In Proc. 13th Int. Conf. on Database Systems for Advanced Applications (DASFAA'08), New Delhi, India, 2008.

Approximate Clustering of Time Series Using Compact Model-based Descriptions

Hans-Peter Kriegel, Peer Kröger, Alexey Pryakhin, Matthias Renz, and Andrew Zherdin

Institute for Computer Science
Ludwig-Maximilians-University of Munich
Oettingenstr. 67, 80538 Munich, Germany
{kriegel,kroegerp,pryakhin,renz,zherdin}@dbs.ifi.lmu.de

Abstract. Clustering time series is usually limited by the fact that the length of the time series has a significantly negative influence on the runtime. On the other hand, approximative clustering applied to existing compressed representations of time series (e.g. obtained through dimensionality reduction) usually suffers from low accuracy. We propose a method for the compression of time series based on mathematical models that explore dependencies between different time series. In particular, each time series is represented by a combination of a set of specific reference time series. The cost of this representation depend only on the number of reference time series rather than on the length of the time series. We show that using only a small number of reference time series yields a rather accurate representation while reducing the storage cost and runtime of clustering algorithms significantly. Our experiments illustrate that these representations can be used to produce an approximate clustering with high accuracy and considerably reduced runtime.

1 Introduction

Clustering time series data is a very important data mining task for a wide variety of application fields including stock marketing, astronomy, environmental analysis, molecular biology, and medical analysis. In such application areas the time series have usually an enormous length which has a significantly negative influence on the runtime of the clustering process. As a consequence, a lot of research work has focused on efficient methods for similarity search in and clustering of time series in the past years.

Time series are sequences of discrete quantitative data assigned to specific moments in time, i.e. a time series X is a sequence of values $X = \langle x_1, \ldots, x_N \rangle$, where x_i is the value at time slot i. This sequence is often also taken as a N-dimensional feature vector, i.e. $X \in \mathrm{R}^N$.

The performance of clustering algorithms for time series data is mainly limited by the cost required to compare pairs of time series (i.e. the processing cost of the used distance function). As indicated above, time series are usually very large containing several thousands of values per sequence. Consequently, the comparison of two time series can be very expensive, particularly when considering the entire sequence of values of the compared objects. The most prominent approaches to measure the similarity of time series are the Euclidean distance and Dynamic Time Warping (DTW). The choice of the distance function mainly depends on the application. In some applications, the Euclidean distance produce better results whereas in other applications, DTW is superior. The big limitation of DTW is its high computational cost of $O(N^2)$ while the Euclidean distance between two time series can be computed in $O(N)$. Since we consider large databases and long time series (i.e. large values of N) in this paper, we focus on the Euclidean distance as similarity function in the following.

In general, if we apply the Euclidean distance to the entire sequences, this is also only adequate for short time series. In case of long time series, we face two problems: The distance computation requires rather high runtimes and, if the time series are indexed by a standard spatial indexing method such as the R-Tree [1] or one of its variants, this index will perform rather bad due to the well-known curse of dimensionality. Thus, the common way is to create adequate but considerably shorter approximations of the data retaining essential features of interest. According to this schema there exist a lot of approaches for dimensionality reduction resulting in suitable time series representations that allow efficient similarity distance computations. However, since the distance computations performed on the approximations do not reflect the exact similarity, they can either be used as a filter step of the data mining task or the preliminary results can be directly taken to approximately solve the problem if the results satisfactorily agree with the exact query response. In the first case, the approximations should fulfill the lower bounding property to guarantee complete results. The advantage of the second solution over the first one is that the approximations do not need to fulfill this lower bounding property which makes it easier to find a proper approximation technique. Furthermore, the second method will yield considerably lower response times because no refinements are required. However, the challenge of the second solution is that the distances on the approximations should accurately estimate the distances on the exact time series in order to achieve satisfying results (i.e. approximate results of high accuracy).

The question at issue is which approximation we should use. Adequate time series approximations can be built by means of mathematical models. Most approaches use models which are based on approximations in time, i.e. models that describe how a time series depends on the time attribute (cf. Section 2). The common characteristics of these techniques are that the approximation quality decreases with increasing length of the time series assuming a constant approximation size. In this paper, we propose a method for the approximation of time series based on mathematical models that explore dependencies between different time series. We represent each time series by an adequate combination of a set of specific reference time series (usually these reference time series can easily be determined e.g. by a domain expert). The resulting representation consists of some low-dimensional feature vector that can easily be indexed by means of any Euclidean index structure. The similarity distance used for the clustering is com-

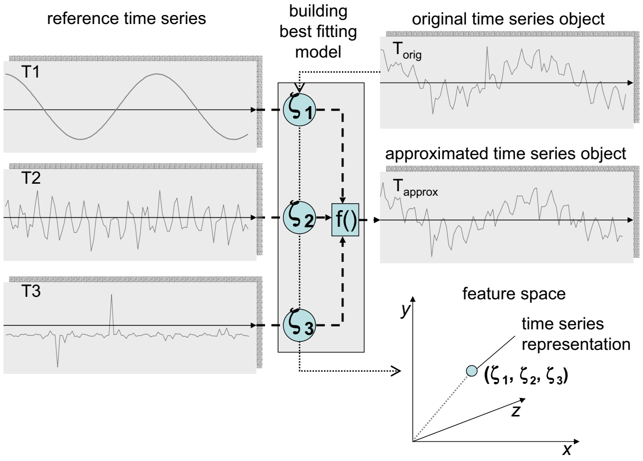

reference time series
building best fitting model
original time series object
T1
T2
T3
T_{orig}
approximated time series object
T_{approx}
ζ1
ζ2
ζ3
f()
feature space
y
z
x
time series representation
(ζ1, ζ2, ζ3)

Fig. 1. Model-based time series representation.

puted by applying the parameters that specify the combination. Consequently, the cost of the clustering process depend only on the number of reference time series rather than on the length of the time series. As we will see in our experiments, the number of reference time series can usually be very small in order to achieve rather accurate results.

Figure 1 illustrates our approach. A set of reference time series (marked as "T1", "T2", and "T3" on the left hand side of Figure 1) is used to approximate an original time series T_{orig}(shown on the upper right hand side of Figure 1) by an arbitrary complex combination T_{approx}. In case of Figure 1 this is a combination of the coefficients $\alpha_1, \ldots, \alpha_3$ representing the three input time series using a function f. The resulting approximated time series (marked as "output" in the middle of the right hand side of Figure 1) is similar to the original time series. For clustering, the approximation is represented by a feature vector of the coefficients of the combination (cf. lower right hand side of Figure 1).

The rest of the paper is organized as follows. In Section 2, we survey related work. In Section 3, we introduce the notion of mathematical models and describe our powerful method for the calculation of compact representation for time series based on the idea of mathematical models. Section 4 presents results of versatile experimental evaluation. Finally, we conclude our paper in Section 5 with a short summary and show directions for further research.

2 Related Work

In general, a time series of length d can be viewed as a feature vector in a d-dimensional space. As discussed above, we focus on similarity in time, i.e. we assume that the similarity of time series is represented by the Euclidean distance of the corresponding feature vector. Since for long time series d is usually large, the efficiency and the effectiveness of data analysis methods is rather limited due to the *curse of dimensionality*. Thus, several more suitable representations of time series data, e.g. by reducing the dimensionality, have been proposed. Most of them are based on the GEMINI indexing approach [2]: extract a few key *features* for each time series and map each time sequence X to a point $f(X)$ in a lower dimensional feature space, such that the distance between X and any other time series Y is always lower-bounded by the Euclidean distance between the two points $f(X)$ and $f(Y)$. For an efficient access, any well known spatial access method can be used to index the feature space. The proposed methods mainly differ in the representation of the time series which can be classified into *non data adaptive methods*, including DFT [3] and extensions [4], DWT [5], PAA [6], and Chebyshev Polynomials [7], as well as *data adaptive* methods, including SVD [8, 9], APCA [10], and cubic splines [11].

Contrary to our approach, all these approximation techniques represent time series by a set of attributes describing how the time series depend on time. As a consequence, the approximation quality of these methods decreases with increasing length of the time series assuming a constant number of approximation attributes.

In [12] the authors use a clipped time series representation rather than applying a dimensionality reduction technique. Each time series is represented by a bit string indicating the intervals where the value of the time series is above the mean value of the time series. This representation can be used to compute an approximate clustering of the time series. The bit level representations are compressed using standard compression algorithms in order to reduce the I/O cost and to speed-up the clustering task. Unfortunately, the authors did not propose any index structure for the approximation data. Each similarity search task results in a full scan over the approximated data.

For clustering time series data, most of the various clustering methods proposed in the past decades have been successfully applied. A general overview over clustering methods is given in [13].

In this paper, we claim the following contribution. We propose a novel compact approximation method for time series data that is independent of the length of the time series. The resulting representation can be indexed using any Euclidean index structure and is rather accurate for an approximate clustering of the database.

3 Mathematical Models for Time Series Data

Mathematical modeling is a powerful method for the description of real-word processes by a compact mathematical representation (e.g., mathematical models of physical or chemical processes). In this section, we introduce a formal definition of mathematical models. Additionally, we describe our method for the description of large time series data by using a compact representation based on the idea of mathematical models.

3.1 Mathematical Model

We start with an informal discussion of the notion of a mathematical model. A mathematical model is an approximate description of a class of certain objects and their relationships. This approximate description is given by mathematical formulas. In context of time series data, a mathematical model describes dependencies between recorded time series data called *inputs* or *exploratory variables* and time series data called *outputs* or *dependant variables* of an observed process. For instance, we can model the relationship between the air pressure in an enclosed container w.r.t. the temperature of the surrounding environment. The observation of both pressure values and temperature values are given in the form of time series. The values of pressure are used as values of the dependant variable. The values of temperature are used as values of the exploratory variable. More formally, a mathematical model can be defined as follows.

Definition 1 (Mathematical Model).
A mathematical model $\mu = (\boldsymbol{X}, \boldsymbol{\alpha}, f)$ for a dependent variable Y (output) consists of a set of exploratory variables *$X_1, \ldots, X_k$ called* inputs *and a* mathematical function *$f(\boldsymbol{X}, \boldsymbol{\alpha})$ that is used to describe the dependency between the variable Y and the variables $X_1, \ldots, X_k$, where $\boldsymbol{\alpha}$ denotes the model parameters also called* coefficients of the model. *The general form of the model is given by $Y = f(\boldsymbol{X}, \boldsymbol{\alpha}) + \varepsilon$, where ε denotes the random error.*

In this definition, the exploratory variables $X_1, \ldots, X_k$ are inputs of the model. The model parameters $\boldsymbol{\alpha}$ are the quantities that are estimated during the modeling process. The value ε represents the random error that makes the relationship between the dependant variable and the exploratory variables a "statistical" one rather than a perfect deterministic one. This statistical character is justified by the fact that the functional relationship holds only in average (i.e., not for each data point).

In general, for building a mathematical model for a time series Y of measured values as a dependant variable we need a mathematical function f and a set $\rho = \{\rho_1, \ldots, \rho_k\}$ of input time series also called *reference time series*. Usually, f and ρ can be given by a domain expert or can be choosen by examining a small sample of the time series in the database. The goal is to find the "best fitting" model. Obviously, in order to find this "best fitting", the random error ε should be minimized. This minimization can be achieved by calculating suitable model parameters $\boldsymbol{\alpha}$. In the last decades, several methods were proposed that allow us to fit the model to the real time series data (i.e. to calculate the model parameters $\boldsymbol{\alpha}$ so that the random error ε is minimized). The most popular method is Least-Squares Estimation which we will use in the following.

Let us consider some examples of mathematical functions that are typically used in mathematical modeling. For a time series Y that fits a straight line with an unknown intercept and slope, there are two parameters $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$, and one exploratory variable X such that $f(\boldsymbol{X}, \boldsymbol{\alpha}) = \alpha_2 \cdot X + \alpha_1$.

Figure 2 illustrates an example for the approximation of a more complex time series $Y = DV$ by a mathematical model using four reference time series

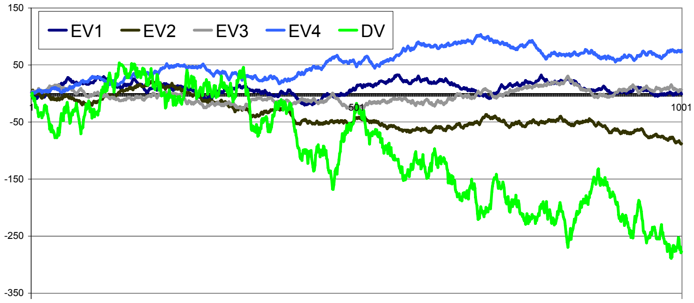


Fig. 2. An example for relationship between a dependant variable (DV) and four exploratory variables (EV1-4).

$\rho_1 = EV1$, $\rho_2 = EV2$, $\rho_3 = EV3$, and $\rho_4 = EV4$ that are combined as given by $DV = EV1 + 2 \cdot EV2 - 4 \cdot EV3 - EV4$.

Thus, the mathematical model describing $Y = DV$ consists of the set of reference time series $\rho = \{\rho_1, \rho_2, \rho_3, \rho_4\}$ and the function $f(\rho, \boldsymbol{\alpha}) = \rho_1 + 2 \cdot \rho_2 - 4 \cdot \rho_3 - \rho_4$ and $\alpha_1 = 1$, $\alpha_2 = 2$, $\alpha_3 = -4$, and $\alpha_4 = -1$.

To summarize, a mathematical model provides an elegant method of describing the relationship between a dependent variable (output time series) and a set of exploratory variables (reference time series). In general, it can use any complex mathematical function such as the combination of quadratical and logarithmical functions to approximate this relationship. In order to express the relationship formally, parameters of a given mathematical function need to be fitted.

3.2 Representation of Time Series based on Mathematical Models

In this section, we introduce the intuition behind our compact representation of a time series and present a novel technique that transforms a very long time series into a compact representation.

Let us consider a given set of reference time series ρ and a given mathematical model $\mu = (\rho, \boldsymbol{\alpha}, f)$. Each time series $T_i \in \mathcal{D}$ in the database can now be considered as a dependant variable Y_i. Values of the dependant variable Y_i can be approximated by values of the mathematical model $\mu_i = (\rho, \boldsymbol{\alpha_i}, f)$ that contains the model parameters $\boldsymbol{\alpha}_i$ that are fitted in order to approximate the values of the dependant variable Y_i as exactly as possible. Thus, the given mathematical model μ_i describes relationships between the reference time series ρ and the approximated time series T_i (i.e., it expresses how strong Y_i depends on each of the reference time series) by means of the model parameters $\boldsymbol{\alpha}_i$. Obviously, dependant variables Y_i and Y_j with similar dependencies should have very similar mathematical models μ_i and μ_j, i.e. the parameters $\boldsymbol{\alpha}_i$ and $\boldsymbol{\alpha}_j$ will be rather similar. In other words, if the underlying physical processes represented by measured values in the database have similar character, their mathematical models look very similar. This relation between original processes and mathematical models is justified by the fact that we consider dependencies based on the same form of the mathematical function and the same reference time series, i.e. all the models μ_i use the same function f and the same set of reference time series ρ but differ only in the parameters $\boldsymbol{\alpha_i}$.

In the following, we describe this intuition more formally:

Definition 2 (Model-based Representation).
Let $\rho = \rho_1, \ldots, \rho_k \subseteq \mathcal{D}$ be a given set of reference time series with $\rho_j = \langle \rho_{j,1}, \ldots, \rho_{j,N} \rangle$ and let $f(\rho, \boldsymbol{\alpha})$ be a given mathematical function. A model-based representation *of a database time series $T_i = \langle t_{i,1}, \ldots, t_{i,N} \rangle \in \mathcal{D}$ is given by a vector of model parameters $\boldsymbol{\alpha}_i$ if $\boldsymbol{\alpha}_i$ minimizes the random error ε of the mathematical model $\mu = (\rho, \boldsymbol{\alpha}, f)$ having the general form $T_i = f(\rho, \boldsymbol{\alpha}_i) + \varepsilon$.*

In the example shown in Figure 2, the model-based representation of the time series DV w.r.t. the reference time series $\rho = \{EV1, EV2, EV3, EV4\}$ is given by a vector $\alpha = (1, 2, -4, -1)$. Let us not that, in this case, we describe a time series of length 1,000 by a short model-based representation with four coefficients.

To summarize, we describe each time series by a small set of model parameters of a mathematical model the shape of which is identical for all time series in the database. The size of our model-based representation is independent on the length of the time series in the underlying database but depends only on the number of reference time series. In particular, the approximation exactness of our model-based representation only depends on the applied model function and the reference time series. Therefore, we can achieve an arbitrary level of exactness of the approximation by choosing a model function and a set of reference time series that are most appropriate for the given application area.

3.3 Model-based Similarity of Time Series

The bottom line of clustering time series is the distance (or similarity) measure used to decide about the similarity of time series. As discussed above, we focus on similarity in time. Thus, for our approach we use the most prominent time-based distance measure for time series, the Euclidean distance. The Euclidean distance is commonly used for the dimensionality reduction techniques mentioned in Section 2.

For long time series, the computation of the Euclidean distance is very expensive. Furthermore the well-known curse of dimensionality limits the efficiency of indexing methods to speed-up similarity queries. For this reason we propose to compute the similarity using the representations based on mathematical models consisting of only a few coefficients (model parameters). We can show that our model-based similarity distance based on the model parameters accurately approximates the Euclidean distance between the original time series. The approximation accuracy mainly depends on how good the model fits to the original time series, i.e. how accurate the model approximates the original time series.

When defining the similarity of time series based on the model parameters, we need to consider that the pairwise similarities between our reference time

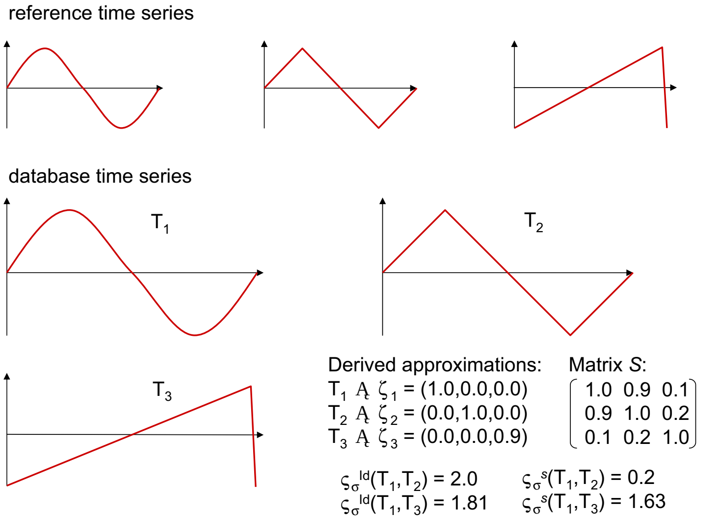

reference time series
database time series
T_1
T_2
T_3
Derived approximations:
T_1 Ą ζ_1 = (1.0,0.0,0.0)
T_2 Ą ζ_2 = (0.0,1.0,0.0)
T_3 Ą ζ_3 = (0.0,0.0,0.9)
Matrix *S*:
$\begin{pmatrix} 1.0 & 0.9 & 0.1 \\ 0.9 & 1.0 & 0.2 \\ 0.1 & 0.2 & 1.0 \end{pmatrix}$
$\varsigma_\sigma^{Id}(T_1,T_2) = 2.0$
$\varsigma_\sigma^{Id}(T_1,T_3) = 1.81$
$\varsigma_\sigma^{s}(T_1,T_2) = 0.2$
$\varsigma_\sigma^{s}(T_1,T_3) = 1.63$

Fig. 3. Motivation for the use of the Mahalanobis-distance.

series need not be identical. An illustrative example is shown in Figure 3. The depicted three time series T_1, T_2, and T_3 are represented by a model μ that is based on the three reference time series presented at the top of Figure 3. Since T_1 is equal to the first reference time series, the coefficients of the model-based representation of T_1 are given by $\alpha_1 = (1.0, 0.0, 0.0)$. Accordingly, the coefficients of the model-based representation of T_2 which is equal to the second reference time series are given by $\alpha_2 = (0.0, 1.0, 0.0)$. The coefficients of the model-based representation of T_3 which is nearly equal to the third reference time series are given by $\alpha_2 = (0.0, 0.0, 0.9)$. If we compare the Euclidean distance between T_1 and T_2 (denoted by $\lambda_\mu^{Id}(T_1, T_2)$ in the Figure) with the Euclidean distance between T_1 and T_3 (denoted by $\lambda_\mu^{Id}(T_1, T_3)$) we see that $\lambda_\mu^{Id}(T_1, T_2) > \lambda_\mu^{Id}(T_1, T_3)$. This is rather unintuitive because the original time series T_1 is much more similar to the original time series T_2 than to T_3. This is because we do not consider that the first reference time series is more similar to the second than to the third.

Thus, we need to consider these different pair-wise similarities of our reference time series when computing the similarity between the model parameters $\boldsymbol{\alpha_i}$ and $\boldsymbol{\alpha_j}$ of two time series $T_i, T_j \in \mathcal{D}$. This can be done using the well-known *Mahalanobis-distance* between the vectors $\boldsymbol{\alpha_i}$ and $\boldsymbol{\alpha_j}$, formally

Definition 3 (Model-based Similarity Distance).
Let $T_i, T_j \in \mathcal{D}$ be two time series and let $\mu = (\rho, \boldsymbol{\alpha}, f)$ be a mathematical model where $\boldsymbol{\alpha_i}$ and $\boldsymbol{\alpha_j}$ are the representations of T_i and T_j based on μ, respectively. The model-based similarity distance *$\lambda_\mu^S(T_i, T_j)$ between T_i and T_j is defined by*

$$\lambda_\mu^S(T_i, T_j) = \sqrt{(\boldsymbol{\alpha_i} - \boldsymbol{\alpha_j}) \cdot S \cdot (\boldsymbol{\alpha_i} - \boldsymbol{\alpha_j})^T}.$$

The key part of the Mahalanobis-distance is the matrix S that is used to rank the pair-wise combinations of the reference time series. Thus, an important issue is to determine a suitable matrix S for distance computation. The following lemma assists in this choice.

Lemma 1. *Let $\mu = (\rho, \boldsymbol{\alpha}, f)$ be a mathematical model and $\boldsymbol{\alpha_i}$ and $\boldsymbol{\alpha_j}$ be the representations of time series T_i and T_j based on μ, respectively. Then, the values of the model-based similarity distances are approximately equal (except for a small random error Δ) to the values of Euclidian distance on the original time series T_i and T_j, i.e. $Dist_{Euclidian}(T_i, T_j) = \lambda_\mu^{\rho \cdot \rho^T}(T_i, T_j) + \Delta$.*

Proof. Without loss of generality, we assume that the function f is linear or is transformable to a linear form, i.e. $T_i = \boldsymbol{\alpha_i} \cdot \rho + \varepsilon_i$ for any $T_i \in \mathcal{D}$.

$$Dist_{Euclidian}(T_i, T_j) = \sqrt{(T_i - T_j) \cdot (T_i - T_j)^T} =$$

$$\sqrt{((\boldsymbol{\alpha_i} \cdot \rho + \varepsilon_i) - (\boldsymbol{\alpha_j} \cdot \rho + \varepsilon_j)) \cdot ((\boldsymbol{\alpha_i} \cdot \rho + \varepsilon_i) - (\boldsymbol{\alpha_j} \cdot \rho + \varepsilon_j))^T} =$$

$$\sqrt{((\boldsymbol{\alpha_i} - \boldsymbol{\alpha_j}) \cdot \rho) \cdot ((\boldsymbol{\alpha_i} - \boldsymbol{\alpha_j}) \cdot \rho)^T + \Delta'} =$$

$$\sqrt{((\boldsymbol{\alpha_i} - \boldsymbol{\alpha_j}) \cdot \rho) \cdot (\rho^T \cdot (\boldsymbol{\alpha_i} - \boldsymbol{\alpha_j})^T) + \Delta'} =$$

$$\sqrt{(\boldsymbol{\alpha_i} - \boldsymbol{\alpha_j}) \cdot (\rho \cdot \rho^T) \cdot (\boldsymbol{\alpha_i} - \boldsymbol{\alpha_j})^T + \Delta'} =$$

$$\lambda_\mu^{\rho \cdot \rho^T}(T_i, T_j) + \Delta$$

The lemma states that the model-based similarity distance approximates the Euclidian distance on the original time series[1] by an error of Δ, if $S = \rho \cdot \rho^T$ where ρ is a matrix consisting of the reference time series. Obviously, Δ depends on the random errors ε_i and ε_j of the model-based approximation of T_i and T_j. Thus, if the errors of the approximation are small (which is a design goal of the approximation and is realized by the Least-Squared Error method), then Δ will also be small. As a consequence, if we set $S = \rho \cdot \rho^T$, the model-based similarity distance will be a rather accurate approximation.

Let us note that if S is the unity matrix, i.e. $S = \rho \cdot \rho^T = Id$, the Mahalanobis-distance is identical to the Euclidean distance. In Figure 3, the model-based similarity distance between T_1 and T_2 as well as between T_1 and T_3 using $S = \rho \cdot \rho^T$ denoted by $\lambda_\mu^S(T_1, T_2)$ and $\lambda_\mu^S(T_1, T_3)$, respectively, are compared with the corresponding Euclidean distance on the model-based representations. As it can be seen, the model-based similarity distance using the Mahalanobis-distance more accurately reflects the intuitive similarity of the original time series than the Euclidean distance on the model-based representations.

Furthermore, let us note that our method may have a slight increase of the CPU cost because we use the Mahalanobis-distance rather than the Euclidean

[1] Please recall that we focus on similarity in time rather than similarity in shape, and, thus use the Euclidean distance as the baseline.

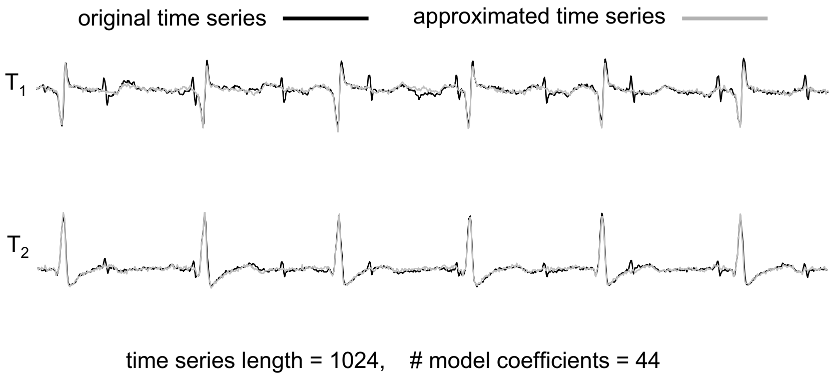


Fig. 4. Approximations for sample time series.

distance used by the existing methods. However, since the size of our approximations is independent of the length of the original data items this marginal CPU performance loss leads to a great benefit in terms of I/O-cost especially when dealing with long time series.

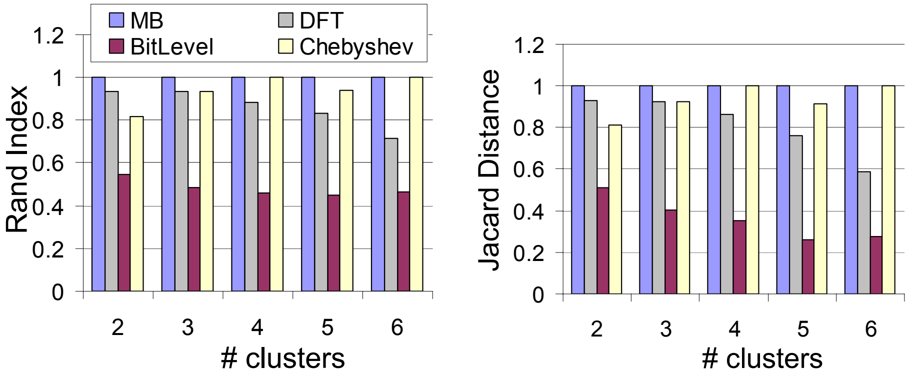
3.4 Choosing the Reference Time Series

Obviously, an important aspect of our model-based representation of time series is the choice of the reference time series. As already sketched, this choice can usually be done by a domain expert. However, if no such domain expert is at hand, we need a procedure for this choice.

In general, the reference time series should have a high correlation to a subset of the remaining time series in the database. Inspired by this intuition, we propose to use the following procedure to derive a set of reference time series. Let us assume that we want to choose k reference time series. We simply cluster the time series using a k-medoid clustering algorithm, e.g. PAM [13]. This yields a set of k cluster medoids (time series), each representing its corresponding cluster. All time series of a cluster are strongly correlated to the corresponding cluster medoid. Obviously, taking these medoids for the derivation of the reference time series should be a very good choice. In addition, in order to avoid wasting to much computational costs, we propose to perform the PAM clustering only on a small sample of the database. In practice, a sample rate of about 1% has shown a sufficient high clustering accuracy.

3.5 Efficient Approximative Clustering

Based on the previously defined similarity distance measure, we can apply any analysis task to time series data. Our main goal is to yield an efficient clustering of the database $\mathcal{D}$ of time series using the approximative representations while generating clusters with sufficient quality. In other words, approximative clustering implements the idea that a user may want very fast response times while



(a)

(b)

Fig. 5. Cluster quality for varying number of clusters. (DS1)

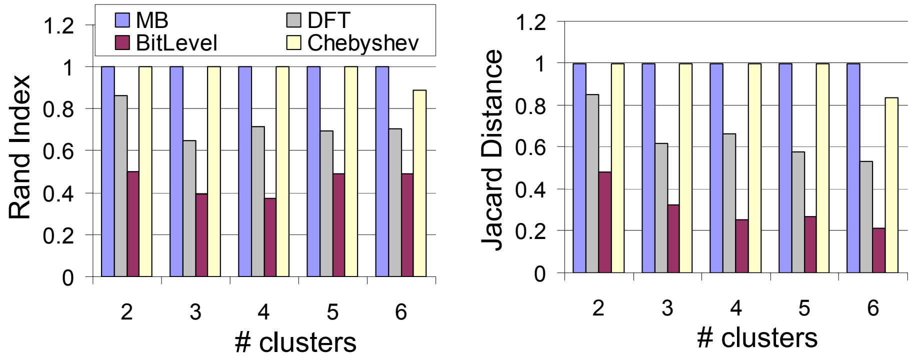
accepting a considerable decrease of accuracy. This is a reasonable setting in many application domains. For our experiments, we used the most prominent clustering method k-means. However, our approximate representation can be integrated into any other clustering algorithm the user is most accustomed to. The key issue for approximate clustering is of course to generate accurate results, i.e. the used approximations should describe the original time series considerably well. We will show in our experiments that a very small set of parameters for the time series approximations is sufficient to achieve high quality clustering results even if the original time series that should be clustered are very long. The example shown in Figure 4 indicates the potentials of our approximation. Two rather complex time series T_1 and T_2 of length 1,024 are compared with some sample approximations using only 44 coefficients. As it can be seen, the compressions approximate the real time series rather accurate.

4 Evaluation

We implemented our method and comparison partners in Java 5. All experiments were performed on a workstation featuring two 3 GHz Xeon CPUs and 32GB RAM. We used four datasets for our experiments, three artificial datasets (DS1, DS2 and DS4) and one real world dataset (DS3) as depicted in Table 4.

The reference time series of the first two artificial datasets DS1 and DS2 are generated by random walk. The corresponding datasets are built by a linear combination of the reference time series compounded by the identity, square, cube and first and second derivatives.

name	type	length N
DS1	artificial	2,560
DS2	artificial	6,000
DS3	real world	1,024
DS4	artificial	2,000-14,000



(a) (b)

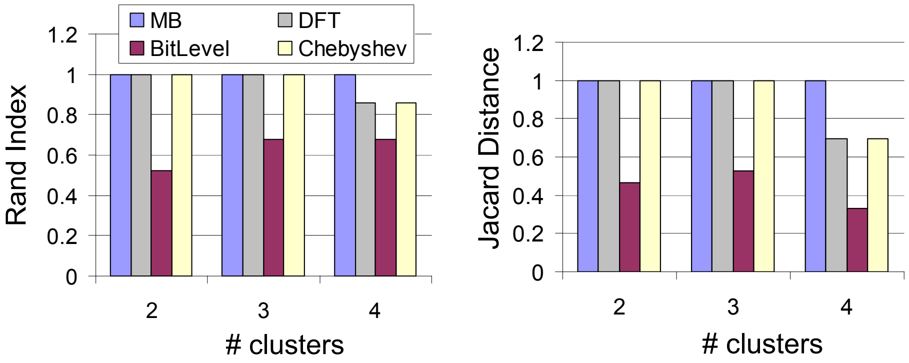
Fig. 6. Cluster quality for varying number of clusters. (DS2)

In order to demonstrate that our approach can handle versatile data, we composed the dataset DS3 in the following way. It consists of real-world time series from the following application areas: (1) wing flutter [2], (2) cutaneous potential recordings of a pregnant woman [3], (3) data from a test setup of an industrial winding process [4], (4) continuous stirred tank reactor [5]. In this dataset (DS3), the reference time series were given by domain experts. The forth dataset DS4 is generated using the Cylinder-Bell-Funnel method[6]. It is an artificial dataset that covers the complete spectrum of stationary/ non-stationary, noisy/ smooth, cyclical/ non-cyclical, symmetric/ asymmetric etc. data characteristics. We used a PAM clustering ($k = 4$) of a random sample in order to derive the reference time series as described above. The k parameter was determined by standard methods.

We compare our mathematical model based time series approximation (MB) with the following competing approximation techniques: *Bit Level* using clipped time series representations as proposed in [12], Discrete Fourier Transformation (DFT) [3] and representations by means of Chebyshev polynomials (Chebyshev) [7]. The competing techniques are evaluated by the approximation quality of k-means clusterings.

Model description of the test datasets. For the mathematical function of the model we used a linear combination of the original set of reference time series,

[2] http://homes.esat.kuleuven.be/ smc/daisy/daisydata.html
[3] http://www.tsi.enst.fr/icacentral/base_single.html
[4] http://homes.esat.kuleuven.be/ smc/daisy/daisydata.html
[5] http://www.fceia.unr.edu.ar/isis/cstr.txt
[6] http://waleed.web.cse.unsw.edu.au/phd/html/node119.html



(a) (b)

Fig. 7. Cluster quality for varying number of clusters. (DS3)

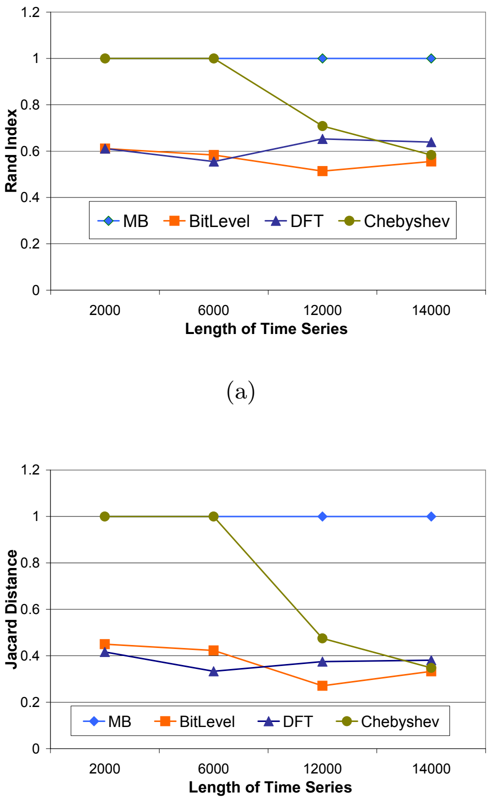
the quadrature and cubature of the reference time series, and the first and second derivation of the reference time series in time. In fact, using n model parameters we only required $n/5$ reference time series.

The overall number of model parameters used for the experiments are justified to the datasets. We used 101 parameters for DS1 and DS2, 51 parameters for DS3 and 20 for DS4. In order to be comparable to the competitors, we used the same number of coefficients for DFT and Chebyshev based approximations.

Measuring clustering quality. For the experimental evaluation of our approach, we built reference clusterings based on the Euclidean distance between the original time series and measured the clustering quality w.r.t. this reference clustering. For the clustering quality measure, we used the two most prominent clustering evaluation methods, the *Rand Index* and the *Jacard Distance* [14].

Experiments on clustering quality. In the first experiment, we examine the quality of our approximation method for a varying number of clusters based on the three datasets DS1 (cf. Figure 5), DS2 (cf. Figure 6) and DS3 (cf. Figure 7). Over all competitors, the *Bit Level* approach yields the lowest clustering quality for all experiments and experimental settings. In comparison we achieve a quality which is at least two times higher than that of the *Bit level* approach. In our experiments, our approach outperforms the method based on DFT coefficients and is even better than the approach using the Chebyshev polynomials when increasing the number of searched clusters.

Against the competitors, our approach achieves optimal clustering quality, even on the real world dataset. This can be justified by the fact that our model-based similarity distance reflects the Euclidean distance on the original time series very accurately.



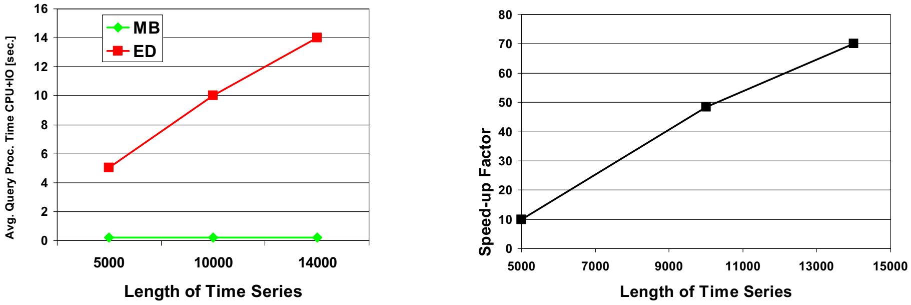
(a)

(b)

Fig. 8. Cluster quality for varying time series length. (DS4)

Dependency on time series length. In the next experiment, we examine how the cluster quality depends on the size of the time series. Figure 8(a) and Figure 8(b) depict the results. Obviously, the characteristic of both dimensionality reduction approaches DFT and Chebyshev is that the clustering quality decreases drastically with increasing time series length. In contrast, we achieve high quality over all investigated time series lengths applying our model-based approximations. Similar to our approach, the Bit Level approach keeps nearly constant quality even for long time series, but yields rather low performance.

Runtime comparison. Last but not least, we compared the speed-up of our method in comparison to the original Euclidean distance in terms of CPU time. For that purpose, we varyed the length of the time series of DS4. The results are illustrated in Figure 9(a). As expected, our model-based approach (marked with “MB” in the figure) scales constant, while the Euclidean distance (marked



(a) (b)

Fig. 9. Performance of our model-based approach vs. Euclidean distance.

with "ED" in the figure) scales linear wr.t. the length of the time series. It can be further observed that our model-based approach clearly outperforms the Euclidean distance for long time series. Figure 9(b) depicts the speed-up factor our model-based approach gains over the approach using the Euclidean distance. Obviously, this speed-up grows with a linear scale when increasing the length of the time series. In summary, our approach yields feature vectors of a constant and considerably lower dimensionality and (beside more efficient indexing) yields to better CPU performance than using the original time series.

5 Conclusions

The performance of clustering time series data is limited by the length of the considered time series. As a consequence, several dimensionality reduction methods have been proposed to represent time series in a compact approximation. Approximative clustering applied to existing compressed representations of time series (e.g. obtained through dimensionality reduction) usually suffers from low accuracy. This holds especially for large time series.

In this paper, we propose an approximation technique for time series based on mathematical models. In fact, each time series is described by the coefficients of a mathematical model involving a given set of reference time series. The great benefit is that the size of our approximation depends only on the number of coefficients of the model (i.e. the number of reference time series). In particular, our method is independent of the length of the original time series and is thus suitable also for very long time series. The compact representation using a feature vector of coefficients of the model allows efficient indexing of the time series approximations for fast similarity search and clustering. We further show how our proposed approximations can be used for approximate clustering. In our experimental evaluation, we illustrate that our novel method outperforms existing state-of-the-art approximation methods in terms of clustering accuracy, i.e. our approximations are significantly better than existing schemata.

In our future work, we plan to extend our ideas of approximating time series by means of mathematical models to stream data.

References

1. Guttman, A.: R-Trees: A dynamic index structure for spatial searching. In: Proceedings of the SIGMOD Conference, Boston, MA. (1984) 47–57
2. Faloutsos, C., Ranganathan, M., Maolopoulos, Y.: "Fast Subsequence Matching in Time-series Databases". In: Proceedings of the SIGMOD Conference, Minneapolis, MN. (1994)
3. Agrawal, R., Faloutsos, C., Swami, A.: "Efficient Similarity Search in Sequence Databases". In: Proc. 4th Conf. on Foundations of Data Organization and Algorithms. (1993)
4. Wichert, S., Fokianos, K., Strimmer, K.: "Identifying Periodically Expressed Transcripts in Microarray Time Series Data". Bioinformatics **20**(1) (2004) 5–20
5. Chan, K., Fu, W.: "Efficient Time Series Matching by Wavelets". In: Proceedings of the 15th International Conference on Data Engineering (ICDE), Sydney, Australia. (1999)
6. Yi, B.K., Faloutsos, C.: "Fast Time Sequence Indexing for Arbitrary Lp Norms". In: Proceedings of the 26th International Conference on Very Large Data Bases (VLDB), Cairo, Egypt. (2000)
7. Cai, Y., Ng, R.: "Index Spatio-Temporal Trajectories with Chebyshev Polynomials". In: Proceedings of the SIGMOD Conference, Paris, France. (2004)
8. Korn, F., Jagadish, H., Faloutsos, C.: "Efficiently Supporting Ad Hoc Queries in Large Datasets of Time Sequences". In: Proceedings of the SIGMOD Conference, Tucson, AZ. (1997)
9. Alter, O., Brown, P., Botstein, D.: "Generalized Singular Value Decomposition for Comparative Analysis of Genome-Scale Expression Data Sets of two Different Organisms". Proc. Natl. Aca. Sci. USA **100** (2003) 3351–3356
10. Keogh, E., Chakrabati, K., Mehrotra, S., Pazzani, M.: "Locally Adaptive Dimensionality Reduction for Indexing Large Time Series Databases". In: Proceedings of the SIGMOD Conference, Santa Barbara, CA. (2001)
11. Bar-Joseph, Z., Gerber, G., Jaakkola, T., Gifford, D., Simon, I.: "Continuous Representations of Time Series Gene Expression Data". J. Comput. Biol. **3-4** (2003) 341–356
12. Ratanamahatana, C.A., Keogh, E., Bagnall, A.J., Lonardi, S.: "A Novel Bit Level Time Series Representation with Implication for Similarity Search and Clustering". In: Proc. 9th Pacific-Asian Int. Conf. on Knowledge Discovery and Data Mining (PAKDD'05), Hanoi, Vietnam. (2005)
13. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Academic Press (2001)
14. Halkidi, M., Batistakis, Y., Vazirgiannis, M.: "On Clustering Validation Techniques". In: Intelligent Information Systems Journal. (2001)