend Analysis Methods and Applications. *ERUDIT Workshop on Applications in Pulp and Paper Industry* **1998**, CD-ROM.

[10] Krzanowsky, W.J. Between Group Comparison of Principal Components. *J. Amer. Stat. Assoc.* **1979**, 703–707.

[11] Ku, W. Storer, R.H. Georgakis, C. Disturbance detection and isolation by dynamic principle component analysis *Chemometrics and Intelligent Laboratory Systems* **1995**, *30*, 179–196.

[12] Singhal, A.; Seborg, D.E. Matching Patterns from Historical Data Using PCA and Distance Similarity Factors. *Proceedings of the American Control Conference* **2001**, 1759–1764.

[13] Bin Shams, M.; Budman, H.; Duever, T. Finding a trade-off between observability and economics in the fault detection of chemical processes. *Computers and Chemical Engineering* **2010**, *In Press*

[14] Tipping, M.E.; Bishop, C.M. Mixtures of probabilistic principal components analysis. *Neural Computation* **1999**, *11*, 443–482.

[15] Vasko, K.; Toivonen, H.T.T. Estimating the number of segments in time series data using permutation tests. *IEEE International Conference on Data Mining* **2002**, 466–473.