Acta Agraria Kaposváriensis (2007) Vol 11 No 2, 197-209
Kaposvári Egyetem, Állattudományi Kar, Kaposvár
University of Kaposvár, Faculty of Animal Science, Kaposvár

Time-Series Similarity Application to Qualitative Process Trend Analysis

B. Balaskó, S. Németh, J. Abonyi
University of Pannonia, Department of Process Engineering, H-8200 Veszprém, Egyetem Str. 10.

ABSTRACT

Beside the widely applied quantitative statistical tools, qualitative methods get more and more popular in the field of data mining techniques. Qualitative results are often easier to understand to a user, but to achieve such results, these methods always claim for a priori knowledge of the object they analyze. This paper proposes a technique that is able to compare and qualify time series in an unsupervised way, whereto even a priori knowledge can be incorporated. The two main steps of our method are: it applies triangular episode segmentation proposed by Cheung and Stephanopoulos to get a symbolic trend representation, and secondly it compares episode sequences by pairwise sequence alignment, a known technique in bioinformatics for aligning amino acid sequences based on a dynamic programming matrix filled with transformation weights. An alignment is considered as optimal if sum of weights is minimal. Instead of weights, our technique applies a predefined similarity measure. The algorithm was made up with data preprocessing methods to handle multidimensional, noisy data as well: Principal Component Analysis and Gaussian-filter, respectively. It is shown that the presented technique is able to compare, classify or qualify time series to discover their similarities. The algorithm was tested on industrial process data as well to show how it works on process trends and how it supports the analysis of product transitions in a multi-product polymerization plant.

(Keywords: qualitative trend analysis, episode segmentation, sequence alignment)

ÖSSZEFOGLALÁS

Idősorok hasonlóságának alkalmazása kvalitatív trend elemzés céljából

Balaskó B., Németh S., Abonyi J.
Pannon Egyetem, Folyamatmérnöki Intézeti Tanszék, 8200 Veszprém, Egyetem u. 10.

Az adatbányászati eszközök közül a széles körben alkalmazott statisztikai módszerek mellett egyre népszerűbbek a minőségi adatelemző technikák. Ezek kimenetei általában könnyebben értelmezhetőek a felhasználó számára, de ilyen eredmények eléréséhez ezek az eszközök gyakran az elemzett rendszer a priori ismeretének felhasználását igénylik. Ez a cikk egy olyan technikát mutat be, amely képes felügyelet nélkül összehasonlítani változók idősorát, és amelybe az előzetes ismeretek felhasználhatóak. A módszer két fő lépése a következő: Cheung és Stephanopoulos által kifejlesztett háromszög epizód szegmentáció alapján a trendeket szimbolikus epizódok szekvenciájára bontja, majd ezeket bioinformatikai szekvencia-illesztéssel összehasonlítja. A szekvencia-illesztés ismert módszer a bioinformatikában aminosav-szekvenciák vizsgálatára, amely egy dinamikus programozási mátrixon alapul, *melynek elemei transzformációs súlyok. Az optimális illesztése két szekvenciának a minimális transzformációs súlyösszegű illesztés. Az alkalmazott technika súlyok helyett előre definiált pontértékeket maximalizál az optimális illesztés megtalálása érdekében. Az algoritmust kiegészítettük annak érdekében, hogy többdimenziós, zajjal terhelt adatok esetében is hatékonyan működjön, ezért főkomponens elemzést és Gauss-szűrőt alkalmaztunk. A cikkben bemutatásra kerül, hogy a kifejlesztett módszer alkalmas többváltozós idősorok összehasonlítására, osztályozására és minősítésére a trendek közötti hasonlóságon alapulva. Az algoritmust valós polimerizációs technológiai adatokon is teszteltük annak érdekében, hogy megvizsgáljuk, hogyan teljesít, valamint alkalmas-e a többtermékes technológia termékváltásainak elemzésére.*

(Kulcsszavak: kvalitatív trendanalízis, epizód-szegmentáció, szekvencia-illesztés)

INTRODUCTION

Modern control systems in chemical industry due to the high level if automation generally have data collection and storage ability, thus enormously large amount of data exist for further analysis to mine useful information. Unfortunately, about 10 percent of the collected data is considered to be analyzed (*Fayyad and Simoudis*, 1997), hence there is a giant need for techniques that can compress data with minimal loss of information or have the potential for fast and efficient information mining. Quantitative methods that claim for large number of patterns to get reliable information are very popular because of *easy-to-use* nature, but their *hard-to-understand* output results that they need to be integrated to qualitative techniques. This semi-qualitative way leads to qualitative trend analysis (QTA) where results are generated with an incorporation of the two approach and they are understandable and informative for a user. QTA techniques consist of two basic steps: (i) data preprocessing and (ii) comparing preprocessed data based on a predefined similarity / distance measure.

Data processing deals with the problem how a time series, i.e. a trend of a variable can be represented to be an effective basis for further analysis. The approach can be non-data-driven, e.g. wavelet transforms, spectral transforms (Fourier), piecewise aggregate approximation (PAA), or data-driven, e.g. piecewise linear approximation (PLA), Singular Value Decomposition (SVD), trees and symbolic representation. Lot of research has been done on every technique in the literature, a hierarchy of these techniques is presented in (*Lin et al.*, 2003), where many references are cited.

To compare the preprocessed data, there is always a need for an adequate similarity or distance measure. In the literature different measures are applied that fulfill the requirements of being a metric function, namely the attribute of *non-negativity*, *identity*, *symmetry* and *subadditivity* (also called triangle-inequality), like Euclidean distance, Mahalanobis distance, Chebyshev distance, etc. Some distance measures are also defined to compute distances between strings, like Hamming distance or its generalization, Levenshtein distance. In bioinformatics similarity matrices are widely spread, which are computed empirically (PAM and BLOSUM matrices), hence they not fulfill all requirement of a metric. The applied data processing technique determines the type of applicable distance measures.

From data compression point of view, the amount of data and its dimensionality have to be lowered significantly with an acceptable loss of information which resulted in various sampling, regression and segmentation techniques for size reduction and different mapping and scaling methods for dimensionality reduction, like principal component analysis (PCA), or multidimensional scaling (MDS).

This paper proposes a segmentation technique based on symbolic representation, triangular episode segmentation (*Cheung, Stephanopoulos*, 1990). Segmentation means finding time intervals where a trajectory of a state variable is homogeneous. Segments can be linear, steady-state or transient, indicative for normal, transient or abnormal operation, hence segmentation based feature extraction was applied by many researchers for system monitoring, process fault diagnosis or operator support system (*Venkatasubramanian*, 1995; *Sundarraman and Srinivasan*, 2003; *Charbonnier et al.*, 2005).

Cheung and Stephanopoulos proposed a second order segmentation method for process trend analysis, the application of episodes with a geometrical representation of triangles. Triangular episodes use the first and second derivatives of a trend on a geometrical basis, hence seven primitive episodes can be achieved as characters, which note the shape of the time series over a time interval. These episodes can be partitioned into fuzzy episodes by change of magnitude and duration to have a larger symbol set for representing trends (*Wong et al.*, 1998).

To compare symbolic representations, one can find much less solution in the literature, which can be explained as: other methods use a representation space, where it is easier to achieve a similarity / distance measure between time series transformed by them, because they under-estimate the original distances (*Faloutsos et al.*, 1994).

Lin and Keogh applied a PAA based symbolic representation technique called SAX (Symbolic Aggregate approXimation), where the distance of two symbols were defined as follows: considering normalized data with Gaussian distribution and breakpoints separating equal-sized areas under Gaussian curve, every area can mapped as a single character that corresponds to a value interval over a PAA segment. Distance of two characters is the resultant of the maximal and minimal distance of their breakpoint values (*Lin et al.*, 2003).

It was also shown that dynamic time warping (DTW) is able to compare DNA sequences if transformation weights of amino acids are at hand (e.g. *Srinivasan*, 2006). DTW applies a dynamic programming matrix that is filled up with a predefined distance or similarity measure of every element in both sequences (or time series), and it aligns them by finding the shortest path with minimal distance / maximal similarity. Originally it was developed for speech recognition (*Itakura*, 1975, *Sakoe and Chiba*, 1978). Going towards this dynamic programming technique, another possibility is *global pairwise sequence alignment* to align symbolic sequences, which is a basic application in computational biology or bioinformatics, using mutation, injection and deletion operators for optimal alignment based on an analogous dynamic programming matrix as DTW (*Needleman and Wunsch*, 1970, *Waterman and Smith*, 1978, *Waterman*, 1984).

The short overview above shows that comparing strings is not famous in trend analysis although there are already techniques in other research fields that can effectively handle aligning symbolic sequences, like DTW and pairwise alignment. The proposed algorithm applies therefore episode segmentation and pairwise sequence alignment to compare time series, it is extended by PCA and Gaussian filter to handle multidimensional and noisy data as well. As a case study, it is applied to compare multidimensional time series attained from a Hungarian multi-product polymer plant. Product transitions are extremely important to be managed in a reproducible way, hence our analysis focuses on their hierarchical clustering and visualization by MDS.

The algorithm is also able to find subsequences similar or equal to a predefined sequence, “motif” (*Lin et al.*, 2002) in a wide range time interval, hence identify product transitions in an unsupervised way.

MATERIAL AND METHOD

Principal Component Analysis (PCA)

This subsection aims a general description of the applied dimensionality reduction technique, Principal Component Analysis. PCA or hotelling algorithm is a widely known and applied method for lowering the dimensionality of a data set from n to q dimensions ($q<n$) based on its multidimensional structure and find patterns in data (*Smith*, 2002).

During an orthogonal linear transformation from n to the lower dimension of q, PCA calculates the eigenvectors and eigenvalues of the n-dimensional preprocessed (zero prospective value) data and selects the largest q eigenvalues, which corresponding eigenvectors create a subspace, where the original data is projected into. In other words, it finds the most significant directions with the largest variance in the data set.

As a formal description, let **X** be an $n\times N$ dimensional data set, where N means the number of observations. The aim of PCA is to find a **P** orthonormal n×n projection matrix that fulfills the following equation:

$$\mathbf{P}^{-1} = \mathbf{P}^{T} \tag{1}$$

$$\mathbf{Y} = \mathbf{P}^{T}\mathbf{X} \tag{2}$$

where **Y** is the diagonal covariance matrix of the transformed data. In general, the principal components of a data set are calculated from the correlation matrix **C** by the following eigenvalue equation:

$$\mathbf{C}\mathrm{p} = \lambda\mathrm{p} \tag{3}$$

The eigenvalues are decreasingly sorted, the eigenvector corresponding to the largest eigenvalue will be the first principal component and so on. The selected principal components are collected in **P** and the projected data are calculated by $\mathbf{Y} = \mathbf{P}^{T}\mathbf{X}$. If not all the principal components are selected, the projection error of PCA (cumulative percentage of the selected eigenvalues) can be given as $\sum_{i=q+1}^{n}\lambda_i$, and the reconstruction error means $\|\mathbf{PY}-\mathbf{X}\|$.

The presented algorithm has an implementation of Dynamic PCA (DPCA), which can extract time-dependent relations in the measurements, because it mimics the concept of an ARMAX (auto-regressive moving average exogenous) time series model by forming the data matrix with the previous observations in each observation vector (*Ku et al.*, 1995). This means that DPCA algorithm has an extra parameter called time delay parameter, which expresses the time shift between data points: e.g. it shifts the output values by the residence time of a reactor in order to have the corresponding input-output data pairs at the same timepoint.

Gaussian filtering

In this paper the widely known and applied Gaussian filter was applied in order to make high frequency noises vanish from signals. The convolution kernel is as follows:

$$x(\sigma,t)=x(t)\circ f(\sigma,t)=\int_{-\infty}^{+\infty}x(u)\cdot\left\{\frac{1}{\sigma\sqrt{2\pi}}\cdot\exp\left[\frac{-(t-u)^{2}}{\sigma^{2}}\right]\right\}du \tag{4}$$

By increasing σ filtering parameter, more and more feature vanish from the signal, hence the sequence length shorten. Considering this, it is always the user's responsibility to choose an appropriate σ value that corresponds to the current problem.

Triangular Episode Segmentation

To get from a quantitative to a qualitative representation of a real-valued *x(t)* function, it is considered, if we know the value and the derivatives of a function over a time interval, the state of that function is completely known. The continuous state (*CS*) over a closed time interval can be defined as a point value, which is a triplet (if *x(t)* is continuous in *t*):

$$CS(x,t) \equiv PtVl(x,t) = \langle x(t), x'(t), x''(t) \rangle \qquad (5)$$

Consequently, a continuous trend can be defined as continuous sequence of states. For discrete functions, as an approximation, an underlying continuous function has to be known since the derivatives of single points cannot be performed. These definitions lead to a qualitative description of a state (QS) and trend if *x* is continuous at *t*, otherwise it is undefined:

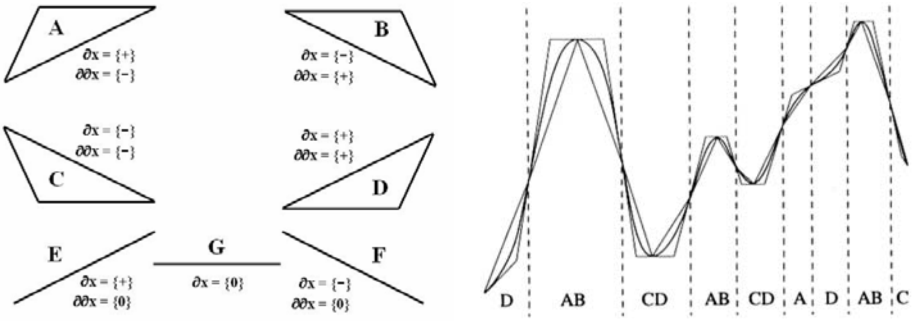
$$QS(x,t) = \langle [x(t)], [x'(t)], [x''(t)] \rangle \qquad (6)$$

where [*x(t)*],[*x'(t)*] and [*x"(t)*] can be {-; 0; +}, depending if they have negative, zero or positive values. Obviously, a qualitative trend of a reasonable variable is given by the continuous sequence of qualitative states.

QS(x; t) is called an *episode* if it is constant for a maximal time interval (the aggregation of time intervals with same *QS*), and the final definition of a trend of a reasonable function is a sequence of these maximal episodes. An ordered sequence of triangular episodes is the geometric language to describe trends. It is composed of seven primitives noted as {A,B,C,D,E,F,G} illustrated in *Figure 1.* These seven primitives can be partitioned into a set of 57 episodes by predefining thresholds in order to assign every episode to be {small(s);medium(m);large(l)} by duration and magnitude (*Wong et al.*, 1998). These thresholds can only be achieved by a priori knowledge, i.e. a preliminary analysis of the possible ranges and changes of the variables.

Figure 1

Seven primitive episodes proposed by Cheung and Stephanopoulos and a sample trend representation



1. ábra: Cheung és Stephanopoulos által javasolt hét epizód alaptípus és egy példa reprezentáció

Pairwise Sequence Alignment

Sequence alignment is typical expression of bioinformatics, where amino acid or nucleotide sequences have to be compared, how far the evolved new sequences are from the elders, i.e. how old they are, and how many mutation steps were needed to result in the new sequence.

Applying the *minimal evolution*, one tries to find the least mutation steps between the elder and offspring sequence. Naive algorithms compare all possible alignments and select one with minimal sum of transformation weights.

Fast algorithms calculate in an other way: Let A_n be a n-element sequence and B_m a m-element sequence, a_n and b_m their nth and mth element; a $\alpha^*(A_n;B_m)$ denotes the set of optimal pairwise alignments of A_n and B_m and $w(\alpha^*(A_n;B_m))$ the sum of transformation weights for these optimal alignment. The basic idea in fast algorithms is that if we know $w(\alpha^*(A_{n-1};B_m))$, $w(\alpha^*(A_n;B_{m-1}))$ and $w(\alpha^*(A_{n-1};B_{m-1}))$, then $w(\alpha^*(A_n;B_m))$ can be calculated within a constant time period. If we leave the last aligned pair in an optimal alignment of A_n and B_m then we get an optimal alignment of $(A_{n-1};B_m)$, $(A_n;B_{m-1})$ or $(A_{n-1};B_{m-1})$, depending on that last mutation step was a deletion, injection or substitution, respectively:

$$w(\alpha^*(A_n,B_m)) = \min\left\{\begin{matrix} w(\alpha^*(A_{n-1},B_m)) + w(a_n \to -) \\ w(\alpha^*(A_n,B_{m-1})) + w(- \to b_m) \\ w(\alpha^*(A_{n-1},B_{m-1})) + w(a_n \to b_m) \end{matrix}\right\} \tag{7}$$

The optimal alignment weights are given in a dynamic programming matrix, **D** with a size of $(n+1)\times(m+1)$. The initial conditions for the 0th row and column:

d0,0 = 0;

$$d_{i,0} = \sum_{l=1}^{i} w(a_l \to -) \tag{8}$$

$$d_{0,j} = \sum_{k=1}^{j} w(- \to b_k) \tag{9}$$

These equations are the main difference between DTW and pairwise alignment, to fill up **D** matrix, the above minimization equation is used. Optimal alignments are started at $d_{n,m}$ and ended in $d_{0,0}$, while in every step the minimal weight is chosen and stepping left means an injection, stepping upwards means a deletion and stepping diagonally upwards means a substitution or perfect match. This method was developed by Needleman and Wunsch.

We applied a scoring matrix instead of transforming weights to align our episode sequences, the only difference is that it maximizes the score of alignment in every step, and the optimal alignment will be the path with the maximal score. The rules for filling up the scoring matrix are as follows:

- Each score of a mutation can be between 0 and 10;
- gaps have a scoring value of -0.1 (injecting a gap is slightly penalized);
- $w(\mathrm{X} \to \mathrm{X}) = 10$, perfect alignment gains a score of 10;
- Score of the alignment of an increasing {A,D,E} and a decreasing episode {B,C,F} is zero;
- Every episode is similar to the steady episode {G} with a score of 2.

This similarity measure is not a classical metric, because it does *not* fulfill triangle inequality.

Scores are weighted according to the type of the episode during alignment: it means a multiplier of 1/3 if duration / change of magnitude is {small} and 2/3 if {medium}. Moreover, score values are normalized by the size of the shorter sequence in order to balance the scores of shorter and longer alignments hence not only for one single alignment but for the alignment of the whole sequence the maximal score is 10.

The proposed algorithm

The algorithm was realized and implemented by the authors in MATLAB® environment and it consists of the following steps:

1) Problem-specific selecting and parametering algorithm features:
    a) PCA, choosing time delay parameter for DPCA (even for a 1-D signal);
    b) Gaussian filter: increasing filter parameter makes high-frequency-features vanish from trend;
    c) Appropriate thresholds for {small, medium, large} markers;
    d) Possible Motif-discovery application;
2) Data pre-processing: (dimensionality reduction + filtering) + episode segmentation;
3) Optimal alignment of two episode sequences based on maximizing a score value.

Application and efficiency highly depends on step 1), the other two steps are automatic and can be repeated as many times as needed for problem solving.

A polymerization technology

The presented algorithm was tested on product changing strategies of a Hungarian polymer plant (Himont technology). Polymer product quality is indicated by melt flow index (MFI) that higly depends on hydrogen contentration. Production is performed in product cycles to minimize or vanish offgrade product: 6 homopolymers are produced with rising MFI, then 11 copolymers with decreasing MFI in a cycle of approximately one month of production (denoted by product codes from H1 to H6 and from C1 to C11 respectively), i.e. frequent process transitions are needed (in every 1-2 days), which means leading the technology from a steady operation regarding a product into an other steady operation regarding the next product. Notation for transitions is as follows: {starting product code}-{final product code}, e.g. H6-C1 means a transition from 6th homopolymer to 1st copolymer product.

From the production data of these polymers a process database was implemented, which contains all the major variables of production with a sample time of 15 seconds (which is the original sampling interval of the control system) of one year continuous production.

The transitions are performed by operators quickly (in less than 2 hours) and manually by tuning five different process values: (i) hydrogen inlet feed to 1st and (ii) 2nd loop reactors, (iii) catalyst inlet flow into the 1st reactor, (iv) reactor temperatures in 1st and (v) 2nd reactors. Concluding from this, every transition can be characterized by these five process variables and a 2-hours-period of production (480 data point values). An example of a process transition can be seen on *Figure 2*.

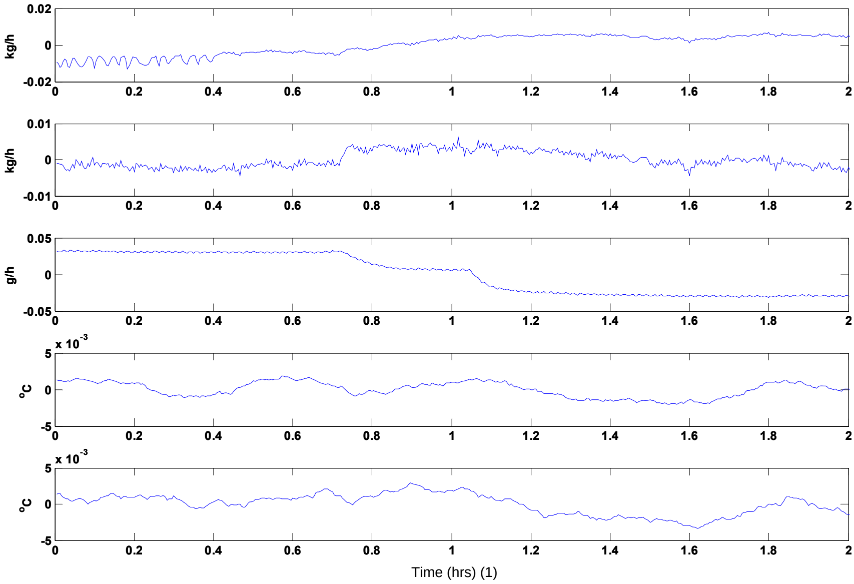
RESULTS AND DISCUSSION

A total set of 31 transitions were compared based on the score of alignment of 2-hour-trends collected from a 2-month-period of production, thus some transitions occur more than one times in order to be able to compare the same type of transitions as well.

While episode segmentation can only handle 1-D trends, every 5-D product transition is projected with the same principal component into a 1-D data series as a function of time with an eigenvalue percentage of 83%, i.e. 17% projection error . An additional analysis showed that application of dynamic PCA increases this information loss, thus static PCA was used.

Figure 2

An example of a C2-C3 product transition profile (from top to bottom: normalized data of hydrogen inlet feed to 1st and 2nd loop reactors, catalyst inlet flow into the 1st reactor, reactor temperatures in 1st and 2nd reactors)



2. ábra: Példa egy C2 termékről C3 termékre történő termékváltásra (normalizált adatok, sorrendben: hidrogén betáplálásaz 1. és 2. reaktorba, katalizátor betáplálás az 1. reaktorba, reaktor hőmérsékletek az 1. és 2. reaktorban).

Idő (óra)(1)

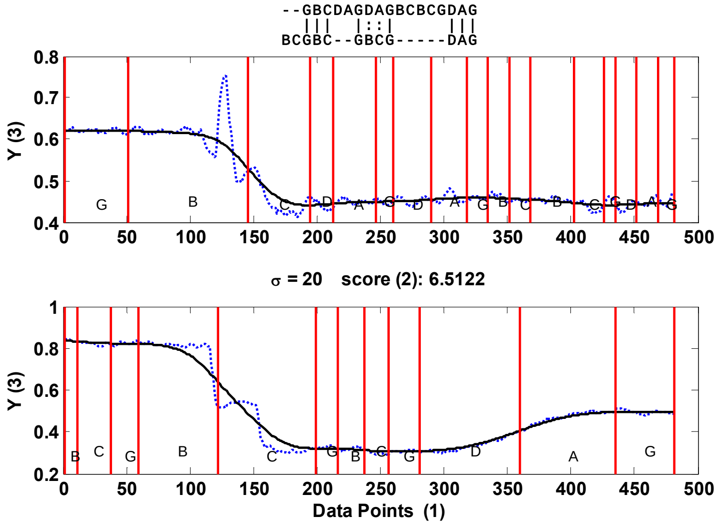
For Gaussian filtering, an overall filtering parameter of $\sigma = 20$ was used. *Figure 3* shows two 1-D projected transition after segmentation, their episode sequence and optimal alignment(‘|’ notes perfect match, ‘:’ mutation) where vertical lines mean episode borders. The parameter dependence and resulting similarity scores for this alignment of two transitions are listed in *Table 1*.

Table 1 concludes that the filtering parameter highly influences the length of the sequences by vanishing more and more feature from the trend thus it changes the similarity between the two trends as well, hence it needs to be tuned carefully and problem specific. Obviously, at very large σ values the two filtered trends will be identical (the more the two trends are similar the smaller σ is sufficient).

After aligning every projected product transition to each other, in other words after calculating the similarity measure for each transition pair, the achieved 31×31 sized scoring matrix can be applied for visualization in a dendrogram as a hierarchical clustering result (*Figure 4*) and in 2-D or 3-D plots with MDS, which projects the transitions into a 2 or 3 dimensional space to show visually the distance between the transitions (*Figure 5*). In all these visualizations, similarity score is transposed into distances of transitions simply by *dist = 1 – Score / 10*.

Figure 3

An example for alignment of two projected (dotted line), filtered (cont. line) and segmented product transition profiles (C5-C6 and C10-C11)



3. ábra: Példa egydimenziós (pontvonal), szűrt (folytonos vonal) szegmentált termékváltási profilok illesztésére (felül C5-C6, alul C10-C11 váltások).

Adatpontok(1), Hasonlósági pontérték(2), Projektálási változó(3)

Table 1

Effect of filtering parameter (σ) to length of triangular sequences and aligning similarity scores

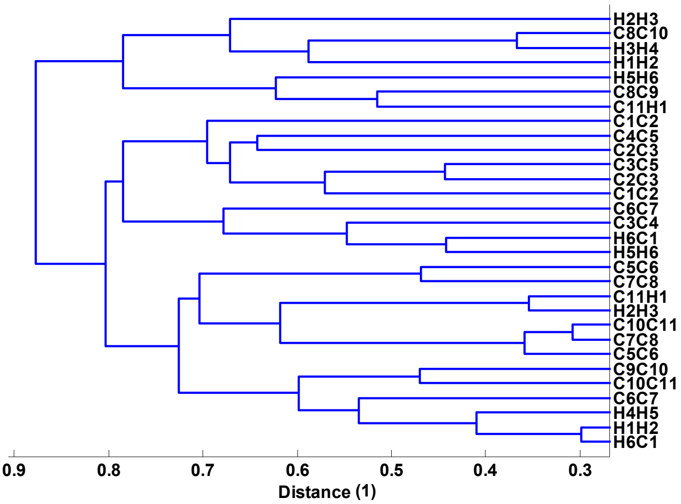
<table>
<tr><th>Transition code (1)</th><th>σ</th><th>Length of sequences (2)</th><th>Aligning score (3)</th><th>Alignment (4)</th></tr>
<tr><td>C5C6</td><td>10</td><td>32</td><td rowspan="2">3.22</td><td rowspan="2">-GBCGBCDBCBCDA----DAGDABCDABCGBCDADAG
 :::||| |||||| ||||| |||||
BCDAGBC-BCBCDABCBCDAGDA--DABCG--------</td></tr>
<tr><td>C10C11</td><td>10</td><td>27</td></tr>
<tr><td>C5C6</td><td>20</td><td>17</td><td rowspan="2">6.51</td><td rowspan="2">--GBCDAGDAGBCBCGDAG
  ||| |::| |||
BCGBC--GBCG-----DAG</td></tr>
<tr><td>C10C11</td><td>20</td><td>12</td></tr>
<tr><td>C5C6</td><td>40</td><td>10</td><td rowspan="2">6.91</td><td rowspan="2">GBCGDAGBCG
 |||| : |
-BCGD-A–G</td></tr>
<tr><td>C10C11</td><td>40</td><td>6</td></tr>
<tr><td>C5C6</td><td>80</td><td>3</td><td rowspan="2">9.9</td><td rowspan="2">BC---G
|| |
BCGDAG</td></tr>
<tr><td>C10C11</td><td>80</td><td>6</td></tr>
<tr><td>C5C6</td><td>115</td><td>3</td><td rowspan="2">10</td><td rowspan="2">BCG
|||
BCG</td></tr>
<tr><td>C10C11</td><td>115</td><td>3</td></tr>
</table>

1. táblázat: A σ szűrési paraméter hatása a szekvenciák hosszára és hasonlósági pontértékre.

Termékváltás kódja(1), Szekvenciahossz (2), Hasonlósági pontérték(3), Illesztés(4)

Figure 4

A dendrogram of product transitions based on their dissimilarity (distance) measure (dist = 1 – Score/10)

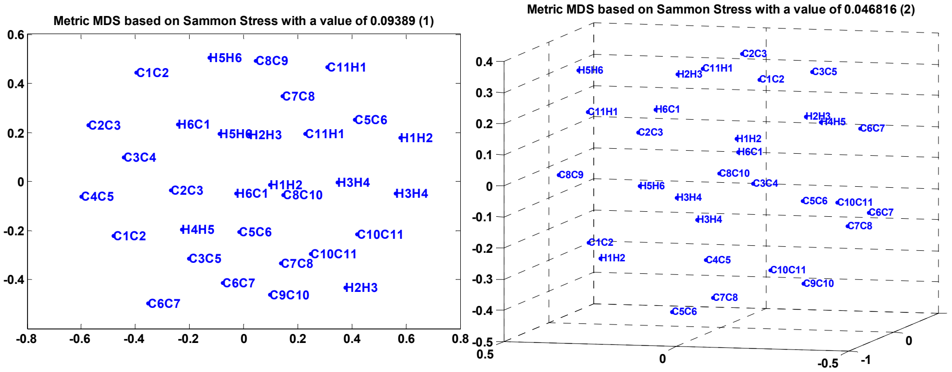


4. ábra: A termékváltások csoportosítása és ábrázolása dendrogrammban a különbözőségi mértékek alapján (távolság = 1-pontérték / 10)

Különbözőségi mérték(1)

Figure 5

2-D and 3-D scaling of product transitions by Sammon MDS methods



5. ábra: 2- és 3-dimenziós termékváltás-ábrázolás MDS segítségével

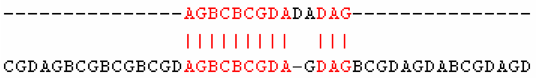
Metrikus 2D-MDS 0,093 értékű (9,3%-os) Sammon-féle hiba alapján(1), Metrikus 3D-MDS 0,046 értékű (4,6%-os) Sammon-féle hiba alapján(2)

The basic expectations for these results were that two main groups could be differentiated: homopolymer-homopolymer and copolymer-copolymer transitions. On the contrary, both visualization techniques show that transitions are currently not managed in a reproducible way, because even the same types are less similar. Analyzis of each transition one-by-one showed that manually driven transition strategies cause these significant differences: 5 process variables are too much degree of freedom to be manually set during a transition, an optimal (e.g. economically optimal) transition strategy would be needed to qualify all the strategies on an objective basis.

Finally, a *motif-discovery problem* solution is presented, as an other possible application of the proposed algorithm. Since, copolymer-copolymer transitions are more similar to each other, a sequence of this transition-type was applied to be a motif and a whole period of copolymer production (451 hrs) was used where the motif (e.g. an optimal transition profile) has to be found by searching for similar subsequences. A result is presented in *Figure 6*, where only the best match was plotted, but by predefining a threshold of accuracy (minimal similarity score) and a larger search window, other less accurate motifs can be found as well.

Figure 6

Finding a similar subsequence (motif) in part of a large episode chain of production



```
---------------AGBCBCGDADADAG---------------
               |||||||||  |||
CGDAGBCGBCGBCGDAGBCBCGDA-GDAGBCGDAGDABCGDAGD
```

6. ábra: Megtalált hasonló szekvencia-mintázat egy kopolimer gyártás hosszú epizódláncának részletében

To use it in an unsupervised way for statistical aims, one has to define only the motif itself and a window length, the algorithm runs every possible window-length-sequence through the whole time series sequence (trivial matches are deleted) to find similar subsequences.

Note, that this time *motif was predefined* because of the engineering case study, it could be an optimal transition strategy, an average of homopolymer-homopolymer transitions (a representative of this transition group) or one typical strategy to see how many times similar strategies occur. This type of application makes the user be able to identify exact locations of e.g. product transitions in a large time horizon.

CONCLUSIONS

The aim of this paper was to extract useful information from quantitative time series in a qualitative way. For decreasing the large amount of data and for resulting in a qualitative description of trends, it applies PCA aided triangular episode segmentation. It has been shown that this tool is able to find similarities in two trends and compare them based on pairwise sequence alignment. Towards this goal, one is able to search for similarities in trends, hence classify process trends based on their shape encoded into episode sequence. As an other type of application, one can find similar subsequences (motifs) in a large production sequence, thus identify product transitions in a trend.

Future work may concentrate on different distance and similarity measures and on defining optimal transition strategies to incorporate minimal subjectivity into the algorithm.

ACKNOWLEDGMENT

The authors would like to acknowledge the support of the Cooperative Research Centre (VIKKK, project III/1) and Hungarian Research Found (OTKA T049534). János Abonyi is grateful for the support of the Bolyai Research Fellowship of the Hungarian Academy of Sciences and the Öveges Fellowship.

REFERENCES

Charbonnier, S., Garcia-Beltan, C., Cadet, C., Gentil S. (2005). Trends extraction and analysis for complex system monitoring and decision support. Engineering Applications of Artificial Intelligence, 18. 21-36.

Cheung, J. T., Stephanopoulos, G. (1990). Representation of process trends. Part I. A formal representation framework. Computers and Chemical Engineering, 14. 495-510.

Faloutsos, C., Ranganathan, M., Manolopoulos, Y. (1994). Fast Subsequence Matching in Time-Series Databases. In: Proceedings of the ACM SIGMOD Int'l Conference on Management of Data. May 24-27, Minneapolis, MN. 419-429.

Fayyad, U. M., Simoudis, E. (1997). Data mining and knowledge discovery. Tutorial Notes at PADD '97 – 1st Int. Conf. Prac. App. KDD & Data Mining, London.

Itakura, F. (1975). Minimum prediction residual applied to speech recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-23. 1. 67-72.

Ku, W., Storer, R. H., Georgakis, C. (1995). Disturbance detection and isolation by dynamic principal component analysis. Chemometrics and Intelligent Laboratory Systems, 30. 179-196.

Lin, J., Keogh, E., Patel, P. Lonardi, S. (2002). Finding Motifs in Time Series. In: Proceedings of the 2nd Workshop on Temporal Data Mining, at the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Edmonton, Alberta, Canada. Jul. 23-26.

Lin, J., Keogh, E., Lonardi, S., Chiu, B. (2003). A Symbolic Representation of Time Series, with Implications for Streaming Algorithms. In proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery. San Diego, CA. Jun. 13.

Needleman, S.B, Wunsch, C.D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. Journal of Molecular Biology, 48. 443-453.

Sakoe, H., Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing, 27. 1. 43-49.

Smith, L.I. (2002). A tutorial on Principal Component Analysis.

Srinivasan, R., Qian, M.S. (2006). Online fault diagnosis and state identification during process transitions using dynamic locus analysis. Chemical Engineering Science, 61. 6109-6132.

Sundarraman, A., Srinivasan, R. (2003). Monitoring transitions in chemical plants using enhanced trend analysis. Computers and Chemical Engineering, 27. 1455-1472.

Venkatasubramanian, V. (1995). A Syntactic Pattern-recognition Approach for Process Monitoring and Fault Diagnosis. Engineering Applications of Artificial Intelligence, 8. 1. 35-51.

Waterman, M.S, Smith, T.F, Beyer W.A. (1976). Some biological sequence metrics. Advanced Mathematics, 20. 376-387.

Waterman, M.S. (1984). General methods of sequence comparison, Bulletin of Mathematical Biology, 46. 473-500.

Wong, J.C., McDonald, K.A. and Palazoglu, A. (1998). Classification of process trends based on fuzzified symbolic representation and hidden Markov models. Journal of Process Control, 8. 5-6. 395-408.

Corresponding author (*Levelezési cím*):

Balaskó Balázs
Pannon University, Department of Process Engineering
H-8200 Veszprém, 10 Egyetem Str.
Pannon Egyetem, Folyamatmérnöki Tanszék
8200 Veszprém, Egyetem u. 10.
Tel.: 36-88-624-770, Fax: 36-88-624-171
e-mail: balaskob@fmt.uni-pannon.hu