Information Sciences 270 (2014) 28–40

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Safe MBR-transformation in similar sequence matching

Yang-Sae Moon [a,*], Byung Suk Lee [b]

[a] Department of Computer Science, Kangwon National University, 192-1, Hyoja2-Dong, Chunchon, Kangwon 200-701, Republic of Korea
[b] Department of Computer Science, University of Vermont, Burlington, VT 05405, USA

ARTICLE INFO

Article history:
Received 4 September 2008
Received in revised form 2 April 2010
Accepted 21 February 2014
Available online 4 March 2014

Keywords:
Safe MBR-transformation
Lower-dimensional transformation
Similar sequence matching
Data mining

ABSTRACT

When a multidimensional index is used for similar sequence matching, the traditional approach is to transform high-dimensional window sequences to low-dimensional sequences and bounding them into a low-dimensional minimum bounding rectangle (MBR). In this paper, we propose a new approach which constructs a low-dimensional MBR by directly transforming a high-dimensional MBR (called *MBR-transformation*) bounding the high-dimensional sequences. This approach significantly reduces the number of lower-dimensional transformations needed in similar sequence matching. However, it poses a risk that some transformed sequences may fall outside the transformed low-dimensional MBR. We thus propose *safe MBR-transformation* which has the property that every possible transformed sequence is inside a safe MBR-transformed MBR. Then, considering the discrete Fourier transform (DFT) and the discrete Cosine transform (DCT), we prove that they are not safe as MBR-transformations, and modify them to become safe MBR-transformations (called *mbrDFT* if DFT-based and *mbrDCT* if DCT-based). Then, we prove the safeness and optimality of mbrDFT and mbrDCT. Analyses and experiments show that the mbrDFT and mbrDCT reduce the execution time by several orders of magnitude due to the reduction in the number of lower-dimensional transformations. The proposed safe MBR-transformation provides a useful framework for a variety of applications that require a direct transformation of a high-dimensional MBR to a low-dimensional MBR.

1. Introduction

Time-series data are sequences of real numbers representing values at specific time points – examples are stock prices, exchange rates, weather data, financial data, network traffic data, etc. Time-series data stored in a database are called data sequences, and those given by the user are called query sequences. Finding data sequences similar to a given query sequence from the database is called a similar sequence matching [1,8,23] problem. It is a common practice for efficiency's sake to divide each data or query sequence into window sequences and perform matching between each corresponding pair of data and query window sequences [8,13,20,22,23]. Similar sequence matching has been widely used in many practical applications including image matching, handwritten recognition, speech recognition, query by humming, and biological sequence matching [12,14,17,18,26]. Our solution can be used for such practical applications as it improves the overall performance of similar sequence matching.

* Corresponding author. Tel.: +82 33 250 8449; fax: +82 33 250 8440.
E-mail addresses: ysmoon@kangwon.ac.kr (Y.-S. Moon), bslee@cems.uvm.edu (B.S. Lee).

http://dx.doi.org/10.1016/j.ins.2014.02.127
0020-0255/

One approach common to many similar sequence matching techniques is to construct minimum bounding rectangles (MBRs) and use a multidimensional index structure like the R*-tree [4]. MBRs are used to reduce the number of *data* window sequences stored in the index [8,20,25] or the number of *query* window sequences used to search the index [13,22,23]. All these techniques use *lower-dimensional transformation* to reduce high-dimensional sequences to low-dimensional sequences. This transformation is needed to avoid the curse of high dimensionality [5]. Besides, MBRs reduce the required index storage space (if applied to data) or search time (if to queries), since only two diagonal corner points are needed for each MBR instead of all individual points in it.

Thus, in the traditional approach, a low-dimensional MBR is constructed by dividing data or query sequences into window sequences, transforming each (high-dimensional) window sequence to a low-dimensional sequence, and bounding the low-dimensional sequence points into MBRs [8,13,20,22,23,25]. This approach requires as many lower-dimensional transformations as the number of window sequences, which can be very large. For example, in subsequence matching [8,22,23], an MBR contains hundreds or thousands of sequences, and we thus need to execute hundreds or thousands of lower-dimensional transformations to construct only one MBR. Likewise, if this overhead is too high, an alternate approach needs to be sought to reduce the number of lower-dimensional transformations. This is the problem addressed in this paper.

The key idea of our approach is to bound high-dimensional window sequences into a high-dimensional MBR and transform it directly to a low-dimensional MBR. (We call this transformation an *MBR-transformation*.) This obviously reduces the number of lower-dimensional transformations to two per low-dimensional MBR. One caution, however, is that using a sequence-transformation as the MBR-transformation gives no guarantee that all possible window sequences in the high-dimensional MBR are mapped into the low-dimensional MBR. (Details will appear in Section 4.) Fortunately, we have found that such a guarantee can be made with a small relaxation of the transformed MBR boundary. We say an MBR-transformation is *safe* when such a guarantee can be made. This notion of safe MBR-transformation is novel to the best of our knowledge, and our work is the first attempt to propose a practical solution to realize it.

In this paper we develop two kinds of safe MBR-transformations. One is based on the discrete Fourier transform (DFT), and the other is based on the discrete Cosine transform (DCT). Both of these use sinusoidal functions as their transformation functions, and both are widely used as lower-dimensional transformation techniques. For each of them, we prove that using it as the MBR-transformation is not safe, and then propose a safe version, called *mbrDFT* and *mbrDCT*, respectively. We then formally prove they are safe MBR-transformations and also show that each is optimal among all possible MBR-safe transformations of its kind.

We demonstrate the merits of the proposed safe MBR-transformation based approach through running-time analyses and experiments. The experimental results show several orders of magnitude reduction in the number of lower-dimensional transformations and the consequential efficiency improvement over the traditional approach. Another experimental results show that there is hardly any adverse effect from the relaxation of the MBR-transformed MBR in practical cases, as typically it suffices to use only the first one or two dimensions of a low-dimensional MBR [1].

The rest of this paper is organized as follows. Section 2 describes existing work related to similar sequence matching and lower-dimensional transformations. Section 3 defines the safe MBR-transformation and outlines and analyzes the lower-dimensional MBR construction algorithms. Section 4 formally develops the DFT- and DCT-based safe MBR-transformations. Section 5 evaluates their performances through experiments. Section 6 concludes the paper.

2. Related work

We discuss the related work broadly in similar sequence matching and specifically in lower-dimensional transformation.

2.1. Similar sequence matching

A similar sequence matching problem can be classified into a whole matching problem and a subsequence matching problem. The whole matching [1,6,31] is to find data sequences similar to a query sequence, where the lengths of data sequences and the query sequence are the same. The subsequence matching [3,8,13,20,22] is to find subsequences of data sequences that are similar to a query sequence of an arbitrary length. Subsequence matching is more general than whole matching, and has broader applications [8,22]. The use of a low-dimensional MBR has been proposed mostly for subsequence matching, but it can certainly be used for whole matching as well. The MBR-transformation technique proposed in this paper is thus applicable to both whole matching and subsequence matching.

In these similar sequence matching problems, similarity is measured with a distance function $D(X, Y)$. ($X \equiv \{x_0, x_1, \ldots, x_{n-1}\}$ and $Y \equiv \{y_0, y_1, \ldots, y_{n-1}\}$ are two matched sequences of the same length n.) A commonly used distance function is the L_p-distance $\left(= \sqrt[p]{\sum_{i=0}^{n-1} |x_i - y_i|^p}\right)$, which includes the Manhattan distance $(= L_1)$, the Euclidean distance $(= L_2)$, and the maximum distance $(= L_\infty)$ [1,6,8,10,20,22,29]. There are also other distance measures like time warping [2,9,13,17,31] and longest common subsequence (LCSS) [30]. Our MBR-transformation technique does not assume any particular distance measure, and thus can be used with a distance measure of any type.

Some similar sequence matching techniques preprocess the data using various techniques, such as moving average [18,25,29], shifting-and-scaling [7,9], and normalization [20,29], in order to remove distortions (e.g., offset translation, amplitude scaling, linear trend, noise). Preprocessing and low-dimensional MBR construction are independent issues, and thus our MBR transformation technique can be applied to data preprocessed using any kind of preprocessing technique.

2.2. Lower-dimensional transformation

As we mentioned in Section 1, most existing similar sequence matching techniques use lower-dimensional transformation to index high-dimensional (window) sequences using a multidimensional index. The lower-dimensional transformation has first been introduced in the whole matching technique of Agrawal et al. [1], and then widely used in other whole matching techniques [6,18,31] and subsequence matching techniques [8,20,21,23,25]. It has also been used in similar sequence matching on streaming time-series for the dimensionality reduction of the data or query sequences [10,11,21].

A number of similar sequence matching techniques use MBRs to reduce the number of points to be stored in the index or to reduce the number of range queries. For example, techniques in [8,20,25] divide data sequences into windows, transform the windows to low-dimensional points, and then store in an index MBRs containing multiple transformed points. Similarly, techniques in [13,22,25] divide a query sequence into windows, transform the windows to low-dimensional points, and then construct range queries with MBRs containing multiple transformed points. Additionally, the technique in [11] transforms multiple continuous query sequences on streaming time-series to low-dimensional points, and then stores in an index MBRs containing multiple transformed points. All these techniques construct low-dimensional MBRs after transforming individual high-dimensional sequences to low-dimensional sequences. In contrast, our approach transforms a high-dimensional MBR itself to a low-dimensional MBR directly.

Well-known lower-dimensional transformation techniques are based on DFT, DCT, or Wavelet transform. The DFT-based technique has been used most among these three techniques, and has been used mostly in similar sequence matching [8,12,18,27,23–25] on various stored or streaming time-series (e.g., stock prices, weather changes). DCT has been used mainly for compressing multimedia data (e.g., images, videos) [14,32], but recently began to be used for lower-dimensional transformation in similar sequence matching on stored or streaming time-series [14,15] as well. Wavelet transform is used for compressing similar images in [26] and for lower-dimensional transformation of time-series data in [6,22]. In addition, piecewise aggregate approximation (PAA) [13,16] and singular value decomposition (SVD) [16,19] are also introduced as lower-dimensional transformation techniques. All these techniques, however, are for transforming sequences or images. To the best of our knowledge, our approach is the first one applied to MBRs.[1] (In this paper we focus on DFT and DCT as the techniques for MBR transformation. It is unknown whether DWT, PAA, and SVD are suitable for that purpose, nor whether it is feasible; this is a subject for future work.)

3. MBR-transformations: concept and algorithm

3.1. Safe MBR-transformation

It is convenient for the purpose of this paper to distinguish between the transformation of a data or query sequence and the transformation of an MBR. (Both are, after all, transformation of data or query points in a multidimensional space.) We refer to them as a *sequence-transformation* (*seqT*) and an *MBR-transformation* (*mbrT*), respectively. Naturally, an MBR-transformation of an MBR $[L, U]$ is done as two separate MBR-transformations on L and U. Table 1 summarizes the notations used in the paper.

The key technical issue of the problem is to find an MBR-transformation *mbrT* that has the following property: for a given sequence-transformation *seqT*, if a sequence is contained in an MBR, then the MBR transformed using *mbrT* always contains the sequence transformed using *seqT*. The following definition formally defines this property.

Definition 1. For a sequence X and an MBR $[L, U]$ in a multidimensional space, and for a sequence-transformation *seqT* and an MBR-transformation *mbrT*, we say *mbrT* is safe for *seqT* if the following Eq. (1) holds.

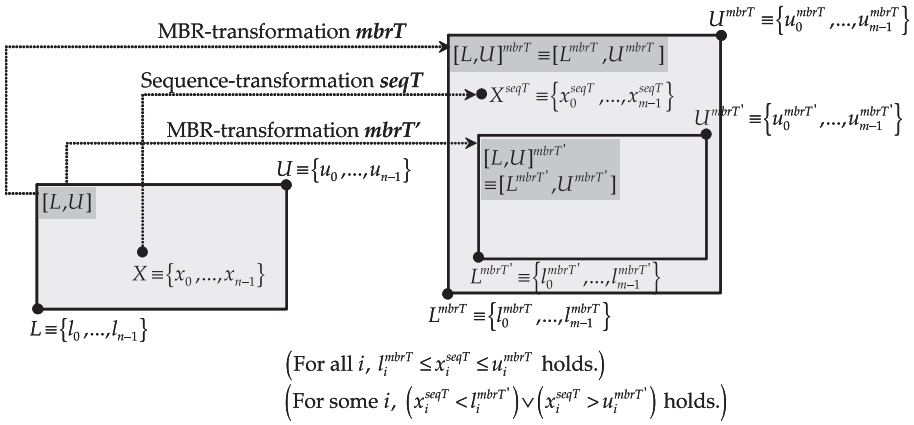
$$X \in [L, U] \rightarrow X^{seqT} \in [L, U]^{mbrT} \quad \square \tag{1}$$

Fig. 1 illustrates the concept of safe MBR-transformation.

[1] An earlier version of this paper has been published in [24], and this paper is an extended version of the previous work. Major changes and extensions are as follows. (1) The notion of "MBR-safe transform" has been changed to "safe MBR-transformation". This new notion separates the transformation of an MBR from the transformation of a data or query sequence. (2) Another type of safe MBR-transformation has been developed for discrete Cosine transform (DCT)-based lower-dimensional transformation, in addition to the discrete Fourier transform (DFT)-based one presented in our previous work. (3) The optimality of the proposed safe MBR-transformations has been proven as Corollaries, for both the DFT-based and DCT-based transformations. (4) Additional experiments have been done using a real data set of stock ticker time-series.

Table 1
Summary of notations.

Notations	Definitions
X	A sequence. $(\equiv \{x_0, x_1, \ldots, x_{n-1}\})$
X^{seqT}	A sequence transformed from X through a sequence-transformation *seqT*. $\left(\equiv \left\{x_0^{seqT}, x_1^{seqT}, \ldots, x_{m-1}^{seqT}\right\}\right)$
$[L, U]$	An MBR whose lower-left and upper-right points are L and U, respectively. $(\equiv [\{l_0, l_1, \ldots, l_{n-1}\}, \{u_0, u_1, \ldots, u_{n-1}\}])$
$[L, U]^{mbrT}$	An MBR transformed from $[L, U]$ through an MBR-transformation *mbrT*. $\left(\equiv [L^{mbrT}, U^{mbrT}] \equiv \left[\left\{l_0^{mbrT}, l_1^{mbrT}, \ldots, l_{m-1}^{mbrT}\right\}, \{u_0^{mbrT}, u_1^{mbrT}, \ldots, u_{m-1}^{mbrT}\}\right]\right)$
$X \in [L, U]$	A predicate that evaluates to TRUE if the sequence X (or, precisely, the point representing X in a multidimensional space) is contained in the MBR $[L, U]$ (i.e., for all i, $l_i \leqslant x_i \leqslant u_i$).



Given the sequence X inside the MBR $[L, U]$, the sequence-transformed X^{seqT} is inside the MBR-transformed $[L, U]^{mbrT}$ but outside the MBR-transformed $[L, U]^{mbrT'}$.

Fig. 1. A safe MBR-transformation (*mbrT*) and a non-safe MBR-transformation (*mbrT′*).

3.2. Low-dimensional MBR construction

The proposed technique which uses this safe MBR-transformation can drastically reduce the number of lower-dimensional transformations, compared with using the traditional technique which constructs an MBR after tens or thousands of lower-dimensional transformations for individual sequences. Algorithm 1 (called *LMBR-seqT*) and Algorithm 2 (called *LMBR-mbrT*) outline the algorithms for constructing low-dimensional MBRs using the traditional sequence-transformation and the proposed MBR-transformation, respectively. (Fig. 2 illustrates how the two algorithms work.) Algorithm 1 transforms each high-dimensional sequence (of the window length n) to a low-dimensional sequence and bounds the resulting low-dimensional sequences into low-dimensional MBRs, with r sequences per MBR. This requires as many transformations as the number of high-dimensional sequences in the time-series data. In contrast, Algorithm 2 bounds high-dimensional sequences into a high-dimensional MBR, one MBR for each r sequences, and transforms each of the resulting MBRs to a low-dimensional MBR. This requires only two transformations for each MBR (one for L and one for U of the MBR $[L, U]$).

Algorithm 1. LMBR-seqT: Sequence-transformation based low-dimensional MBR construction.

Input: l (data or query sequence length), n (window sequence length), r (number of sequences per MBR)
Divide the data or query sequence into window sequences $(X_0, X_1, \ldots X_{p=\lfloor l/n \rfloor})$ of length n each.
for all window sequence $X_i (i = 0, 1, \ldots, p)$ **do**
sequence-transform a high-dimensional sequence X_i to a low-dimensional sequence X_i^{seqT}.
end for
for all set $S_j (j = 0, 1, \ldots, \lfloor p/r \rfloor)$ of r consecutive X_i^{seqT}'s **do**
construct a low-dimensional MBR $[L_j, U_j]$ to bound the low-dimensional sequence X_i^{seqT}'s in S_j.
end for

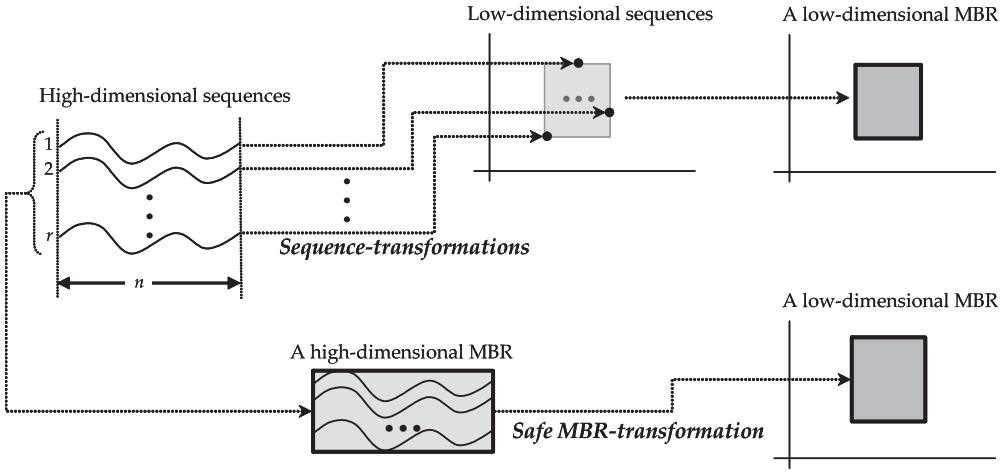


Fig. 2. Comparison of low-dimensional MBR constructions.

Algorithm 2. LMBR-mbrT: MBR-transformation based low-dimensional MBR construction.

Input: l (data or query sequence length), n (window sequence length), r (number of sequences per MBR)
Divide the data or query sequence into window sequences $(X_0, X_1, \ldots X_{p=\lfloor l/n \rfloor})$ of length n each.
for all set $S_j (j = 0, 1, \ldots, \lfloor p/r \rfloor)$ of r consecutive X_i's **do**
construct a high-dimensional MBR $[L_j, U_j]$ to bound the high-dimensional sequence X_i's in S_j.
end for
for all MBR $[L_j, U_j] (j = 0, 1, \ldots, \lfloor p/r \rfloor)$ **do**
MBR-transform a high-dimensional MBR $[L_j, U_j]$ to a low-dimensional MBR $[L_j^{mbrT}, U_j^{mbrT}]$.
end for

The running-time of these algorithms is the summation of MBR construction time and lower-dimensional transformation time. Each MBR construction takes $\Theta(r)$, as it can be done in one scan of the data and, thus, the running-time is proportional to the number of data or query points (i.e., sequences) enclosed in an MBR. As for the time for lower-dimensional transformations, if we denote the time for transforming a sequence of length n as $f(n)$, then Algorithms 1 and 2 take $rf(n)$ and $2f(n)$, respectively, to construct each low-dimensional MBR. The running-time of DFT and DCT is known to be $\Theta(n \log n)$ [28]. Thus, the total running-time of Algorithm 1 is $\Theta(rn \log n) + \Theta(r) = \Theta(rn \log n)$. Similarly, the running-time of Algorithm 2 is $\Theta(r) + \Theta(2n \log n) = \Theta(n \log n)$.

4. Safe MBR-transformations based on DFT and DCT

In this section we propose safe MBR-transformations based on DFT and DCT. Section 4.1 shows that using the original DFT-based lower-dimensional transformation as the MBR-transformation is not safe, and proposes a safe MBR-transformation. Likewise, Section 4.2 shows that using the original DCT-based lower-dimensional transformation as the MBR-transformation is not safe, and proposes a safe MBR-transformation. Due to the similarity (i.e., sinusoidal forms) between DFT – specifically, its real (cosine) part – and DCT, these two sections parallel each other in their presentations.

4.1. mbrDFT: DFT-based safe MBR-transformation

DFT transforms an n-dimensional sequence $X \equiv \{x_0, x_1, \ldots, x_{n-1}\}$ where $x_t (t = 0, 1, \ldots, n-1)$ is a real number to another n-dimensional sequence $Y \equiv \{y_0, y_1, \ldots, y_{n-1}\}$ where $y_i (i = 0, 1, \ldots, n-1)$ is a complex number defined as in Eq. (2) [1,28].

$$y_i = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} x_t e^{-j \cdot \frac{2\pi}{n} it} \text{ for } 0 \leqslant i \leqslant n-1. \tag{2}$$

From Euler's formula [28] and the definition of a complex number, we can rewrite Eqs. (2) and (3) of the real part and imaginary part.

$$y_i = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} x_t \cos\left(-\frac{2\pi}{n} it\right) + j \cdot \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} x_t \sin\left(-\frac{2\pi}{n} it\right) \text{ for } 0 \leqslant i \leqslant n-1. \tag{3}$$

DFT concentrates the energy in the first few coefficients, which means that the other coefficients are relatively negligible. Thus, only a few coefficients in the first few dimensions of Y are used in the *lower-dimensional transformation* [1,8]. The following Definition 2 defines the traditional DFT-based lower-dimensional sequence-transformation.

Definition 2. The *DFT-based lower-dimensional sequence-transformation*, denoted as *seqDFT*, transforms an n-dimensional sequence X to an $m(\ll n)$-dimensional sequence X^{seqDFT} where each coordinate x_i^{seqDFT}, $i = 0, 1, \ldots, m-1$, is computed as

$$x_i^{seqDFT} = \begin{cases} \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} x_t \cos\theta_{it} & \text{if } i \text{ is even,} \\ \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} x_t \sin\theta_{it} & \text{if } i \text{ is odd} \end{cases} \tag{4}$$

where $\theta_{it} = -\frac{2\pi}{n} \lfloor \frac{i}{2} \rfloor t$.

In similar sequence matching, a high-dimensional sequence typically has an order of ten to thousand $(= n)$ dimensions and a low-dimensional sequence has only one to six $(= m)$ dimensions.

If we apply this DFT-based lower-dimensional sequence-transformation as is to transform an n-dimensional MBR $[L, U]$ to an m-dimensional MBR $[L, U]^{seqDFT} \equiv [L^{seqDFT}, U^{seqDFT}]$, then L^{seqDFT} and U^{seqDFT} are computed in the same manner as X^{seqDFT} is computed in Definition 2. That is, for each integer $i \in [0, m-1]$,

$$\begin{cases} l_i^{seqDFT} = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} l_t \cos\theta_{it},\ u_i^{seqDFT} = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} u_t \cos\theta_{it} & \text{if } i \text{ is even,} \\ l_i^{seqDFT} = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} l_t \sin\theta_{it},\ u_i^{seqDFT} = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} u_t \sin\theta_{it} & \text{if } i \text{ is odd.} \end{cases} \tag{5}$$

This lower-dimensional MBR-transformation, however, is not safe for the lower-dimensional sequence-transformation, as shown in the following example.

Example 1. Consider a 4-dimensional sequence $X = \{3.90, 3.70, 4.60, 3.50\}$ and a 4-dimensional MBR $[L, U]$ where $L = \{3.70, 3.50, 4.50, 3.00\}$ and $U = \{4.00, 4.00, 5.00, 4.00\}$. $X \in [L, U]$ holds for these values. The DFT-based lower-dimensional transformation (Definition 2) transforms X to $X^{seqDFT} = \{7.85, -0.35\}$[2] and transforms $[L, U]$ to $[L^{seqDFT}, U^{seqDFT}]$ where $L^{seqDFT} = \{7.35, -0.50\}$ and $U^{seqDFT} = \{8.50, -0.40\}$. From these, we see that $-0.50 \leqslant -0.35 \not\leqslant -0.40$ (i.e., $l_2^{seqDFT} \leqslant x_2^{seqDFT} \not\leqslant u_2^{seqDFT}$), that is, $X^{seqDFT} \in [L, U]^{seqDFT}$ does not hold. □

In order to render the MBR-transformation safe, we need to make sure the resulting MBR contains every possible point that can be transformed from all possible points in the original MBR $[L, U]$. This is achieved by applying to MBR a modified DFT-based lower dimensional transformation, called *mbrDFT*, as defined below.

Definition 3. The *DFT-based lower-dimensional MBR-transformation*, denoted as *mbrDFT*, transforms an n-dimensional MBR $[L, U]$ to an $m(\ll n)$-dimensional MBR $[L, U]^{mbrDFT} (\equiv [L^{mbrDFT}, U^{mbrDFT}])$ where the coordinates l_i^{mbrDFT} and u_i^{mbrDFT}, $i = 0, 1, \ldots, m-1$, are computed as in Eq. (6) for even i and Eq. (7) for odd i.

If i is even,

$$l_i^{mbrDFT} = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} a_t \cos\theta_{it},\ u_i^{mbrDFT} = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} b_t \cos\theta_{it} \text{ where } \begin{cases} a_t = l_t, b_t = u_t & \text{if } \cos\theta_{it} \geqslant 0, \\ a_t = u_t, b_t = l_t & \text{if } \cos\theta_{it} < 0; \end{cases} \tag{6}$$

if i is odd,

$$l_i^{mbrDFT} = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} c_t \sin\theta_{it},\ u_i^{mbrDFT} = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} d_t \sin\theta_{it} \text{ where } \begin{cases} c_t = l_t, d_t = u_t & \text{if } \sin\theta_{it} \geqslant 0, \\ c_t = u_t, d_t = l_t & \text{if } \sin\theta_{it} < 0, \end{cases} \tag{7}$$

where $\theta_{it} = -\frac{2\pi}{n} \lfloor \frac{i}{2} \rfloor t$.

An MBR constructed using mbrDFT (Eqs. (6) or (7)) always contains an MBR that would be constructed using seqDFT (Eq. (5)). The following Theorem 1 shows that mbrDFT is safe.

Theorem 1. *For an n-dimensional sequence X and an n-dimensional MBR $[L, U]$, if $X \in [L, U]$ holds then $X^{seqDFT} \in [L, U]^{mbrDFT}$ holds as well, that is, mbrDFT is a safe MBR-transformation for the DFT-based lower-dimensional sequence-transformation seqDFT.*

Proof. Given that $X \in [L, U]$ holds, that is, $l_t \leqslant x_t \leqslant u_t$ for all $t = 0, 1, \ldots, n-1$, we need to show that $l_i^{mbrDFT} \leqslant x_i^{seqDFT} \leqslant u_i^{mbrDFT}$ holds for all $i = 0, 1, \ldots, m-1$. We prove this for the following two cases: (1) i is an even number and (2) i is an odd number.

Case 1 (i is even): In this case, $x_i^{seqDFT} = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} x_t \cos\theta_{it}$ from Eq. (4). Moreover, if $\cos\theta_{it} \geqslant 0$ then the following three equations hold: $l_i^{mbrDFT} = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} l_t \cos\theta_{it}$ from Eq. (6), $u_i^{mbrDFT} = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} u_t \cos\theta_{it}$ from Eq. (7), and $l_t \cos\theta_{it} \leqslant x_t \cos\theta_{it} \leqslant u_t \cos\theta_{it}$ from the assumption $l_t \leqslant x_t \leqslant u_t$. From these we conclude that $l_i^{mbrDFT} \leqslant x_i^{seqDFT} \leqslant u_i^{mbrDFT}$. On

[2] In the DFT-based lower-dimensional transformation, the imaginary part of the first complex number (i.e., x_1^{seqDFT}) is always 0. Thus, we use $\{x_0^{seqDFT}, x_2^{seqDFT}\}$ instead of $\{x_0^{seqDFT}, x_1^{seqDFT}\}$. The same is true for l_1^{seqDFT} and u_1^{seqDFT} as well.

the other hand, if $\cos\theta_{it} < 0$ then the following three equations hold instead: $l_i^{mbrDFT} = \frac{1}{\sqrt{n}}\sum_{t=0}^{n-1} u_t \cos\theta_{it}$ from Eq. (6), $u_i^{mbrDFT} = \frac{1}{\sqrt{n}}\sum_{t=0}^{n-1} l_t \cos\theta_{it}$ from Eq. (7), and $u_t \cos\theta_{it} \leqslant x_t \cos\theta_{it} \leqslant l_t \cos\theta_{it}$ from the assumption $l_t \leqslant x_t \leqslant u_t$. From these we also conclude that $l_i^{mbrDFT} \leqslant x_i^{seqDFT} \leqslant u_i^{mbrDFT}$.

Case 2 (i is odd): The proof is identical to the proof of Case 2 except for using $\sin\theta_{it}$ instead of $\cos\theta_{it}$. We omit the details here. □

The following example verifies that mbrDFT is a safe MBR-transformation.

Example 2. Consider the same sequence X and MBR $[L, U]$ as in Example 1. Then, the seqDFT-transformed sequence X^{seqDFT} is $\{7.85, -0.35\}$ and the mbrDFT-transformed MBR $[L^{mbrDFT}, U^{mbrDFT}]$ is $[\{7.35, -0.65\}, \{8.50, -0.25\}]$. From these we see that both $7.35 \leqslant 7.85 \leqslant 8.50$ (i.e., $l_0^{mbrDFT} \leqslant x_0^{seqDFT} \leqslant u_0^{mbrDFT}$) and $-0.65 \leqslant -0.35 \leqslant -0.25$ (i.e., $l_2^{mbrDFT} \leqslant x_2^{seqDFT} \leqslant u_2^{mbrDFT}$) hold, that is, $X^{seqDFT} \in [L, U]^{mbrDFT}$ holds. □

The proposed mbrDFT is optimal in that it constructs the smallest MBR among the all possible DFT-based safe MBR-transformations. The following Corollary 1 states it formally.

Corollary 1. *Consider any n-dimensional MBR $[L, U]$ and its mbrDFT-transformed m-dimensional MBR $[L^{mbrDFT}, U^{mbrDFT}]$. If another DFT-based safe MBR-transformation T transforms $[L, U]$ to an m-dimensional MBR $[L^T, U^T]$, then $[L^{mbrDFT}, U^{mbrDFT}]$ is always included in $[L^T, U^T]$.*

Proof (*By contradiction*). Suppose $[L^{mbrDFT}, U^{mbrDFT}]$ is not included in $[L^T, U^T]$. Then, $l_i^{mbrDFT} < l_i^T$ or $u_i^{mbrDFT} > u_i^T$ should be true for some $i = 0, 1, \ldots, m-1$. We will show both cases lead to a contradiction.

Case 1 $\left(l_i^{mbrDFT} < l_i^T\right)$: We distinguish this case further into the cases of an even number i and an odd number i. First, assume an even i. Let $X = \{x_0, x_1, \ldots, x_{n-1}\}$ be a sequence located at one of the corners of the MBR $[L, U]$, where $x_t (t = 0, 1, \ldots, n-1)$ is either l_t (if $\cos\theta_{it} \geqslant 0$) or u_t (if $\cos\theta_{it} < 0$), where $\theta_{it} = -\frac{2\pi}{n}\left\lfloor\frac{i}{2}\right\rfloor t$. Then, by comparing Eqs. (4) and (6), we see that x_i^{seqDFT} and l_i^{mbrDFT} are the same. Furthermore, since T is a safe MBR-transformation, we see that $l_i^T \leqslant x_i^{seqDFT}$ holds and, since $x_i^{seqDFT} = l_i^{mbrDFT}$, $l_i^T \leqslant l_i^{mbrDFT}$ holds as well. This contracts the condition $l_i^{mbrDFT} < l_i^T$ of Case 1. Second, assume an odd i. The proof of this case is identical to the proof of the even i case except for using $\sin\theta_{it}$ instead of $\cos\theta_{it}$. We can prove in the same manner that $l_i^{mbrDFT} < l_i^T$ cannot be true.

Case 2 $(u_i^{mbrDFT} > u_i^T)$: The proof of this case is identical to the proof of Case 1 except for using the upper bounds (u_i^{mbrDFT} and u_i^T) instead of the lower bounds (l_i^{mbrDFT} and l_i^T). We can prove in the same manner as in Case 1 that $u_i^{mbrDFT} > u_i^T$ cannot be true. □

4.2. mbrDCT: DCT-based safe MBR-transformation

DCT is similar to DFT except that the transformed n-dimensional sequence $Y(\equiv \{y_0, y_1, \ldots, y_{n-1}\})$ is expressed as follows [28].

$$y_i = \frac{2 \cdot c(i)}{n}\sum_{t=0}^{n-1} x_t \cos\left(\frac{\pi}{n}\left(t + \frac{1}{2}\right)i\right) \text{ for } 0 \leqslant i \leqslant n-1 \tag{8}$$

where $c(i)$ is expressed as

$$c(i) = \begin{cases} \frac{\sqrt{2}}{2} & \text{if } i = 0, \\ 1 & \text{if } 0 < i \leqslant n-1. \end{cases} \tag{9}$$

Like DFT, DCT concentrates the energy in the first few coefficients and thus, only a few coefficients in the first few dimensions of Y are used in the lower-dimensional transformation. The following Definition 4 defines the traditional DCT-based lower-dimensional sequence-transformation.

Definition 4. The *DCT-based lower-dimensional sequence-transformation*, denoted as *seqDCT*, transforms an n-dimensional sequence X to a new $m(\ll n)$-dimensional sequence X^{seqDCT} where each coordinate x_i^{seqDCT}, $i = 0, 1, \ldots, m-1$, is computed as

$$x_i^{seqDCT} = \frac{2 \cdot c(i)}{n}\sum_{t=0}^{n-1} x_t \cos\theta_{it} \tag{10}$$

where $\theta_{it} = \frac{\pi}{n}(t + \frac{1}{2})i$. □

Like the DFT case, applying the above DCT-based lower-dimensional sequence-transformation to transform an n-dimensional MBR $[L, U]$ to an m-dimensional MBR $[L^{seqDCT}, U^{seqDCT}]$ results in an unsafe MBR-transformation, shown in Eq. (11) below. Example 3 below verifies it.

$$l_i^{seqDCT} = \frac{2 \cdot c(i)}{n} \sum_{t=0}^{n-1} l_t \cos\theta_{it}, \quad u_i^{seqDCT} = \frac{2 \cdot c(i)}{n} \sum_{t=0}^{n-1} u_t \cos\theta_{it}. \tag{11}$$

Example 3. Consider a 4-dimensional sequence $X = \{2.40, 2.40, 2.50, 2.20\}$ and a 4-dimensional MBR $[L, U]$ where $L = \{2.00, 2.20, 2.30, 2.10\}$ and $U = \{2.50, 2.45, 2.60, 2.30\}$. Then, the DCT-based lower-dimensional transformation (Definition 4) transforms X to $X^{seqDCT} = \{3.36, 0.07\}$ and $[L, U]$ to $[L^{seqDCT}, U^{seqDCT}] = [\{3.04, -0.07\}, \{3.48, 0.06\}]$. From these we see that $-0.07 \leqslant 0.07 \not\leqslant 0.06$ (i.e., $l_1^{seqDCT} \leqslant x_1^{seqDCT} \not\leqslant u_1^{seqDCT}$), that is, $X^{seqDCT} \in [L, U]^{seqDCT}$ does not hold. □

The DCT-based safe MBR-transformation, denoted as *mbrDCT*, is defined as follows.

Definition 5. The *DCT-based lower-dimensional MBR-transformation*, denoted as *mbrDCT*, transforms an n-dimensional MBR $[L, U]$ to an $m(\ll n)$-dimensional MBR $[L, U]^{mbrDCT} (\equiv [L^{mbrDCT}, U^{mbrDCT}])$, where the coordinates l_i^{mbrDCT} and u_i^{mbrDCT}, $i = 0, 1, \ldots, m-1$, are computed as

$$l_i^{mbrDCT} = \frac{2 \cdot c(i)}{n} \sum_{t=0}^{n-1} a_t \cos\theta_{it}, \; u_i^{mbrDCT} = \frac{2 \cdot c(i)}{n} \sum_{t=0}^{n-1} b_t \cos\theta_{it} \text{ where } \begin{cases} a_t = l_t, b_t = u_t & \text{if } \cos\theta_{it} \geqslant 0, \\ a_t = u_t, b_t = l_t & \text{if } \cos\theta_{it} < 0 \end{cases} \tag{12}$$

where $\theta_{it} = \frac{\pi}{n}(t + \frac{1}{2})i$.

Like mbrDFT in Definition 3, in order to guarantee safeness of mbrDCT, we deliberately make L^{mbrDCT} and U^{mbrDCT} in Eq. (12) contain every possible point that can be generated from the original MBR $[L, U]$. The following Theorem 2 shows that mbrDCT is a safe MBR-transformation.

Theorem 2. *For an n-dimensional sequence X and an n-dimensional MBR $[L, U]$, if $X \in [L, U]$ holds, then $X^{seqDCT} \in [L, U]^{mbrDCT}$ holds as well, that is, mbrDCT is a safe MBR-transformation for the DCT-based lower-dimensional sequence-transformation seqDCT.*

Proof. To prove $X^{seqDCT} \in [L^{mbrDCT}, U^{mbrDCT}] (= [L, U]^{mbrDCT})$, we need to show that $l_i^{mbrDCT} \leqslant x_i^{seqDCT} \leqslant u_i^{mbrDCT}$ holds for all $i = 0, 1, \ldots, m-1$. Using the same steps as in the proof of Case 1 in Theorem 1, we can easily show that both $\frac{2 \cdot c(i)}{n} \sum_{t=0}^{n-1} a_t \cos\theta_{it} \left(= l_i^{mbrDCT}\right) \leqslant \frac{2 \cdot c(i)}{n} \sum_{t=0}^{n-1} x_t \cos\theta_{it} \left(= x_i^{seqDCT}\right)$ and $\frac{2 \cdot c(i)}{n} \sum_{t=0}^{n-1} b_t \cos\theta_{it} \left(= u_i^{mbrDCT}\right) \geqslant \frac{2 \cdot c(i)}{n} \sum_{t=0}^{n-1} x_t \cos\theta_{it} \left(= x_i^{seqDCT}\right)$ hold, that is, $l_i^{mbrDCT} \leqslant x_i^{seqDCT} \leqslant u_i^{mbrDCT}$ holds, for all $i = 0, 1, \ldots, m-1$. □

An MBR constructed using mbrDCT (Eq. (12)) always contains an MBR that would be constructed using seqDCT (Eq. (11)). The following example verifies that mbrDCT is a safe MBR-transformation.

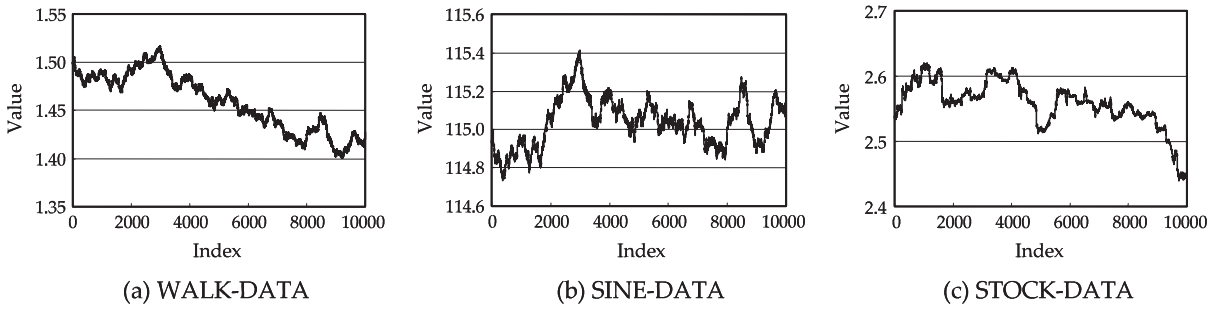
Example 4. Consider the same sequence X and MBR $[L, U]$ as in Example 3. Then, the seqDCT-transformed sequence X^{seqDCT} is $\{3.36, 0.07\}$ and the mbrDCT-transformed MBR $[L^{mbrDCT}, U^{mbrDCT}]$ is $[\{3.04, -0.22\}, \{3.48, 0.21\}]$. From these we see that both $3.04 \leqslant 3.36 \leqslant 3.48$ (i.e., $l_0^{mbrDCT} \leqslant x_0^{seqDCT} \leqslant u_0^{mbrDCT}$) and $-0.22 \leqslant 0.07 \leqslant 0.21$ (i.e., $l_1^{mbrDCT} \leqslant x_1^{seqDCT} \leqslant u_1^{mbrDCT}$) hold, that is, $X^{seqDCT} \in [L, U]^{mbrDCT}$ holds. □

Like mbrDFT, mbrDCT is also optimal among the all possible DCT-based safe MBR-transformations. The following Corollary 2 states it formally.

Corollary 2. *Consider any n-dimensional MBR $[L, U]$ and its mbrDCT-transformed m-dimensional MBR $[L^{mbrDCT}, U^{mbrDCT}]$. If another DCT-based safe MBR-transformation T transforms $[L, U]$ to an m-dimensional MBR $[L^T, U^T]$, then $[L^{mbrDCT}, U^{mbrDCT}]$ is always included in $[L^T, U^T]$.*

Proof (*By contradiction*). Suppose $[L^{mbrDCT}, U^{mbrDCT}]$ is not included in $[L^T, U^T]$. Then, $l_i^{mbrDCT} < l_i^T$ or $u_i^{mbrDCT} > u_i^T$ should be true for some $i = 0, 1, \ldots, m-1$. We show that both cases lead to a contradiction.

Case 1 $\left(l_i^{mbrDCT} < l_i^T\right)$: Let X be a sequence located at one of the corners of the MBR $[L, U]$, where $x_t (t = 0, 1, \ldots, n-1)$ is either l_t (if $\cos\theta_{it} \geqslant 0$) or u_t (if $\cos\theta_{it} < 0$), where $\theta_{it} = \frac{\pi}{n}(t + \frac{1}{2})i$. Then, by comparing Eqs. (10) and (12), we see x_i^{seqDCT} and l_i^{mbrDCT} are the same. Furthermore, since T is a safe MBR-transformation, we see that $l_i^T \leqslant x_i^{seqDCT}$ holds and, since $x_i^{seqDCT} = l_i^{mbrT}$, $l_i^T \leqslant l_i^{mbrDCT}$ holds as well. This contradicts the condition $l_i^{mbrDCT} < l_i^T$.



(a) WALK-DATA (b) SINE-DATA (c) STOCK-DATA

Fig. 3. Part of experimental data sets (10,000 of 1,000,000 entries).

Case 2 ($u_i^{mbrDCT} > u_i^T$): The proof of this case is identical to the proof of Case 1 except for using the upper bounds (u_i^{mbrDCT} and u_i^T) instead of the lower bounds (l_i^{mbrDCT} and l_i^T). We can prove in the same manner as in Case 1 that $u_i^{mbrDCT} > u_i^T$ cannot be true. □.

Our mbrDFT and mbrDCT guarantee the correctness of similar sequence matching, that is, they find all similar sequences correctly in similar sequence matching. In order that a similar sequence matching algorithm guarantees its correctness (i.e., does not incur any false dismissal), it should use the lower-dimensional transformation that satisfies the Parseval's theorem [1,8], which means that the original distance (before applying the transformation) should be no less than the transformed distance (after applying the transformation). Thus, most previous lower-dimensional transformations satisfy the Parseval's theorem. Our seqDFT in Definition 2 and seqDCT in Definition 4 are such examples. Moreover, our mbrDFT and mbrDCT satisfy the Parseval's theorem since they are safe MBR-transformations of seqDFT and seqDCT, respectively, as we presented in Theorems 1 and 2. This means that mbrDFT and mbrDCT do not incur any false dismissal and guarantee the correctness of similar sequence matching.

5. Performance evaluation

We have compared the efficiency of lower-dimensional MBR construction between the traditional sequence-transformation based technique (LMBR-seqT in Algorithm 1) and the proposed MBR-transformation based technique (LMBR-mbrT in Algorithm 2). Additionally, we have compared the total boundary-lengths[3] of the lower-dimensional MBRs resulting from the two techniques; this is to examine the adverse effect of MBR-transformation based techniques' trading the tightness of MBR for the safeness of MBR. The experimental results show that the proposed LMBR-mbrT technique is more efficient than the traditional LMBR-seqT technique by several orders of magnitude, and the resulting low-dimensional MBRs are tight enough for practical use. In this section we first describe the experimental setup in Section 5.1 and then present the results and our observations in Section 5.2.

5.1. Experimental setup

Data sets: What matters on the efficiency of MBR construction is the number of elements in time-series and not the values of elements. The element values, however, make a difference on the boundary-lengths of the constructed MBRs. With this in mind, we have used three types of data sets which determine the element values in different ways.

- *WALK-DATA*: This data set contains a synthetic time-series of one million entries, and is the same data set as used in other works on similar subsequence matching [8,22,23]. The entries are obtained using a random walk process. The first entry (x_0) is set to 1.5, and subsequent entries ($x_1, x_2, \ldots, x_{999999}$) are obtained by adding to the previous entry a random value in the range $[-0.001, 0.001]$, i.e., $x_i = x_{i-1} + \texttt{RANDOM}[-0.001, 0.001]$. Fig. 3(a) shows a part (10,000 entries) of WALK-DATA.
- *SINE-DATA*: This data set contains a synthetic streaming time-series of one million entries, and is similar to those used in other works on continuous similarity matching on streaming time-series [10,11]. The entries are obtained by mixing a sinusoidal fluctuation and a random walk. Specifically, the i-th entry y_i is computed as $y_i = 100\left(\sin 0.1x_i + 1.0 + \frac{i}{1{,}000{,}000}\right)(i = 0, 1, \ldots, 999, 999)$ [10,11], where x_i is the i-th entry of WALK-DATA. Fig. 3(b) shows a part of SINE-DATA.
- *STOCK-DATA*: This data set contains a real stock ticker time-series of 329,112 entries, and is the same data set as used in [8,22,23]. To facilitate a comparison with two synthetic data sets, we have increased the number of entries to one million by repeating the same data set. Fig. 3(c) shows a part of STOCK-DATA.

[3] Using the boundary-length is adequate for the following reason: in similar sequence matching, a range query on a multidimensional index has the form of a regular square and, thus, an MBR (in the index) with a longer boundary-length is more likely to be retrieved as the query result.

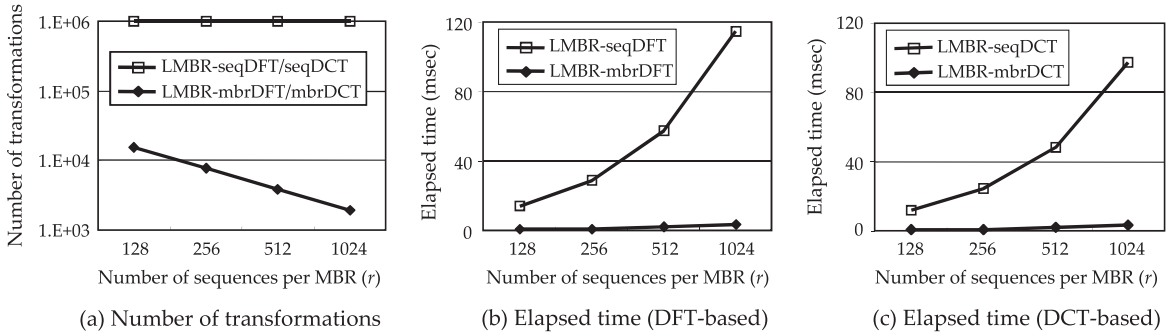


Fig. 4. Efficiency comparison of LMBR-seqT and LMBR-mbrT for varying r ($n = 256$, $m = 2$).

Parameter setting: As seen in Algorithms 1 and 2, there are two key parameters affecting the efficiency of lower-dimensional MBR construction: the length of a window sequence (n) and the number of sequences enclosed in an MBR (r). (The length of a data or query sequence (l) is not relevant to each MBR construction time.) In the experiments, we have picked the values of n and r from the following four numbers: 128, 256, 512, and 1024. Another parameter is the number of dimensions used in a lower-dimensional transformation, which is the length of a low-dimensional sequence (m). For this we use a value in the range of 1 to 4, as in [1]. This means that a 128- to 1024-dimensional sequence is transformed to a 1- to 4-dimensional sequence.

Hardware platform: All experiments have been done on a PC with Intel Pentium IV (2.80 GHz CPU, 512 MB RAM, 70.0 GB hard disk) with GNU/Linux Version 2.6.6 operating system.

5.2. Experimental results

We have performed four sets of experiments to compare the traditional LMBR-seqT and the proposed LMBR-mbrT techniques in terms of their lower-dimensional MBR construction performances. The first and second sets of experiments are to compare the efficiency for varying r with a fixed n, and for varying n with a fixed r, respectively. The third set of experiments is to compare the boundary-length of constructed MBRs. The fourth set of experiments is to compare the actual performance of similar sequence matching that exploits seqDFT, seqDCT, mbrDFT, or mbrDCT. In this subsection we refine the names of the techniques, LMBR-seqT and LMBR-mbrT, with their actual transformation names (i.e., LMBR-seqDFT, LMBR-seqDCT, LMBR-mbrDFT, LMBR-mbrDCT).

5.2.1. Experiment 1: efficiency for varying number of sequences per MBR (r)

Fig. 4(a) shows the number of lower-dimensional transformations for all the three data sets, and Fig. 4(b) and (c) shows the elapsed time per MBR for DFT-based and DCT-based lower-dimensional transformations, respectively, for varying r. We have fixed the value of n to 256 and m to 2. The elapsed time is measured repeatedly over the entire time-series and averaged out to remove the noise. As mentioned in Section 5.1, different data sets do not make any difference to the results of the first and second sets of experiments; we thus show only one result regardless of the data set used.

As shown in Fig. 4(a), our LMBR-mbrDFT and LMBR-mbrDCT significantly reduce the number of transformations over LMBR-seqDFT and LMBR-seqDCT, respectively. This is because LMBR-seqT has to consider all the individual sequences in an MBR while LMBR-mbrT requires only two transformations for an MBR. In particular, as r increases, the number of MBRs to be transformed in LMBR-mbrDFT and LMBR-mbrDCT decreases, and thus their number of transformations also decreases. In Fig. 4(b) and (c) we see that LMBR-mbrDFT and LMBR-mbrDCT reduce the elapsed time over LMBR-seqDFT and LMBR-seqDCT, respectively, by one to two orders of magnitude. The ratio of the elapsed time increases roughly linearly with r, which is consistent with the running-time analysis done in Section 3.2. This confirms that the number of lower-dimensional transformations is the main factor in the performance difference. (The curve appears to swerve upward because of the difference in scales between the vertical and horizontal axes.)

5.2.2. Experiment 2: efficiency for varying window sequence length (n)

Fig. 5(a) shows the number of lower-dimensional transformations, and Fig. 5(b) and (c) shows the elapsed time per MBR for DFT-based and DCT-based lower-dimensional transformations, respectively, for varying n. We have fixed the value of r to 256 and m to 2. As in Experiment 1, we average out the repeated measurements of elapsed time and show only one graph regardless of the data set. From Fig. 5(a), we note that the numbers of transformations do not change even as the length of sequences increases. This is because the numbers are dependent on the number of sequences in LMBR-seqT or the number of MBRs in LMBR-mbrT, but are independent of the length of sequences in both LMBR-seqT and LMBR-mbrT. Additionally, in Fig. 5(b) and (c) we see that LMBR-mbrDFT and LMBR-mbrDCT significantly reduce the elapsed time over LMBR-seqDFT and LMBR-seqDCT, respectively. The ratio of the elapsed time is roughly constant to the value of $\frac{r}{2}\left(=\frac{rn\log n}{2n\log n}\approx 128\right)$ over the varying n; this is consistent with the running-time analysis done in Section 3.2.

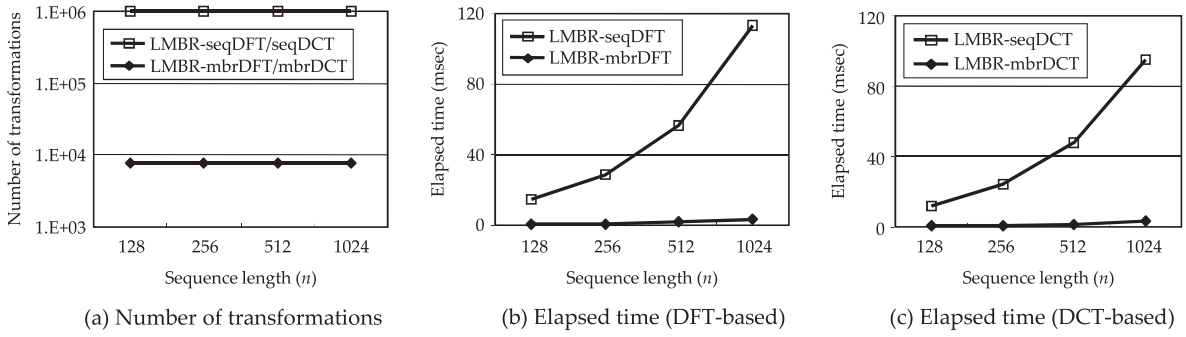


Fig. 5. Efficiency comparison of LMBR-seqT and LMBR-mbrT for varying n ($r = 256,\ m = 2$).

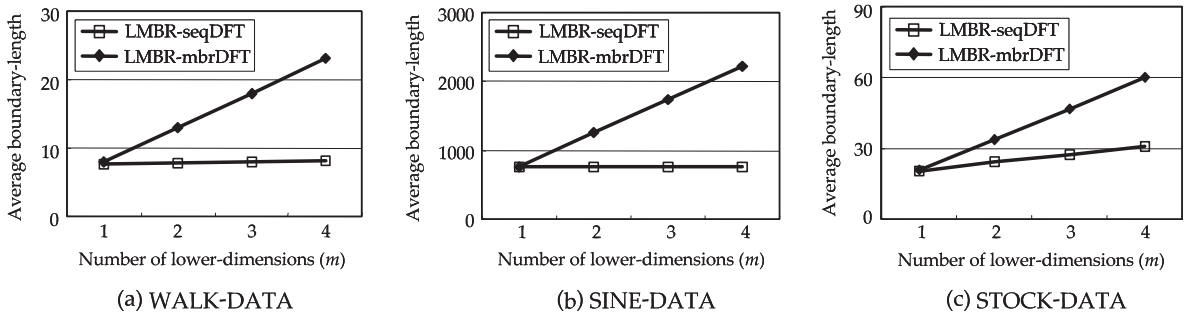


Fig. 6. MBR boundary-length of LMBR-seqDFT and LMBR-mbrDFT.

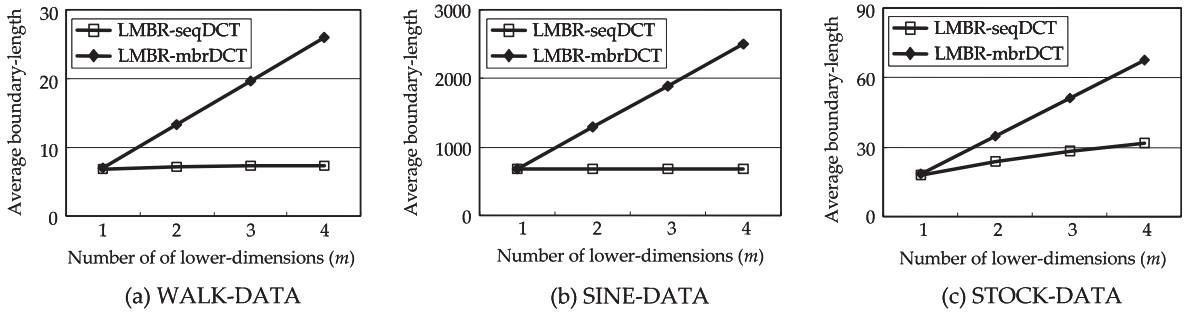


Fig. 7. MBR boundary-length of LMBR-seqDCT and LMBR-mbrDCT.

5.2.3. Experiment 3: MBR boundary-length for varying number of lower-dimensions (m)

Figs. 6 and 7 show the average boundary-lengths of MBRs constructed using DFT-based and DCT-based lower-dimensional transformations, respectively, for varying m (from 1 to 4). We have fixed both r and n to 256. As mentioned in Section 5.1, different data sets give different results in this set of experiments and, therefore, we show the results for all three data sets. In the figures we see that the boundary-length of an MBR from using MBR-transformation is only 0.2% to 2.6% longer than that from using sequence-transformation when m is 1, 38.2% to 65.2% longer when m is 2, 71.8% to 126.8% longer when m is 3, and 94.5% to 190.1% longer when m is 4. Note that it is adequate enough for m to be 1 or 2 in practice, since DFT and DCT concentrate most of the energy in the first dimension [1].

It is interesting that the LMBR-seqDFT slope of the curve in Fig. 6(c) is a bit larger than the slopes in Fig. 6(a) and (b) (and the same for LMBR-seqDCT in Fig. 7). It happens because the fluctuation of data values in the real stock data set is larger than the fluctuations in the synthesis data sets. This larger fluctuation results in larger low-dimensional MBRs as it causes some of the energy to diffuse into other dimensions.

5.2.4. Experiment 4: Similar sequence matching performance for varying the query sequence length

As we explained in Experiment 3, LMBR-mbrDFT and LMBR-mbrDCT increase boundary-lengths of MBR, that is, they decrease the transformation accuracy in constructing MBRs, and this may degrade the overall matching performance. In general, a similar sequence matching algorithm consists of two steps: (1) the index-filtering step and (2) the post-processing step [6,8,23]. Our LMBR-mbrDFT and LMBR-mbrDCT improve the performance of the index-filtering step since they

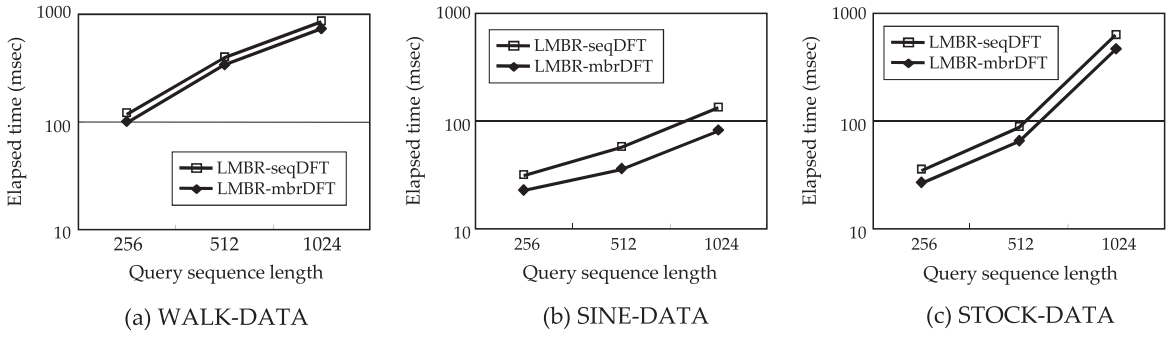


Fig. 8. Subsequence matching performance of LMBR-seqDFT and LMBR-mbrDFT.

significantly reduce the number of transformations. On the other hand, it may not be the case in the post-processing step since larger MBRs retrieve more candidate sequences. Thus, in this experiment we investigate the actual performance of overall similar sequence matching considering both steps. As a similar sequence matching algorithm, we choose DualMatch [13,22], an efficient subsequence matching algorithm. This is because DualMatch uses MBRs at query time, and thus we can easily observe the performance changed by the MBR constructing methods. In the experiment, we set m to 2 and l to 128, where l means the window size in subsequence matching [8,13,22,23]. We generate query sequences from each data set by taking subsequences starting from random offsets [8,22]. To avoid the effects of noise, we experiment with ten different query sequences of the same length and use the average as the result. We set the selectivity [8,22] to 0.01%, that is, assume 0.01% of all possible subsequences are retrieved as similar subsequences.

Fig. 8 shows the subsequence matching performance of LMBR-seqDFT and LMBR-mbrDFT for varying the query sequence length from 256 to 1024. (Since we set the window size to 128, the query sequence length should start from 256 in DualMatch [13,22].) As shown in the figures, LMBR-mbrDFT improves the matching performance compared with LMBR-seqDFT. However, the performance improvement in Fig. 8 is relatively small compared with Figs. 4 and 5. This is because the similar sequence matching algorithm includes the post-processing step as well as the index-filtering step. That is, much time is required in the post-processing step, and thus the overall performance improvement becomes relatively small. In WALK-DATA, we note that the performance difference is much smaller than those in SINE-DATA and STOCK-DATA. We can explain this difference by the characteristics of adjacent entries as follows: adjacent entries of WALK-DATA are very similar [22] and these similar entries cause the candidate set to be large and, accordingly, more effort is required in the post-processing step. We have performed the same experiment for LMBR-seqDCT and LMBR-mbrDCT, but we omit their experimental results here because the results are very similar to those of Fig. 8.

6. Conclusion

We have presented a new approach to constructing low-dimensional MBRs for similar sequence matching. The traditional approach constructs one by bounding low-dimensional sequences transformed from high-dimensional window sequences. In contrast, our proposed approach constructs one by directly transforming a high-dimensional MBR which bounds the high-dimensional window sequences. This drastically reduces the number of required lower-dimensional transformations.

This approach, however, poses a risk that some of the high-dimensional sequences may end up outside the low-dimensional MBR, that is, if the MBR is transformed using the same sequence-transformation. This, thus, brings a need for a new, modified transformation specific to the MBR-transformation (as opposed to the sequence-transformation). We call it the safe MBR-transformation. In this paper we have formally developed safe MBR-transformation based on DFT and DCT (called *mbrDFT* and *mbrDCT*, respectively), and proved that they are optimal among all MBR-transformations of the same kind (i.e., DFT- or DCT-based). We have also conducted experiments and confirmed that our proposed approach is one to two orders of magnitude more efficient than the traditional approach and also showed that enlargement of a resulting low-dimensional MBR is negligible in practice.

From these results we conclude that the proposed safe MBR-transformation will provide a useful framework for a variety of applications that require a direct transformation of a high-dimensional MBR to a low-dimensional MBR. Thus, for the future work we will apply the developed safe MBR-transformation to real applications, such as similarity search, multimedia data retrieval, and geographic information system (GIS). In this paper we focused on DFT and DCT. Developing safe MBR-transformation for other types of transformations, such as wavelet transform, piecewise aggregation approximation (PAA), and singular value decomposition (SVD), would be an interesting and challenging future work.

Acknowledgement

This work was partially supported by Defense Acquisition Program Administration and Agency for Defense Development under the contract.

References

[1] R. Agrawal, C. Faloutsos, A. Swami, Efficient similarity search in sequence databases, in: Proc. of the 4th Int'l Conf. on Foundations of Data Organization and Algorithms, Chicago, Illinois, October 1993, pp. 69–84.
[2] I. Assent, M. Wichterich, R. Krieger, H. Kremer, T. Seidl, Anticipatory DTW for efficient similarity search in time series databases, in: Proc. of the 35th Int'l Conf. on Very Large Data Bases, Lyon, France, August 2009, pp. 826–837.
[3] V. Athitsos, P. Papapetrou, M. Potamias, G. Kollios, D. Gunopulos, Approximate embedding-based subsequence matching of time series, in: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data, Vancouver, Canada, June 2008, pp. 365–378.
[4] N. Beckmann, H.-P. Kriegel, R. Schneider, B. Seeger, The R*-tree: an efficient and robust access method for points and rectangles, in: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data, Atlantic City, New Jersey, May 1990, pp. 322–331.
[5] S. Berchtold, C. Bohm, H.-P. Kriegel, The pyramid-technique: towards breaking the curse of dimensionality, in: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data, Seattle, Washington, June 1998, pp. 142–153.
[6] K.-P. Chan, A.W.-C. Fu, C.T. Yu, Haar wavelets for efficient similarity search of time-series: with and without time warping, IEEE Trans. Knowl. Data Eng. 15 (3) (2003) 686–705.
[7] K.W. Chu, M.H. Wong, Fast time-series searching with scaling and shifting, in: Proc. of the 18th ACM SIGACT-SIGMOD-SIGART Symp. on Principles of Database Systems, Philadelphia, Pennsylvania, June 1999, pp. 237–248.
[8] C. Faloutsos, M. Ranganathan, Y. Manolopoulos, Fast subsequence matching in time-series databases, in: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data, Minneapolis, Minnesota, May 1994, pp. 419–429.
[9] A.W.-C. Fu, E.J. Keogh, L.Y.H. Lau, C. Ratanamahatana, R.C.-W. Wong, Scaling and time warping in time series querying, VLDB J. 17 (4) (2008) 899–921.
[10] L. Gao, X.S. Wang, Continually evaluating similarity-based pattern queries on a streaming time series, in: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data, Madison, Wisconsin, June 2002, pp. 370–381.
[11] L. Gao, Z. Yao, X.S. Wang, Evaluating continuous nearest neighbor queries for streaming time series via pre-fetching, in: Proc. of the 11th ACM Int'l Conf. on Information and Knowledge Management, McLean, Virginia, November 2002, pp. 485–492.
[12] C. Hamzacebi, Improving artificial neural networks' performance in seasonal time series forecasting, Inf. Sci. 178 (23) (2008) 4550–4559.
[13] W.-S. Han, J. Lee, Y.-S. Moon, H. Jiang, Ranked subsequence matching in time-series databases, in: Proc. of the 33rd Int'l Conf. on Very Large Data Bases, Vienna, Austria, September 2007, pp. 423–434.
[14] G.R. Hjaltason, H. Samet, Incremental similarity search in multimedia databases, in: Dept. of Computer Science, University of Maryland, College Park, Technical Report 4199, November 2000.
[15] M.J. Hsieh, M.S. Chen, P.S. Yu, Integrating DCT and DWT for approximating cube streams, in: Proc. of the 14th ACM Int'l Conf. on Information and Knowledge Management, Bremen, Germany, October 2005, pp. 179–186.
[16] E.J. Keogh, K. Chakrabarti, M.J. Pazzani, S. Mehrotra, Dimensionality reduction for fast similarity search in large time series databases, Knowl. Inf. Syst. 3 (3) (2001) 263–286.
[17] E.J. Keogh, L. Wei, X. Xi, S.-H. Lee, M. Vlachos, S.-H. Lee, P. Protopapas, Supporting exact indexing of arbitrarily rotated shapes and periodic time series under euclidean and warping distance measures, VLDB J. 18 (3) (2009) 611–630.
[18] B.-S. Kim, Y.-S. Moon, J. Kim, Noise control boundary image matching using time-series moving average transform, in: Proc. of the 18th Int'l Conf. on Database and Expert Systems Applications, Turin, Italy, September 2008, pp. 362–375.
[19] F. Korn, H.V. Jagadish, C. Faloutsos, Efficiently supporting ad hoc queries in large datasets of time sequences, in: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data, Tucson, Arizona, June 1997, pp. 289–300.
[20] W.-K. Loh, S.-W. Kim, K.-Y. Whang, A subsequence matching algorithm that supports normalization transform in time-series databases, Data Mining Knowl. Discovery 9 (1) (2004) 5–28.
[21] W.-K. Loh, Y.-S. Moon, J. Srivastava, Distortion-free predictive streaming time-series matching, Inf. Sci. 180 (8) (2010) 1458–1476.
[22] Y.-S. Moon, K.-Y. Whang, W.-K. Loh, Duality-based subsequence matching in time-series databases, in: Proc. of the 17th IEEE Int'l Conf. on Data Engineering, Heidelberg, Germany, April 2001, pp. 263–272.
[23] Y.-S. Moon, K.-Y. Whang, W.-S. Han, General match: a subsequence matching method in time-series databases based on generalized windows, in: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data, Madison, Wisconsin, June 2002, pp. 382–393.
[24] Y.-S. Moon, An MBR-safe transform for high-dimensional MBRs in similar sequence matching, in: Proc. of the 12th Int'l Conf. on Database Systems for Advanced Applications, Bangkok, Thailand, April 2007, pp. 79–90.
[25] Y.-S. Moon, J. Kim, Efficient moving average transform-based subsequence matching algorithms in time-series databases, Inf. Sci. 177 (23) (2007) 5415–5431.
[26] A. Natsev, R. Rastogi, K. Shim, WALRUS: a similarity retrieval algorithm for image databases, IEEE Trans. Knowl. Data Eng. 16 (3) (2004) 301–316.
[27] J. Nin, V. Torra, Towards the evaluation of time series protection methods, Inf. Sci. 179 (11) (2009) 1663–1677.
[28] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, Numerical Recipes in C: The Art of Scientific Computing, 2nd ed., Cambridge University Press, 1992.
[29] D. Rafiei, A.O. Mendelzon, Querying time series data based on similarity, IEEE Trans. Knowl. Data Eng. 12 (5) (2000) 675–693.
[30] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, E.J. Keogh, Indexing multi-dimensional time-series with support for multiple distance measures, in: Proc. of the 9th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining, Washington, DC, August 2003, pp. 216–225.
[31] B.-K. Yi, H.V. Jagadish, C. Faloutsos, Efficient retrieval of similar time sequences under time warping, in: Proc. of the 14th IEEE Int'l Conf. on Data Engineering, Orlando, Florida, February 1998, pp. 201–208.
[32] D. Zhao, W. Gao, Y.K. Chan, Morphological representation of DCT coefficients for image compression, IEEE Trans. Circ. Syst. Video Technol. 12 (9) (2002) 819–823.