Principal Component Analysis based Time-series Segmentation – Clustering of Multivariate Time-series

Janos Abonyi, Balazs Feil, Sandor Nemeth, Peter Arva

University of Veszprem, Department of Process Engineering

P.O.Box. 158, H-8200, Veszprem, Hungary

www.fmt.vein.hu/softcomp

e-mail: abonyij@fmt.vein.hu

***Abstract*—The segmentation of time-series is a constrained clustering problem: the data points should be grouped by their similarity, but with the constraint that all points in a cluster must come from successive time points. This paper proposes a clustering algorithm for the simultaneous identification of fuzzy sets which represent the segments in time and the local PCA models used to measure the homogeneity of the segments. The algorithm is applied to the monitoring of the production of high-density polyethylene.**

I. Introduction

A time-series $T = \{\mathbf{x}_k | 1 \leq k \leq N\}$ is a finite set of N samples labelled by time points $t_1, \ldots, t_N$, where $\mathbf{x}_k = [x_{1,k}, x_{2,k}, \ldots, x_{n,k}]^T$. Real-life time-series can be taken from business, physical, social and behavioral science, economics, engineering [4, 6, 12], etc.

Depending on the application, the goal of the segmentation of a time-series is to locate stable periods of time, to identify change points, or to simply compress the original time-series into a more compact representation [2]. 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 $S_i(s_i, s_{i+1} - 1)$.

Usually the goal 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. For example in [1, 3] the sum of variances of the variables in the segment was defined as $cost(S(a, b))$:

$$cost(S(a,b)) = \frac{1}{b-a+1} \sum_{k=a}^{b} \| \mathbf{x}_k - \mathbf{v}_i \|^2, \qquad (1)$$

$$\mathbf{v}_i = \frac{1}{b-a+1} \sum_{k=a}^{b} \mathbf{x}_k.$$

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

Usually the segmentation algorithms simultaneously determine the θ_i parameters of the models 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.

- **Search for inflection points:** Searching for primitive episodes located between two inflection points [4].
- **Sliding window:** A segment is grown until it exceeds some error bound. The process repeats with the next data point not included in the newly approximated segment. For example a linear model is fitted on the observed period and the modelling error is analyzed [5].
- **Top-down method:** The time-series is recursively partitioned until some stopping criteria is met. [5].
- **Bottom-up method:** Starting from the finest possible approximation, segments are merged until some stopping criteria is met [5].

Although in many real-life applications a lot of variables must be simultaneously tracked and monitored, most of the segmentation algorithms are used for the analysis of only one time-variant variable [6].

The aim of this paper is to develop an algorithm that is able to handle time-varying multivariate data: (i) changes in the mean; (ii) changes in the variance; and (iii) changes in the correlation structure among variables.

Multivariate statistical tools, like Principal Component Analysis (PCA) can be applied to discover such information [7]. PCA maps the data points into a lower dimensional space, which is useful in the analysis and visualization of the correlated high-dimensional data [4]. Linear PCA models have two particularly desirable features: they can be understood in great detail and they are straightforward to implement.

Time-series segmentation may be viewed as clustering, but with a time-ordered structure. The second contribution of this

paper is the introduction of a new fuzzy clustering algorithm which can be effectively used to segment large, multivariate time-series. Since the points in a cluster must come from successive time points, the time-coordinate of the data should be also considered. One way to deal with time is to use time as an additional variable and cluster the observations using a distance between cases which incorporates time. Hence, the clustering is based on a distance measure that contains two terms: the first distance term is based on how the data is in the given segment according to Gaussian fuzzy sets defined in the time domain, while the second term measures how far the data is from hyperplane of the PCA model used to measure the homogeneity of the segments.

The changes of the variables of the time-series are usually vague and do not focused on any particular time point. Therefore, it is not practical to define crisp bounds of the segments. For example, if humans visually analyze historical process data, they use expressions like "this point belongs to this operating point less and belongs to the another more". A good example of this kind of fuzzy segmentation is how fuzzily the start and the end of *early morning* is defined. Fuzzy logic is widely used in various applications where the grouping of overlapping and vague elements is necessary [9], and there are many fruitful examples in the literature for the combination of fuzzy logic with time-series analysis tools [2, 10–12].

The remainder of this paper is organized as follows. Section **??** defines the time-segmentation problem and presents the proposed clustering algorithm. Section **??** contains computational examples using historical process data collected during the production of high-density polyethylene. The aim of this example is to show how the proposed algorithm is able to detect meaningful temporal shapes from multivariate historical process data. The monitoring of a medium and high-density polyethylene (MDPE, HDPE) plant is considered. Finally, conclusions are given in Section VI.

II. PCA based Segmentation Costs

Since the aim of this paper is to design a segmentation algorithm that is able to detect changes in the correlation structure among variables, the cost function of the segmentation is based on the Principal Component Analysis of the $\mathbf{F}_i$ covariance matrices of the segments:

$$\mathbf{F}_i = \frac{1}{s_{i+1} - s_i} \sum_{k=s_i}^{s_{i+1}} \left(\mathbf{x}_k - \mathbf{v}_i^x\right)\left(\mathbf{x}_k - \mathbf{v}_i^x\right)^T. \qquad (3)$$

Principal Component Analysis (PCA) is based on the decomposition of the $\mathbf{F}_i$ covariance matrix

$$\mathbf{F}_i = \mathbf{U}_i \Lambda_i \mathbf{U}_i^T \qquad (4)$$

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:

$$\mathbf{y}_{i,k} = \Lambda_{i,p}^{-\frac{1}{2}} \mathbf{U}_{i,p}^T \mathbf{x}_k \qquad (5)$$

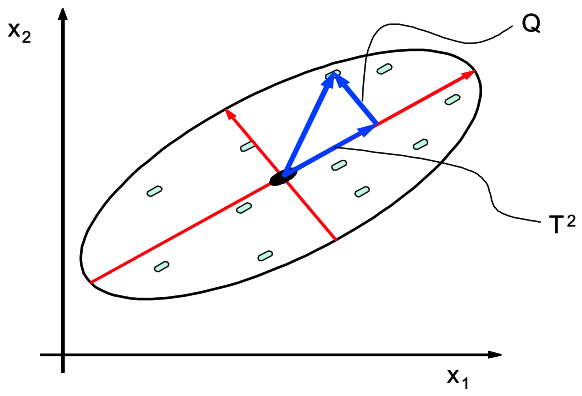

Fig. 1. Distance measures based on the PCA model.

When the PCA model has adequate number of dimensions, the distance of the data from the p-dimensional hyperplane of the PCA model is resulted by measurement failures, disturbances and negligible information. Hence, it is useful to analyze the reconstruction error of the projection:

$$Q_{i,k} = (\mathbf{x}_k - \hat{\mathbf{x}}_k)^T(\mathbf{x}_k - \hat{\mathbf{x}}_k) = \mathbf{x}_k^T(\mathbf{I} - \mathbf{U}_{i,p}\mathbf{U}_{i,p}^T)\mathbf{x}_k. \qquad (6)$$

The analysis of the distribution of the projected data is also informative. The Hotelling T^2 measure is often used to calculate the distance of the mapped data from the center of the linear subspace

$$T_{i,k}^2 = \mathbf{y}_{i,k}^T \mathbf{y}_{i,k}. \qquad (7)$$

Figure 1 illustrates these measures in case of two variables and one principal component.

These T^2 and Q measures are often used for the monitoring of multivariate systems and for the exploration of the errors and the causes of the errors. The main idea of this paper is to use these measures as the measure of the homogeneity of the segments:

$$\begin{aligned} cost_{T^2}(S(a,b)) &= \frac{1}{b-a+1}\sum_{k=a}^{b} T_{i,k}^2 \qquad (8) \\ cost_{Q}(S(a,b)) &= \frac{1}{b-a+1}\sum_{k=a}^{b} Q_{i,k} \end{aligned}$$

III. Bottom-up segmentation Algorithm

In data mining, the bottom-up algorithm has been used extensively to support a variety of time series data mining tasks [14, 15, 16]. In medicine, the algorithm was used by Hunter and McIntosh to provide the high level representation for their medical pattern matching system [10].

The algorithm begins creating a fine approximation of the time series, and the algorithm begins to iteratively merge the lowest cost pair of segments until a stopping criteria is met. When the pair of adjacent segments S_i and S_{i+1} are merged, the algorithm needs to perform some bookkeeping. First, the

cost of merging the new segment with its right neighbor must be calculated. In addition, cost of merging the S_{i1} segment with its new larger neighbor must be recalculated. The pseudocode for algorithm is shown in Table I.

TABLE I
BOTTOM-UP SEGMENTATION ALGORITHM

- Create initial fine approximation.
- Find the cost of merging for each pair of segments: $mergecost(i) = cost(S(s_i, s_{i+2} - 1))$
- while $min(mergecost) < maxerror$
  - Find the cheapest pair to merge: $i = argmin_i(mergecost(i))$
  - Merge the two segments and update the s_i boundary indices.
    $mergecost(i) = cost(S(s_i, s_{i+2} - 1))$
    $mergecost(i-1) = cost(S(s_{i-1}, s_{i+1} - 1))$

  end

IV. HIERARCHIAL CLUSTERING OF SEGMENTS AND TIME-SERIES

For the automatic detection of groups of the segments clustering algorithm can be used to find clusters. Although several clustering algorithms exist, e.g. K-means, Fuzzy C-variaties [12], hierarchical clustering is by far the most widely used clustering method [7].

The starting point for a hierarchical clustering experiment is the similarity matrix which is formed by first computing the distances between the segments.

An advantage in using the segment representation of the time-series is that is allows one to define a variety of distance measures to represent the similarities between two time-series.

With the use of the previously presented segmentation algorithm the historical data sets are organized resulted non-overlapping segments of data. The similarity between these windows of historical data and the current data can then be calculated via appropriate similarity measures.

For the clustering of univariate time-series Keogh has been worked out an efficient method ....

Krzanowski (1979) developed a method for measuring the similarity of two data sets using a PCA similarity factor, D_{PCA}. Consider two data sets which contain the same n variables but not necessarily the same number of measurements. We assume that the PCA model for each data set contains p principal components. The number of principal components (PC) is chosen such that p PCs describe at least 95% of the total variance in each dataset. The similarity between the two data sets is then quantified by comparing their principal components. The appeal of the similarity factor approach is that the similarity between two data sets is quantified by a single number, D_{PCA}.

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 PCs each. The corresponding $(n \times k)$ subspaces defined by the eigenvectors of the covariance matrices are denoted

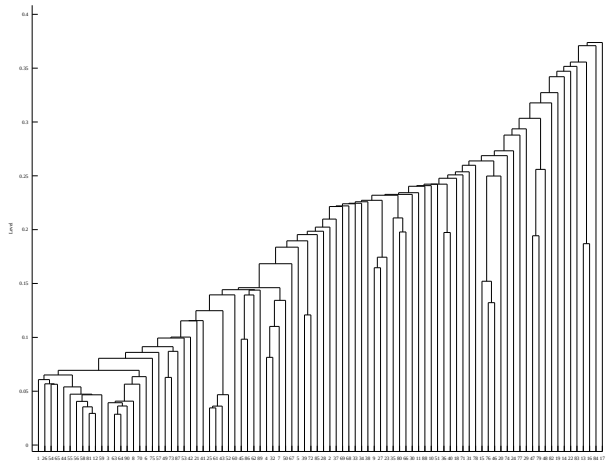

Fig. 2. Dendrogram representation of the result of hierarchial clustering.

by $\mathbf{U}_{i,p}$ and $\mathbf{U}_{j,p}$ respectively. The distance between these subspaces is defined to be (Krzanowski, 1979),

$$D_{PCA} = 1 - \frac{1}{p}\mathbf{U}_{i,p}^T \mathbf{U}_{j,p} \mathbf{U}_{j,p}^T \mathbf{U}_{i,p} \tag{9}$$

The geometric interpretation of D_{PCA} is that it is based on the sum of the squares of the cosines of the angles between each principal component of $\mathbf{U}_{i,p}$ and $\mathbf{U}_{j,p}$.

$$D_{PCA} = 1 - \frac{1}{p}\sum_{i=1}^{p}\sum_{j=1}^{p} \cos^2 \theta_{i,j} \tag{10}$$

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, D_{PCA} is also a measure of similarity between the segments S_i and S_j.

Each distance is then converted into a similarity value . The similarity values are organized in the form of a matrix. The similarity matrix is then scanned for the largest value, which corresponds to the most similar data pair. The two sample constituting the pair are combined to form a new point, which is located midway between the two original points. The rows and columns corresponding to the old data points are then removed from the matrix. The similarity matrix for the data set is then recomputed. This process is repeated until all points have been linked. There are a variety of ways to compute the distances between data points and clusters in hierarchical clustering. The utilized single-linkage method assesses similarity by measuring the distance to the farthest point in the cluster. The results of a hierarchical clustering are usually displayed as a dendrogram, which is a tree-shaped map of the intersample distances in the data set. The dendrogram shows the merging of samples into clusters at various stages of the analysis and the similarities at which the clusters merge, which the clustering displayed hierarchically. Interpretation of the results is intuitive, which is the major reason of these methods.

V. APPLICATION TO PROCESS MONITORING

Manual process supervision relies heavily on visual monitoring of characteristic shapes of changes in process variables,

especially their trends. Although humans are very good at visually detecting such patterns, for a control system software it is a difficult problem. Researchers with different background, for example from pattern recognition, digital signal processing and data mining, have contributed to the process trend analysis development and have put emphasis on different aspects of the field, and at least the following issues characteristic to the process control viewpoint are employed: (I.) Methods operate on process measurements or calculated values in time-series which are not necessarily sampled at constant intervals. (II.) Trends have meaning to human experts. (III.) Trend is a pattern or structure in one-dimensional data. (IV.) Application area is in process monitoring, diagnosis and control [4, 6, 12].

The aim of this example is to show how the proposed algorithm is able to detect meaningful temporal shapes from multivariate historical process data. The monitoring of a medium and high-density polyethylene (MDPE, HDPE) plant is considered. HDPE

HDPE is versatile plastic used for household goods, packaging, car parts and pipe. The plant is operated by TVK Ltd., which is the largest Hungarian polymer production company (www.tvk.hu) in Hungary. An interesting problem with the process is that it requires to produce about ten product grades according to market demand. Hence, there is a clear need to minimize the time of changeover because off-specification product may be produced during transition. The difficulty of the problem comes from the fact that there are more than ten process variables to consider. Measurements are available in every 15 seconds on process variables $\vec{x}_k$, which are the ($x_{k,1}$ polymer production intensity (PE), the inlet flowrates of hexene ($C6_{in}$), ethlylene (C_{2in}), hydrogen ($H2_{in}$), the isobutane solvent (IB_{in}) and the catalyzator (Kat), the concentrations of ethylene (C_2), hexene (C_2), and hydrogen (H_2) and the slurry in the reactor ($slurry$), and the temperature of the reactor (T).

The section used for this example is a 125 hours period which includes at least three segments: a product transition around the 25th hour, a "calm" period until the 100th hour, and a "stirring" period of 30 hours. (see Figure 5).

The distance between the data points and the cluster centers can be computed by two methods corresponding to the two distance measures of PCA: the Q reconstruction error and the Hotelling T^2. These methods give different results which can be used for different purposes. In this paper the results obtained by using the *Fuzzy-PCA-Q method* are presented. The Fuzzy-PCA-Q method is sensitive to the number of the principal components. With the increase of p the reconstruction error decreases. In case of $p = n = 11$ the reconstruction error becomes zero in the whole range of the data set. If p is too small, the reconstruction error will be large for the entire time-series. In these two extreme cases the segmentation does not based on the internal relationships among the variables, so equidistant segments are detected. When the number of the latent variables is in the the range $p = 3, ..., 8$, reliable segments are detected, the algorithm finds the grade transition about the 25th hours Figure **??**).

VI. Conclusions

This paper presented a new clustering algorithm for the fuzzy segmentation of multivariate time-series. The algorithm is based on the simultaneous identification of fuzzy sets which represent the segments in time and the hyperplanes of local PCA models used to measure the homogeneity of the segments. Two homogeneity measures corresponding to the two typical application of PCA models have been defined. The Q reconstruction error segments the time-series according to the change of the correlation among the variables, while the Hotelling T^2 measure segments the time-series based on the drift of the center of the operating region. The algorithm described here was applied to the monitoring of the production of high-density polyethylene. The results suggest that the proposed tool can be applied to distinguish typical operational conditions and analyze product grade transitions.

Acknowledgements

The authors would like to acknowledge the support of the Cooperative Research Center (VIKKK) (project 2001-II-1 and I-7), the Hungarian Ministry of Education (FKFP-0073/2001), and the Hungarian Science Foundation (T037600). Janos Abonyi is grateful for the financial support of the Janos Bolyai Research Fellowship of the Hungarian Academy of Science.

References

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

[2] M. Last, Y. Klein, and A. Kandel, "Knowledge discovery in time series databases," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 31, no. 1, pp. 160–169, 2000.

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

[4] G. Stephanopoulos and C. Han, "Intelligent systems in process engineering: A review," *Comput. Chem. Engng.*, vol. 20, pp. 743–791, 1996.

[5] E. Keogh, S. Chu, D. Hart, and M. Pazzani, "An online algorithm for segmenting time series," *IEEE International Conference on Data Mining*, p. http://citeseer.nj.nec.com/keogh01online.html, 2001.

[6] S. Kivikunnas, "Overview of process trend analysis methods and applications," *ERUDIT Workshop on Applications in Pulp and Paper Industry*, p. CD ROM, 1998.

[7] M. E. Tipping and C. M. Bishop, "Mixtures of probabilistic principal components analysis," *Neural Computation*, vol. 11, pp. 443–482, 1999.

[8] K. Chakrabarti and S. Mehrotra, "Local dimensionality reduction: A new approach to indexing high dimensional spaces," *Proceedings of the 26th VLDB Conference Cairo Egypt*, p. P089, 2000.

[9] J. Abonyi, *Fuzzy Model Identification for Control*. Birkhauser Boston, 2003.

[10] A. B. Geva, "Hierarchical-fuzzy clustering of temporal-patterns and its application for time-series prediction," *Pattern Recognition Letters*, vol. 20, pp. 1519–1532, 1999.

[11] J. F. Baldwin, T. P. Martin, and J. M. Rossiter, "Time series modelling and prediction using fuzzy trend information," *Proceedings of 5^{th} International Conference on Soft Computing and Information Intelligent Systems*, pp. 499–502, 1998.

[12] J. C. Wong, K. McDonald, and A. Palazoglu, "Classification of process trends based on fuzzified symbolic representation and hidden markov models," *Journal of Process Control*, vol. 8, pp. 395–408, 1998.

[13] J. Abonyi, R. Babuska, and F. Szeifert, "Modified Gath-Geva fuzzy clustering for identification of takagi-sugeno fuzzy models," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 32(5), pp. 612–321, 2002.

[14] R. Babuška, P. J. van der Veen, and U. Kaymak, "Improved covariance estimation for gustafson-kessel clustering," *IEEE International Conference on Fuzzy Systems*, pp. 1081–1085, 2002.

[15] F. Hoppner, F. Klawonn, R. Kruse, and T. Runkler, *Fuzzy Cluster Analysis*. Wiley, Chichester, 1999.

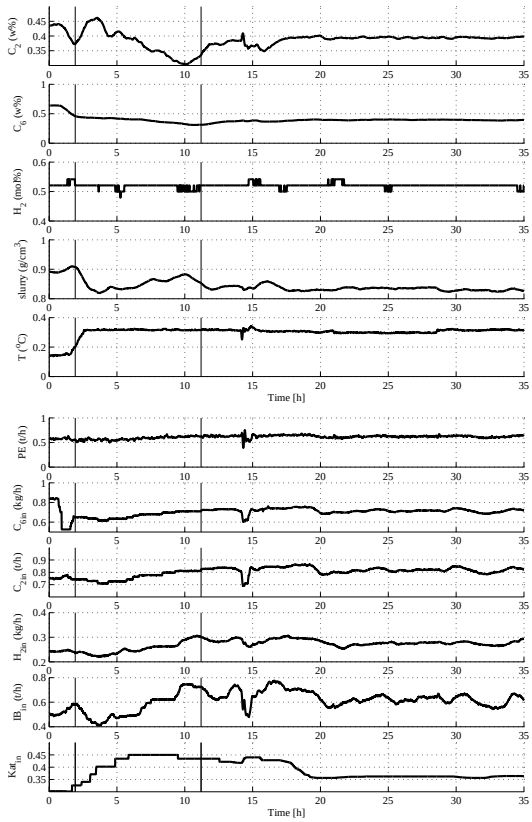

Fig. 3. Example of a time-series of a grade transition.

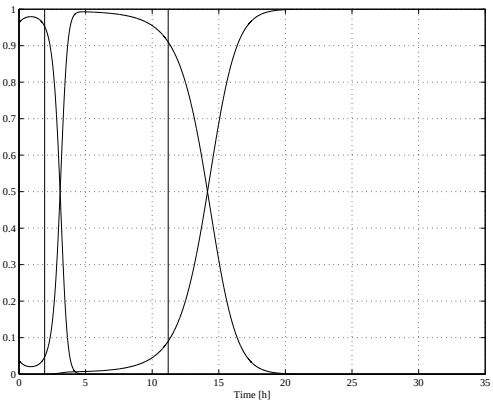

Fig. 4. Example of a time-series of a grade transition.

APPENDIX: FUZZY CLUSTERING BASED SEGMENTATION

When the variance of the segments are minimized during the segmentation, (2) results in the following equation:

$$cost(S_T^c) = \sum_{i=1}^{c} \sum_{k=s_{i-1}+1}^{s_i} \| \mathbf{x}_k - \mathbf{v}_i^x \|^2 = \sum_{i=1}^{c} \sum_{k=1}^{N} \beta_i(t_k) D_{i,k}^2(\mathbf{v}_i^x, \mathbf{x}_k)\,. \quad (11)$$

where $D_{i,k}^2(\mathbf{v}_i^x, \mathbf{x}_k)$ represents the distance how far the $\mathbf{v}_i^x =$ mean of the variables in the i-th segment (center of the i-th cluster) is from the $\mathbf{x}_k$ data point; and $\beta_i(t_k) = \{0, 1\}$ stands

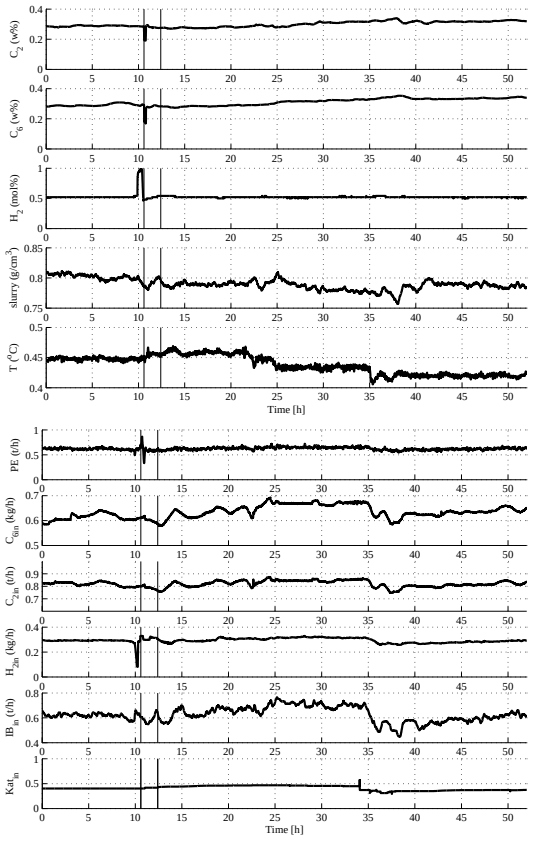

Fig. 5. Example of a time-series of a grade transition.

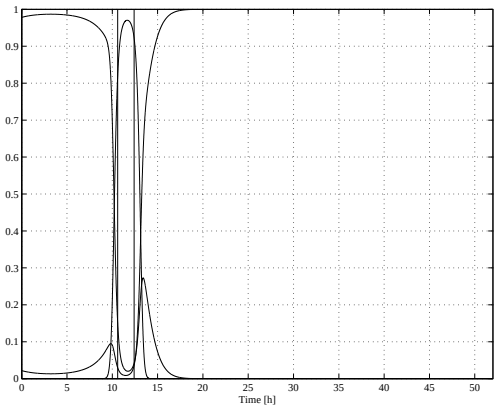

Fig. 6. Example of a time-series of a grade transition.

for the crisp membership of the k-th data point is in the i-th segment:

$$\beta_i(t_k) = \begin{cases} 1 & \text{if } s_{i-1} < k \leq s_i \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

This equation is well comparable to the typical error measure of standard k-means clustering but in this case the clusters are limited to being contiguous segments of the time-series instead of Voronoi regions in R^n.

In such a case the segmentation problem can be described as constrained clustering: data points should be grouped by their similarity, but with the constraint that all points in a cluster must

come from successive time points.

The changes of the variables of the time-series are usually vague and do not focused on any particular time point. As it is not practical to define crisp bounds of the segments, Gaussian membership functions are used to represent the $\beta_i(t_k) = [0,1]$ fuzzy segments of time-series $A_i(t_k)$, which choice leads to the following compact formula for the *normalized degree of fulfillment* of the membership degrees of the k-th observation is in the i-th segment:

$$A_i(t_k) = \exp\left(-\frac{1}{2}\frac{(t_k - v_i^t)^2}{\sigma_i^2}\right),\ \beta_i(t_k) = \frac{A_i(t_k)}{\sum_{j=1}^{c} A_j(t_k)} \tag{13}$$

For the identification of the v_i^t centers and σ_i^2 variances of the membership functions, a fuzzy clustering algorithm is introduced. The algorithm, which is similar to the modified Gath-Geva clustering [13], assumes that the data can be effectively modelled as a mixture multivariate (including time as a variable) Gaussian distribution, so it minimizes the sum of the weighted squared distances between the $\mathbf{z}_k = [t_k, \mathbf{x}_k^T]^T$ data points and the η_i cluster prototypes

$$J = \sum_{i=1}^{c}\sum_{k=1}^{N} (\mu_{i,k})^m D_{i,k}^2(v_i^t, t_k) D_{i,k}^2(\mathbf{v}_i^x, \mathbf{x}_k) \tag{14}$$

where $\mu_{i,k}$ represents the degree of membership of the observation $\mathbf{z}_k = [t_k, \mathbf{x}_k^T]^T$ is in the i-th cluster ($i = 1, \ldots, c$), $m \in [1, \infty)$ is a weighting exponent that determines the fuzziness of the resulting clusters (usually chosen as $m = 2$).

As can be seen, the clustering is based on a distance measure which consists of two terms. The first term, $D_{i,k}^2(v_i^t, t_k)$, is the distance between the k-the data point and the v_i^t center of the i-th segment in time

$$1 \Big/ D_{i,k}^2(v_i^t, t_k) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{1}{2}\frac{(t_k - v_i^t)^2}{\sigma_i^2}\right), \tag{15}$$

where the center and the standard deviation of the Gaussian function are

$$v_i^t = \frac{\sum_{k=1}^{N} (\mu_{i,k})^m t_k}{\sum_{k=1}^{N} (\mu_{i,k})^m},\ \sigma_i^2 = \frac{\sum_{k=1}^{N} (\mu_{i,k})^m (t_k - v_k^t)^2}{\sum_{k=1}^{N} (\mu_{i,k})^m}. \tag{16}$$

The second term represents the distance between the cluster prototype and the data in the feature space

$$1 \Big/ D_{i,k}^2(\mathbf{v}_i^x, \mathbf{x}_k) = \tag{17}$$
$$\frac{\alpha_i \sqrt{\det(\mathbf{A}_i)}}{(2\pi)^{r/2}} \exp\left(-\tfrac{1}{2}(\mathbf{x}_k - \mathbf{v}_i^x)^T (\mathbf{A}_i)(\mathbf{x}_k - \mathbf{v}_i^x)\right),$$

where α_i represents the *a priori* probability of the cluster and $\mathbf{v}_i^x$ the coordinate of the i-th cluster center in the feature space,

$$\alpha_i = \frac{1}{N}\sum_{k=1}^{N} \mu_{i,k},\ \mathbf{v}_i^x = \frac{\sum_{k=1}^{N} (\mu_{i,k})^m \mathbf{x}_k}{\sum_{k=1}^{N} (\mu_{i,k})^m}, \tag{18}$$

and r is the rank of $\mathbf{A}_i$ distance norm corresponding to the i-th cluster. The $\mathbf{A}_i$ distance norm can be defined in many ways. where

$$\mathbf{F}_i = \frac{\sum_{k=1}^{N} (\mu_{i,k})^m (\mathbf{x}_k - \mathbf{v}_i^x)(\mathbf{x}_k - \mathbf{v}_i^x)^T}{\sum_{k=1}^{N} (\mu_{i,k})^m}. \tag{19}$$

- **PCA-Q method:** The distance measure is based on the Q model error. The $\mathbf{A}_i$ distance norm is the inverse of the matrix constructed from the $n-p$ smallest eigenvalues and eigenvectors of $\mathbf{F}_i$ fuzzy covariance matrix

$$\mathbf{A}_i^{-1} = \mathbf{U}_{i,n-p} \Lambda_{i,n-p} \mathbf{U}_{i,n-p}^T, \tag{20}$$

- **PCA-T^2 method:** The distance measure is based on the Hotelling T^2. $\mathbf{A}_i$ is defined with the largest p eigenvalues and eigenvectors of $\mathbf{F}_i$

$$\mathbf{A}_i^{-1} = \mathbf{U}_{i,p} \Lambda_{i,p} \mathbf{U}_{i,p}^T, \tag{21}$$

The optimal parameters of the $\eta_i = \{\mathbf{v}_i^x, \mathbf{A}_i, v_i^t, \sigma_i^2, \alpha_i\}$ cluster prototypes are determined by the following clustering algorithm:

Initialization:

Given a time-series T specify c, choose a termination tolerance $\epsilon > 0$, and initialize the $\mathbf{U} = [\mu_{i,k}]_{c \times N}$ partition matrix randomly.

Repeat for $l = 1, 2, \ldots$

Step 1 Calculate the η_i parameters of the clusters by (16), (18), and (20) or (21).

Step 2

Compute the $D_{i,k}^2(\eta_i, \mathbf{z}_k)$ distance measures (14) by (15), (18).

Step 3

Update the partition matrix

$$\mu_{i,k}^{(l)} = \frac{1}{\sum_{j=1}^{c} \left(D_{i,k}/D_{j,k}\right)^{2/(m-1)}}, \tag{22}$$
$$1 \le i \le c,\ 1 \le k \le N.$$

$||\mathbf{U}^{(l)} - \mathbf{U}^{(l-1)}|| < \epsilon.$