Peter Clote · Jürg Straubhaar

Symmetric time warping, Boltzmann pair probabilities and functional genomics

Abstract. Given two time series, possibly of different lengths, time warping is a method to construct an optimal alignment obtained by stretching or contracting time intervals. Unlike pairwise alignment of amino acid sequences, classical time warping, originally introduced for speech recognition, is not symmetric in the sense that the time warping distance between two time series is not necessarily equal to the time warping distance of the reversal of the time series. Here we design a new *symmetric* version of time warping, and present a formal proof of symmetry for our algorithm as well as for one of the variants of Aach and Church [1]. We additionally design quadratic time dynamic programming algorithms to compute both the *forward* and *backward* Boltzmann partition functions for symmetric time warping, and hence compute the Boltzmann probability that any two time series points are aligned. In the future, with the availability of increasingly long and accurate time series gene expression data, our algorithm can provide a sense of biological

Department of Biology, Courtesy appt. in Computer Science, Boston College, Chestnut Hill, MA 02467 `clote@bc.edu`

Department of Molecular Medicine, University of Massachusetts Medical School, Worcester, MA 01655 `juerg.straubhaar@umassmed.edu`
Send offprint requests to: Peter Clote

Key words: Time warping, Boltzmann partition function, gene expression data, time series

significance for aligned time points – e.g. our algorithm could be used to provide evidence that expression values of two genes have higher Boltzmann probability (say) in the G1 and S phase than in G2 and M phases. Algorithms, source code and web interface, developed by the first author, are made publicly available via the *Boltzmann Time Warping* web server at `bioinformatics.bc.edu/clotelab/`.

1. Introduction

Functional genomics concerns the algorithmic determination of gene function, pursuant to high-throughput experimental assays, including methods using oligo and cDNA array expression chips. An early contribution to this field was made in 1998 by Eisen et al. [3], who used Pearson correlation coefficient of time series expression data for two genes as a measure of their similarity, and implemented average-linkage cluster analysis (a.k.a. the well-known UPGMA[1] phylogenetic tree construction algorithm, cf. text by Clote and Backofen [2]), to cluster cDNA microarray from *S. cerevisiae*. In 1998 Cho et al. [5] used Affymetrix arrays containing oligonucleotide sequences from 6218 genes of *S. cerevisiae*, and by measuring the logarithm of absolute expression levels at times 0,10,20,...,160 minutes, representing roughly two yeast cell cycles, determined that 416 genes were regulated by cell cycle. More than 25% of the 416 genes were found to be adjacent to other genes induced in the same cell cycle phase. In 1998, using cDNA microarrays (in contrast to oligonucleotide arrays) Spellman et al. [4] extended and refined

---

[1] UPGMA is an acronym for <u>u</u>nweighted <u>p</u>air <u>g</u>roup <u>m</u>ethod with <u>a</u>rithmetic mean.

the work of Cho et al. [5], by identifying 800 genes algorithmically determined to be regulated by cell cycle. Spellman et al. used procedures involving Fourier transform, Pearson correlation coefficient and average-linkage analysis of Eisen et al. Prior to [5,4], only 104 yeast genes had been determined to be regulated by cell cycle, using traditional experimental methods. Later, Cho et al. [6] used Affymetrix oligonucleotide arrays to determine human cell cycle-regulated genes, by measuring the logarithm of the absolute expression levels at times 0,2,4,...,24 hours, representing approximately two cell cycles of *H. sapiens*.

It is common practice to compute the Pearson correlation coefficient between two gene expression time series sequences [4], in order to determine putative functionally related genes. However, this requires that both sequences have the same length; in contrast, for sequences of unequal length, *time warping* can be used, as first noticed by Aach and Church [1]. Time warping is a kind of dynamic programming sequence alignment algorithm, where unlike the algorithm of Needleman and Wunsch [11] for global alignment or of Smith and Waterman [12] for local alignment, the sequences consist of real numbers instead of nucleotide or amino acid symbols. Aach and Church [1] implemented several variants of time warping algorithms first described in 1983 by Kruskal and Liberman [7] in the context of speech recognition. While Aach and Church focused principally on investigating the stability of time warping using the data set of Spellman et al. [4], in this paper, we propose a method to quantify the biological significance of

any two points from time series A and B, which are aligned in an optimal global time warping of series A with series B.

Specifically, we introduce a new symmetric[2] version of time warping, by modifying the time warping algorithm of Kruskal and Liberman [7]. We present a formal proof of symmetry for our algorithm, as well as that for one of the variants of Aach and Church [1].[3]

A technical advantage of any symmetric version of time warping is that it allows an unambiguous computation of the Boltzmann probability that two time series points are aligned in an optimal global time warping of gene expression values, given time series of possibly different lengths – this is analogous to the case of sequence alignment [13,10,9]. Along these lines, we design a dynamic programming algorithm to compute the forward and backward *partition functions* for our version of time warping, thus yielding the Boltzmann probability that any two time series points are aligned. We illustrate our algorithm, using gene expression time series for *S. cerevisiae*

---

[2] By symmetric, we mean that the time warping distance between series A and B equals that between series A^r and B^r, where A^r resp. B^r denotes the reversal of the series from last time point to first time point. While dynamic programming sequence alignment [11] has this property, classical time warping does not.

[3] Aach and Church [1] did not originally notice that the time warping algorithm implemented by their software `genewarp` with flag `-a 2` is symmetric; i.e. `genewarp.exe -i1 input1.txt -i2 input2.txt -a 2 -o out.txt` . Since our proof of symmetry for `genewarp -a 2` is similar to, but notationally different than the proof of symmetry for our new algorithm, for economy of space, we present the proof of symmetry of [1] in the appendix.

[5] and for *H. sapiens* [6], both approximately two cell cycles in duration. While this paper focuses on the new time warping algorithm, the formal proof of symmetry, and how to compute Boltzmann pair probabilities, a sequel will focus on applications, and investigate the relation between time warping, homology and function from the gene ontology (GO) database `www.geneontology.org`.

2. Classical time warping algorithm

Time warping of two sequences $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$, where each $a_i, b_j \in$ $\mathbf{R}^k$ is a k-vector of *features* was introduced by Kruskal and Liberman in the context of speech recognition, where in the 1980's, k was chosen between 6 and 15, and the ith component (or feature) was taken to be the "power present in a speech utterance in the ith frequency band at time t (using a short-time spectral analysis)" [7].[4] Time warping is reminiscent and algorithmically similar, though distinct, to dynamic programming pairwise sequence alignment [2], where *compression* resp. *expansion* are analogous to sequence *deletion* resp. *insertion*. (One can easily imagine this situation when comparing a Texan drawl with a British accent, where in the former vowels may be drawn out while in the latter terminal syllables of words may be inaudible.) In [1], Aach and Church implemented variants of the Kruskal-Liberman algorithms for time warping and interpolated time warping for gene expression data analysis, where in some applications, k was chosen to

---

[4] In this paper, unlike the approach of [1], k will always be taken to be 1; nevertheless, all of our results hold for arbitrary k.

be large (e.g. 450), to accomodate a fixed sequence of genes taken together as features.

Given time intervals $\tau, \mu > 0$, and sequences $a = (a_1, \ldots, a_n)$ and $b = (b_1, \ldots, b_m)$ of elements of $\mathbf{R}^k$, where a_i resp. b_j is the time series value at time $(i-1)\tau$ resp. $(j-1)\mu$, consider functions $u : \{1, 2, \ldots, T\} \to \{1, \ldots, n\}$ and $v : \{1, 2, \ldots, T\} \to \{1, \ldots, m\}$.[5] Functions u, v are said to constitute a (discrete) *time warping* of a, b, provided that:

1. the boundary conditions $u(1) = 1 = v(1)$, $u(T) = n$, $v(T) = m$ hold.

2. u, v are monotonically *increasing*, though not necessarily strictly increasing, i.e. $i \leq j$ implies that $u(i) \leq u(j)$ and $v(i) \leq v(j)$.

3. u, v satisfy the *continuity* condition, i.e. $u(i) \leq u(i + 1) \leq u(i) + 1$, $v(j) \leq v(j + 1) \leq v(j) + 1$.

For example, consider the time warping given in the left panel of Figure 1, where $T = 9$, $n = 7$, $m = 6$, and functions u, v are defined by the Table 1. The right panel of this figure depicts the corresponding *path graph*, which graphs the successive aligned positions $\{(u(i), v(i)) : 1 \leq i \leq T\}$.

Recalling that a_i [resp. b_j] is the time series value at time $(i-1)\tau$ [resp. $(j-1)\mu$] for $1 \leq i \leq n$, $1 \leq j \leq m$, one can identify a_i with b_j if $u(t) = i$

---

[5] The definition we give is for equal interval time series, used in our application. A more general definition of time warping for unequal interval time series can be found in [7] as well as in the algorithm of Aach and Church [1]; the latter is explained in the appendix.

| t	1	2	3	4	5	6	7	8	9
u	1	1	2	3	4	5	5	6	7
v	1	2	3	3	3	4	5	6	6
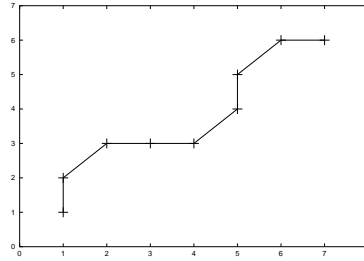
Table 1. (Left) Time warpings $u : \{1, 2, \ldots, T\} \rightarrow \{1, \ldots, n\}$ and $v : \{1, 2, \ldots, T\} \rightarrow \{1, \ldots, m\}$. (Right) Path graph for alignment induced by these time warpings. Figure 1 displays the corresponding trace.

and $v(t) = j$ for some $t \in \{1, \ldots, T\}$. For the example in Table 1, this yields the alignment in Figure 1, which is reminiscent of a *trace* for a pairwise sequence alignment. (See [14, 2] for more on traces and sequence alignment.) The difference between a *time warping* alignment and a *trace* lies in the fact that in a time warping, there are no skipped sequence elements – instead, multiple edges from the same sequence element indicate an expansion (e.g. Texan drawl of a vowel). In a trace, there are no muliple edges from the same element, and instead, skipped sequence elements correspond to gaps (insertions or deletions). Finally, time warping handles sequences of *features*, or elements of $\mathbf{R}^k$, as opposed to sequences of amino acids or nucleotides.

Time warping distance was originally defined for the case of differentiable functions, where u' resp. v' denotes the derivative of u resp. v with respect to time. The discrete analogogue is defined as follows.
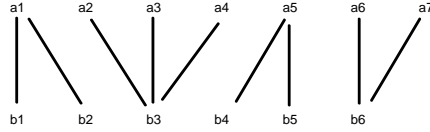
Fig. 1. A time warping of $a_1, \ldots, a_7$ with $b_1, \ldots, b_6$. Here, a_1 is warped against both b_1 and b_2, b_3 is warped against a_2, a_3, a_4, etc. Table 1 displays the time warpings and path graph associated with this trace.

Definition 1 (Kruskal-Liberman [7]). *Suppose that $a = (a_1, \ldots, a_n)$ and $b = (b_1, \ldots, b_m)$ are sequences of time series values in $\mathbf{R}^k$, where a_i resp. b_j is the value at time $(i-1)\tau$ resp $(j-1)\mu$, and let $\rho : \mathbf{R}^k \times \mathbf{R}^k \to \mathbf{R}^+$ be a given metric.[6] If $\mathcal{A}$ is a time warping between a and b given by $u : \{1, \ldots, T\} \to \{1, \ldots, n\}$ and $v : \{1, \ldots, T\} \to \{1, \ldots, m\}$, then the score of $\mathcal{A}$, denoted $S(\mathcal{A})$, is*

$$\sum_{t=1}^{T} \rho\left(a_{u(t)}, b_{v(t)}\right) \frac{(u(t) - u(t-1)) \cdot \tau + (v(t) - v(t-1)) \cdot \mu}{2}.$$

Time warping distance between a and b is the minimum score, over all possible time warpings $u : \{1, \ldots, T\} \to \{1, \ldots, n\}$, $v : \{1, \ldots, T\} \to \{1, \ldots, m\}$ between a and b. More intuitively, the time warping distance is the minimum score over all possible path graphs which satisfy the boundary, monotonicity and continuity conditions. An optimal time warping $\mathcal{A}$ between a and b is a time warping having minimum possible time warping score.

---

[6] Recall that a metric, or distance function, ρ must satisfy the properties: $\rho(a, a) = 0$, reflexivity $\rho(a, b) = \rho(b, a)$, and triangle inequality $\rho(a, c) \leq \rho(a, b) + \rho(b, c)$. For instance, ρ could be Euclidean distance in $\mathbf{R}^k$.

Notice that as in the case of sequence alignment, though time warping distance is uniquely defined, there may be more than one optimal time warping whose score equals the time warping distance.

Following Definition 1, adapted from [7], classic time warping distance is computed by inductively computing $D_{i,j}$ to be the minimum time warping distance between $a_1, \ldots, a_i$ and $b_1, \ldots, b_j$, for all $1 \leq i \leq n$ and $1 \leq j \leq m$. Let $D_{1,1} = 0$, $D_{i+1,1} = D_{i,1} + \frac{\tau}{2}\rho(a_i, b_1)$, $D_{1,j+1} = D_{1,j} + \frac{\mu}{2}\rho(a_1, b_i)$, and inductively define $D_{i,j}$ for $1 < i \leq n$, $1 < j \leq m$ by

$$
D_{i,j} = \min \begin{cases} D_{i-1,j-1} + \frac{\tau+\mu}{2} \cdot \rho(a_i, b_j) \\[2mm] D_{i-1,j} + \frac{\tau}{2} \cdot \rho(a_i, b_j) \\[2mm] D_{i,j-1} + \frac{\mu}{2} \cdot \rho(a_i, b_j) \end{cases}
$$

It follows that $D_{n,m}$ is the optimal time warping distance between sequences a, b. Clearly, the corresponding dynamic programming algorithm requires space and time $O(n \cdot m)$.

As illustrated in the right panel of Figure 1, a *path graph* for a time warping $\mathcal{A}$ of time series $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$ as follows. Place $1, 2, \ldots, n$ along the x-axis and $1, 2, \ldots, m$ along the y-axis. Whenever a_i is aligned in $\mathcal{A}$ with b_j, a point is placed at coordinate (i, j). These coordinates are connected so that *(i)* if a_i is aligned with both b_j, b_{j+1} in $\mathcal{A}$, then there is a vertical line segment from (i, j) to $(i, j+1)$, *(ii)* if a_i, a_{i+1} are both aligned with b_j in $\mathcal{A}$, then there is a horizontal line segment from (i, j) to $(i+1, j)$, *(iii)* if a_i is aligned with b_j and a_{i+1} is aligned with b_{j+1} in $\mathcal{A}$, then there

is a diagonal line segment from (i, j) to $(i+1, j+1)$. A path graph for the time warping produced by Algorithm 1 is given in Figure 4.

$$a_1 \quad - \quad a_2 \quad a_3 \quad a_4 \quad a_5 \quad - \quad a_6 \quad a_7$$
$$b_1 \quad b_2 \quad b_3 \quad - \quad - \quad b_4 \quad b_5 \quad b_6 \quad -$$

Fig. 2. A trace of $a_1, \ldots, a_7$ with $b_1, \ldots, b_6$, analogous to the time warping of Figure 1.

The classical notion of time warping distance, just introduced, is not symmetric, in the sense that time warping distance between $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$ is not necessarily the same as that between $a_n, \ldots, a_1$ and $b_m, \ldots, b_1$.[7] In the context of speech recognition, which was the original context for the development of time warping algorithms [7], symmetric time warping is not reasonable because of the directionality of time in spoken words (ie drawing out syllables). This lack of symmetry in time warping is in contrast with ordinary sequence alignment, a feature used by [10,9] (independently rediscovered by the first author) in order to define the Boltzmann probability $Pr[\mathcal{A}]$ of sequence alignment $\mathcal{A}$, as well as the Boltzmann probability of aligning amino acid a_i with a_j – the latter is given by the partition function of all alignments in which a_i is aligned with a_j divided by the partition function of all alignments. In this paper, by modifying classical time warping, we give an algorithm to compute symmetric time warping distance. After reading a preliminary version of this paper, J. Aach conjec-

---

[7] i.e. time warping distance from left-to-right is not necessarily equal to that from right-to-left.

tured that his algorithm `genewarp` [1] with flag `-a 2` is symmetric as well. Here we prove the symmetry of our algorithm, and for economy of space, we prove Aach's conjecture in the appendix. Additionally, we exploit the symmetry of our algorithm to design a dynamic programming algorithm to compute the forward and backward partition function, which then allows us to compute Boltzmann probabilities for the alignment of one time series point above another time series point. (Although we do not do so, one can similarly implement the partition function for `genewarp` with flag `-a` 2. We believe that time points from time series A and B which have large Boltzmann probability are biologically significant, in the same sense that Vingron and Argos [13] claimed a biological significance for certain regions of protein sequence alignments. Current gene expression data has far too few points to envision this approach to yield biological insights of the form, for instance, that certain yeast genes are better aligned in the G1 and S phase with certain other human genes than in the G2 and M phase of cell cycle. We feel that the design and implementation of our algorithm will provide a resource in the future when gene expression time series data sets are available with a greater number of time points.

3. Symmetric time warping

.

In this section, we describe a new symmetric time warping method in Algorithm 1, for which an inductive proof of symmetry is given. Before giving the pseudocode of the algorithm, we describe the underlying idea

to ensure symmetry. Recall that a time warping algorithm is defined to be symmetric if when given two numerical sequences $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$, time warping distance is the same when computed left to right (using $D_{i,j}$) as when computed right to left (using $E_{i,j}$ defined below). Equivalently time warping distance is the same between $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$ as between $a_n, \ldots, a_1$ and $b_m, \ldots, b_1$. Symmetry clearly holds for sequence alignment distance as computed by the Needleman-Wunsch algorithm [11]; however, symmetry does not hold for the classical time warping algorithm.

Given numerical sequences $a = a_1, \ldots, a_n$ and $b = b_1, \ldots, b_m$, suppose that we have an optimal (partial) time warping of the subsequences $a_1, \ldots, a_i$ and $b_1, \ldots, b_j$, for fixed i, j. To extend the time warping to the next number a_{i+1} and b_{j+1} from each sequence, three cases arise: *(i)* a_{i+1} is aligned with b_{j+1}, *(ii)* a_{i+1} is warped against b_j, *(iii)* b_{j+1} is warped against a_i. In particular, it can happen that the optimal time warping of a and b involves warping (say) b_m against a_{n-1}, leaving the last element a_n which must then be aligned with b_m. In this hypothetical case, $D_{n,m} = D_{n-1,m} + \frac{\tau}{2} \cdot \rho(a_n, b_m)$.

If we now consider time warping the sequences a and b from right to left, the first term, which compares a_n with b_m is given by $\rho(a_n, b_m)$. Compared with the last sentence of the previous paragraph, we are missing a multiplicative factor of $\frac{\tau}{2}$. For this reason, in Algorithm 1, we pad $a_1, \ldots, a_n$ both on the left and on the right to produce $a_0, a_1, \ldots, a_n, a_{n+1}$, where $a_0 = a_1$ and $a_n = a_{n+1}$, and similarly we pad the sequence $b_1, \ldots, b_m$ on the left

and right to produce $b_0, b_1, \ldots, b_m, b_{m+1}$, where $b_0 = b_1$ and $b_m = b_{m+1}$. Additionally, in cases of time expansion and compression, we interpolate values. Thus instead of the classical treatment of time expansion with term $D_{i-1,j} + \frac{\tau}{2} \cdot \rho(a_i, b_j)$, we consider the term $D_{i-1,j} + \frac{\tau}{2} \frac{\rho(a_i,b_j) + \rho(a_i,b_{j+1})}{2}$. Similarly, instead of the classical treatment of time compression with term $D_{i,j-1} + \frac{\mu}{2} \cdot \rho(a_i, b_j)$, we consider the term $D_{i,j-1} + \frac{\mu}{2} \frac{\rho(a_i,b_j) + \rho(a_{i+1},b_j)}{2}$, which can be considered an attempt to interpolate or smooth out the transition. Our algorithm follows.

Define $D_{0,0} = 0$ and for $1 \le i \le n$ and $1 \le j \le m$, define

$$D_{i,0} = D_{i-1,0} + \frac{\tau}{2} \cdot \rho(a_i, b_0) \tag{1}$$

and

$$D_{0,j} = D_{0,j-1} + \frac{\mu}{2} \cdot \rho(a_0, b_j). \tag{2}$$

For $1 \le i \le n$ and $1 \le j \le m$, define

$$D_{i,j} = \min \begin{cases} D_{i-1,j-1} + \frac{\tau+\mu}{2}\rho(a_i, b_j) \\ D_{i-1,j} + \frac{\tau}{2}\frac{\rho(a_i,b_j) + \rho(a_i,b_{j+1})}{2} \\ D_{i,j-1} + \frac{\mu}{2}\frac{\rho(a_i,b_j) + \rho(a_{i+1},b_j)}{2}. \end{cases} \tag{3}$$

Define $E_{n+1,m+1} = 0$ and for $1 \le i \le n$ and $1 \le j \le m$, define

$$E_{i,m+1} = E_{i+1,m+1} + \frac{\tau}{2}\rho(a_i, b_{m+1}) \tag{4}$$

and

$$E_{n+1,j} = E_{n+1,j+1} + \frac{\mu}{2}\rho(a_{n+1}, b_j) \tag{5}$$

and for $0 \le i < n$ and $0 \le j < m$,

$$
E_{i,j} = \min \begin{cases} E_{i+1,j+1} + \frac{\tau + \mu}{2} \cdot \rho(a_i, b_j) \\[2mm] E_{i+1,j} + \frac{\tau}{2} \cdot \frac{\rho(a_i,b_j) + \rho(a_i,b_{j-1})}{2} \\[2mm] E_{i,j+1} + \frac{\mu}{2} \cdot \frac{\rho(a_i,b_j) + \rho(a_{i-1},b_j)}{2} \end{cases} \tag{6}
$$

Abstractly considered, any time warping algorithm finds the optimal path graph with points $\{(a_{u(1)}, b_{v(1)}), \ldots, (a_{u(T)}, b_{v(T)})\}$ which minimizes the score $\sum_{i=1}^{T} w_i \cdot \rho(a_{u(i)}, b_{v(i)})$. Variants of time warping may handle base cases differently and may consider different values for w_i. Compared with classic time warping, the Algorithm 1 initially pads each given sequence a, b both on the left and right, assigns the same weight as in classical time warping for diagonal transitions in the path graph, and assigns different interpolated values for horizontal and vertical transitions in the path graph. The time warping distance for Algorithm 1 is the minimum score, using these new weights w_i, over all path graphs which align the padded sequences a, b.

We summarize the above in the following pseudocode for inputs $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$.

Algorithm 1 (Forward time warping)

$D_{0,0} = 0$; $a_0 = a_1$; $b_0 = b_1$; $a_{n+1} = a_n$; $b_{m+1} = b_m$

```
for i = 1 to n
```
$\quad D_{i,0} = D_{i-1,0} + \frac{\tau}{2} \cdot \rho(a_i, b_0)$
```
for j = 1 to m
```
$\quad D_{0,j} = D_{0,j-1} + \frac{\mu}{2} \cdot \rho(a_0, b_j)$

```
for i = 1 to n

  for j = 1 to m
```

$$case_1 = D_{i-1,j-1} + \frac{\tau+\mu}{2} \cdot \rho(a_i, b_j)$$

$$case_2 = D_{i-1,j} + \frac{\tau}{2} \cdot \frac{\rho(a_i,b_j)+\rho(a_i,b_{j+1})}{2}$$

$$case_3 = D_{i,j-1} + \frac{\mu}{2} \cdot \frac{\rho(a_i,b_j)+\rho(a_{i+1},b_j)}{2}$$

$$D_{i,j} = \min\{case_1, case_2, case_3\}$$

Algorithm 2 (Backward time warping)

$$E_{n+1,m+1} = 0; \quad a_0 = a_1; \quad b_0 = b_1; \quad a_{n+1} = a_n; \quad b_{m+1} = b_m$$

```
for i = n down to 1
```

$$E_{i,m+1} = E_{i+1,m+1} + \frac{\tau}{2} \cdot \rho(a_i, b_{m+1})$$

```
for j = m down to 1
```

$$E_{n+1,j} = E_{n+1,j+1} + \frac{\mu}{2} \cdot \rho(a_{n+1}, b_j)$$

```
for i = n down to 1

  for j = m down to 1
```

$$case_1 = E_{i+1,j+1} + \frac{\tau+\mu}{2} \cdot \rho(a_i, b_j)$$

$$case_2 = E_{i+1,j} + \frac{\tau}{2} \cdot \frac{\rho(a_i,b_j)+\rho(a_i,b_{j-1})}{2}$$

$$case_3 = E_{i,j+1} + \frac{\mu}{2} \cdot \frac{\rho(a_i,b_j)+\rho(a_{i-1},b_j)}{2}$$

$$E_{i,j} = \min\{case_1, case_2, case_3\}$$

The optimal forward [resp. reverse] time warping distance between $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$ is $D_{n,m}$ [resp. $E_{1,1}$]. Algorithm 1 [resp. Algorithm 2] computes the optimal forward [resp. reverse] time warping distances, but not the optimal time warping itself. For this, analogous to the construction of an optimal sequence alignment one must use tracebacks, proceeding from $D_{n,m}$ back to $D_{0,0}$ [resp. $E_{1,1}$ to $E_{n+1,m+1}$]. See the text by Clote and Backofen [2] for more details on tracebacks for sequence alignment.

CLAIM A0: For $0 \le i \le n$ and $0 \le j \le m$,

$$D_{i,m} + E_{i+1,m+1} \ge D_{n,m}$$

$$D_{n,j} + E_{n+1,j+1} \ge D_{n,m}.$$

(7)

PROOF. The first assertion is proved by reverse induction on i, proceeding for values $i = n, \dots, 0$. Since $E_{n+1,m+1}$ is defined to be 0, the base case of the assertion holds. Assume by the induction hypothesis that $D_{i+1,m} + E_{i+2,m+1} \ge D_{n,m}$ By (4), $E_{i+1,m+1} = E_{i+2,m+1} + \frac{\tau}{2}\rho(a_{i+1}, b_{m+1})$. Thus $D_{i,m} + E_{i+1,m+1}$ equals

$$= D_{i,m} + \left(E_{i+2,m+1} + \frac{\tau}{2} \cdot \rho(a_{i+1}, b_{m+1}) \right)$$

$$= E_{i+2,m+1} + \left(D_{i,m} + \frac{\tau}{2} \cdot \rho(a_{i+1}, b_{m+1}) \right)$$

$$= E_{i+2,m+1} + \left(D_{i,m} + \frac{\tau}{2} \cdot \frac{\rho(a_{i+1}, b_m) + \rho(a_{i+1}, b_{m+1})}{2} \right)$$

$$\ge E_{i+2,m+1} + D_{i+1,m} = D_{i+1,m} + E_{i+2,m+1}$$

$$\ge D_{n,m}$$

where the the third equality holds since $b_m = b_{m+1}$, the first inequality arises from (3) and the second inequality is justified by the induction hypothesis.

The second assertion is proved by reverse induction on j, proceeding for values $j = m, \dots, 0$. Since $E_{n+1,m+1}$ is defined to be 0, the base case of the assertion holds. Assume by the induction hypothesis that $D_{n,j+1} + E_{n+1,j+2} \ge D_{n,m}$ By (5), $E_{n+1,j+1} = E_{n+1,j+2} + \frac{\mu}{2} \cdot \rho(a_{n+1}, b_{j+1})$. Thus $D_{n,j} + E_{n+1,j+1}$ is equal to

$$= D_{n,j} + \left(E_{n+1,j+2} + \frac{\mu}{2} \cdot \rho(a_{n+1}, b_{j+1}) \right)$$

$$= E_{n+1,j+2} + \left(D_{n,j} + \frac{\mu}{2} \cdot \rho(a_{n+1}, b_{j+1}) \right)$$

$$= E_{n+1,j+2} + \left(D_{n,j} + \frac{\mu}{2} \cdot \frac{\rho(a_n, b_{j+1}) + \rho(a_{n+1}, b_{j+1})}{2} \right)$$

$$\geq E_{n+1,j+2} + D_{n,j+1} = D_{n,j+1} + E_{n+1,j+2}$$

$$\geq D_{n,m}$$

where the third equality holds because $a_n = a_{n+1}$, the first inequality arises from (3) and the second inequality is justified by the induction hypothesis. This establishes the Claim A0. Q.E.D.

CLAIM A: For $0 \leq i \leq n$ and $0 \leq j \leq m$,

$$D_{i,j} + E_{i+1,j+1} \geq D_{n,m}. \tag{8}$$

PROOF. By reverse induction on $i + j$, where $0 \leq i \leq n$ and $0 \leq j \leq m$, proceeding for values $i + j = n + m, \ldots, 0$. Inequality (7) from Claim A0 establishes the base case when either $i = n$ or $j = m$. Assume the induction hypothesis

$$D_{i,j-1} + E_{i+1,j} \geq D_{n,m}$$
$$D_{i-1,j} + E_{i,j+1} \geq D_{n,m}. \tag{9}$$

Now by (6), $E_{i+1,j+1}$ is the minimum value over three cases.

CASE 1: $E_{i,j} = E_{i+1,j+1} + \frac{\tau+\mu}{2} \cdot \rho(a_i, b_j)$. In this case,

$$D_{i-1,j-1} + E_{i,j} = D_{i-1,j-1} + \left(E_{i+1,j+1} + \frac{\tau + \mu}{2} \cdot \rho(a_i, b_j) \right)$$

$$= E_{i+1,j+1} + \left(D_{i-1,j-1} + \frac{\tau + \mu}{2} \cdot \rho(a_i, b_j) \right)$$

$$\geq E_{i+1,j+1} + D_{i,j} = D_{i,j} + E_{i+1,j+1}$$

$$\geq D_{n,m}$$

where the first inequality arises from (3) and the second inequality is justified by the induction hypothesis.

CASE 2: $E_{i,j} = E_{i+1,j} + \frac{\tau}{2} \cdot \frac{\rho(a_i,b_j)+\rho(a_i,b_{j-1})}{2}$. In this case,

$$
\begin{aligned}
D_{i-1,j-1} + E_{i,j} &= D_{i-1,j-1} + \left(E_{i+1,j} + \frac{\tau}{2} \cdot \frac{\rho(a_i,b_j) + \rho(a_i,b_{j-1})}{2} \right) \\
&= E_{i+1,j} + \left(D_{i-1,j-1} + \frac{\tau}{2} \cdot \frac{\rho(a_i,b_j) + \rho(a_i,b_{j-1})}{2} \right) \\
&\geq E_{i+1,j} + D_{i,j-1} = D_{i,j-1} + E_{i+1,j} \\
&\geq D_{n,m}
\end{aligned}
$$

where the first inequality arises from (3) and the second inequality is justified by the induction hypothesis.

CASE 3: $E_{i,j} = E_{i,j+1} + \frac{\mu}{2} \cdot \frac{\rho(a_i,b_j)+\rho(a_{i-1},b_j)}{2}$. In this case,

$$
\begin{aligned}
D_{i-1,j-1} + E_{i,j} &= D_{i-1,j-1} + \left(E_{i,j+1} + \frac{\mu}{2} \cdot \frac{\rho(a_i,b_j) + \rho(a_{i-1},b_j)}{2} \right) \\
&= E_{i,j+1} + \left(D_{i-1,j-1} + \frac{\mu}{2} \cdot \frac{\rho(a_i,b_j) + \rho(a_{i-1},b_j)}{2} \right) \\
&\geq E_{i,j+1} + D_{i-1,j} = D_{i-1,j} + E_{i,j+1} \\
&\geq D_{n,m}
\end{aligned}
$$

where the first inequality arises from (3) and the second inequality is justified by the induction hypothesis. This establishes the Claim A. Q.E.D.

CLAIM B0: For $0 \leq i \leq n$ and $0 \leq j \leq m$,

$$
\begin{aligned}
D_{i,0} + E_{i+1,1} &\geq E_{1,1} \\
D_{0,j} + E_{1,j+1} &\geq E_{1,1}.
\end{aligned}
\tag{10}
$$

PROOF. The first assertion is proved by induction on i, proceeding for values $i = 0, \ldots, n$. Since $D_{0,0}$ is defined to be 0, the base case of the assertion holds. Assume by the induction hypothesis that $D_{i,0} + E_{i+1,1} \geq E_{1,1}$. By (1), $D_{i+1,0} = D_{i,0} + \frac{\tau}{2} \cdot \rho(a_{i+1}, b_0)$. Thus $D_{i+1,0} + E_{i+2,1}$ is equal to

$$
\begin{aligned}
&= \left(D_{i,0} + \frac{\tau}{2} \cdot \rho(a_{i+1}, b_0) \right) + E_{i+2,1} \\
&= D_{i,0} + \left(E_{i+2,1} + \frac{\tau}{2} \frac{\rho(a_{i+1}, b_0) + \rho(a_{i+1}, b_1)}{2} \right) \\
&\geq D_{i,0} + E_{i+1,1} \\
&\geq E_{1,1}
\end{aligned}
$$

where the second equality holds because $b_0 = b_1$, the first inequality arises from (6) and the second inequality is justified by the induction hypothesis.

The second assertion is proved by induction on j, proceeding for values $j = 0, \ldots, m$. Since $D_{0,0}$ is defined to be 0, the base case of the assertion holds. Assume by the induction hypothesis that $D_{0,j} + E_{1,j+1} \geq E_{1,1}$. By (2), $D_{0,j+1} = D_{0,j} + \frac{\mu}{2} \cdot \rho(a_0, b_{j+1})$. Thus $D_{0,j+1} + E_{1,j+2}$ is equal to

$$
\begin{aligned}
&= \left(D_{0,j} + \frac{\mu}{2} \cdot \rho(a_0, b_{j+1}) \right) + E_{1,j+2} \\
&= D_{0,j} + \left(E_{1,j+2} + \frac{\mu}{2} \cdot \rho(a_0, b_{j+1}) \right) \\
&= D_{0,j} + \left(E_{1,j+2} + \frac{\mu}{2} \cdot \frac{\rho(a_0, b_{j+1}) + \rho(a_1, b_{j+1})}{2} \right) \\
&\geq D_{0,j} + E_{1,j+1} \\
&\geq E_{1,1}
\end{aligned}
$$

where the third equality holds because $a_0 = a_1$, the first inequality arises from (6) and the second inequality is justified by the induction hypothesis. This establishes the Claim B0. Q.E.D.

CLAIM B: For $0 \le i \le n$ and $0 \le j \le m$,

$$D_{i,j} + E_{i+1,j+1} \ge E_{1,1}. \tag{11}$$

PROOF. By induction on $i + j$, where $i + j$ takes values from 0 to $n + m$. Inequality (10) from Claim B0 establishes the base case when either $i = 0$ or $j = 0$. Assume the induction hypothesis

$$D_{i,j-1} + E_{i+1,j} \ge E_{1,1}$$
$$D_{i-1,j} + E_{i,j+1} \ge E_{1,1}. \tag{12}$$

Now by (3) $D_{i,j}$ is the minimum value over three cases.

CASE 1: $D_{i,j} = D_{i-1,j-1} + \frac{\tau+\mu}{2}\rho(a_i, b_j)$ In this case,

$$
\begin{aligned}
D_{i,j} + E_{i+1,j+1} &= \left(D_{i-1,j-1} + \frac{\tau + \mu}{2} \cdot \rho(a_i, b_j) \right) + E_{i+1,j+1} \\
&= D_{i-1,j-1} + \left(E_{i+1,j+1} + \frac{\tau + \mu}{2} \cdot \rho(a_i, b_j) \right) \\
&\ge D_{i-1,j-1} + E_{i,j} \\
&\ge E_{1,1}
\end{aligned}
$$

where the first inequality arises from (6) and the second inequality arises from the induction hypothesis.

CASE 2: $D_{i,j} = D_{i-1,j} + \frac{\tau}{2} \cdot \frac{\rho(a_i,b_j)+\rho(a_i,b_{j+1})}{2}$. In this case,

$$
\begin{aligned}
D_{i,j} + E_{i+1,j+1} &= \left(D_{i-1,j} + \frac{\tau}{2} \cdot \frac{\rho(a_i, b_j) + \rho(a_i, b_{j+1})}{2} \right) + E_{i+1,j+1} \\
&= D_{i-1,j} + \left(E_{i+1,j+1} + \frac{\tau}{2} \cdot \frac{\rho(a_i, b_j) + \rho(a_i, b_{j+1})}{2} \right) \\
&\ge D_{i-1,j} + E_{i,j+1} \\
&\ge E_{1,1}
\end{aligned}
$$

where the first inequality arises from (6) and the second inequality arises from the induction hypothesis.

CASE 3: $D_{i,j} = D_{i,j-1} + \frac{\mu}{2} \cdot \frac{\rho(a_i,b_j)+\rho(a_{i+1},b_j)}{2}$.

In this case,

$$
\begin{aligned}
D_{i,j} + E_{i+1,j+1} &= \left(D_{i,j-1} + \frac{\mu}{2} \cdot \frac{\rho(a_i,b_j)+\rho(a_{i+1},b_j)}{2} \right) + E_{i+1,j+1} \\
&= D_{i,j-1} + \left(E_{i+1,j+1} + \frac{\mu}{2} \cdot \frac{\rho(a_i,b_j)+\rho(a_{i+1},b_j)}{2} \right) \\
&\geq D_{i,j-1} + E_{i+1,j} \\
&\geq E_{1,1}
\end{aligned}
$$

where the first inequality arises from (6) and the second inequality arises from the induction hypothesis. This establishes Claim B.

Theorem 3. *Given sequences $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$, we have $D_{n,m} = E_{1,1}$*

In other words, time warping Algorithm 1 is *symmetric*; i.e. time warping distance between sequences $a = a_1, \ldots, a_n$ and $b = b_1, \ldots, b_m$ is the same when computed from left to right as when computed from right to left, or in other words, the output of Algorithm 1 is the same as that of Algorithm 2.

We can denote this as $D(a,b) = E(a,b)$ or $D(a_1, \ldots, a_n, b_1, \ldots, b_m) = E(a_1, \ldots, a_n, b_1, \ldots, b_m)$.

Note that we do not claim that $D_{i,j} + E_{i+1,j+1} = D_{n,m}$. Indeed $D_{i,j} + E_{i+1,j+1}$ is the optimal time warping distance when aligning $a_1, \ldots, a_i$ with $b_1, \ldots, b_j$ along with aligning $a_{i+1}, \ldots, a_n$ with $b_{j+1}, \ldots, b_m$.

4. Examples

Using Algorithm 1, we computed the time warping distance between each *normalized*[8] yeast and each *normalized* human gene expression profile, using Affymetrix time series data for *S. cerevisiae* with time intervals $0, 10, 20, \ldots, 160$ minutes [5] and Affymetrix time series data for *H. sapiens* with time intervals $0, 2, 4, \ldots, 24$ hours [6].

Consider yeast protein YMR001C, which is a protein kinase of the Ser/Thr family ascribed by Cho et al. [5] as related to DNA replication. We computed that human U05340 has smaller time warping distance with YMR001C (with the value 26.18), than any other human gene among the set of 7077 human genes from data of Cho et al. [6]. Another name for YMR001C is CDC5 and another name for U05340 is CDC20, and both are CDC proteins (cell division control) proteins. U05340 is named CDC20 because it is homologous to the yeast protein CDC20.

In Figure 3(i), the time series for yeast gene YMR001C and human gene U05340 are superposed, where it should be noted that YMR001C has 17 time points, at times $0, 10, 20, \ldots, 160$ minutes, and that U05340 has 13 time points, at times $0, 2, 4, \ldots, 24$ hours (both series represent approximately two cell cycles for each species). Note the obvious similarities in up- and down-expression, though the two genes cannot be directly compared using, for

---

[8] Normalization means that time series expression data for each gene $x_1, \ldots, x_n$ is replaced by Z-scores $z_1, \ldots, z_n$, where $z_i = \frac{x_i - \mu}{\sigma}$, $\mu = \sum_{i=1}^{n} x_i/n$, $\sigma = \sqrt{\sum_{i=1}^{n} (x_i - \mu)^2/n - 1}$.
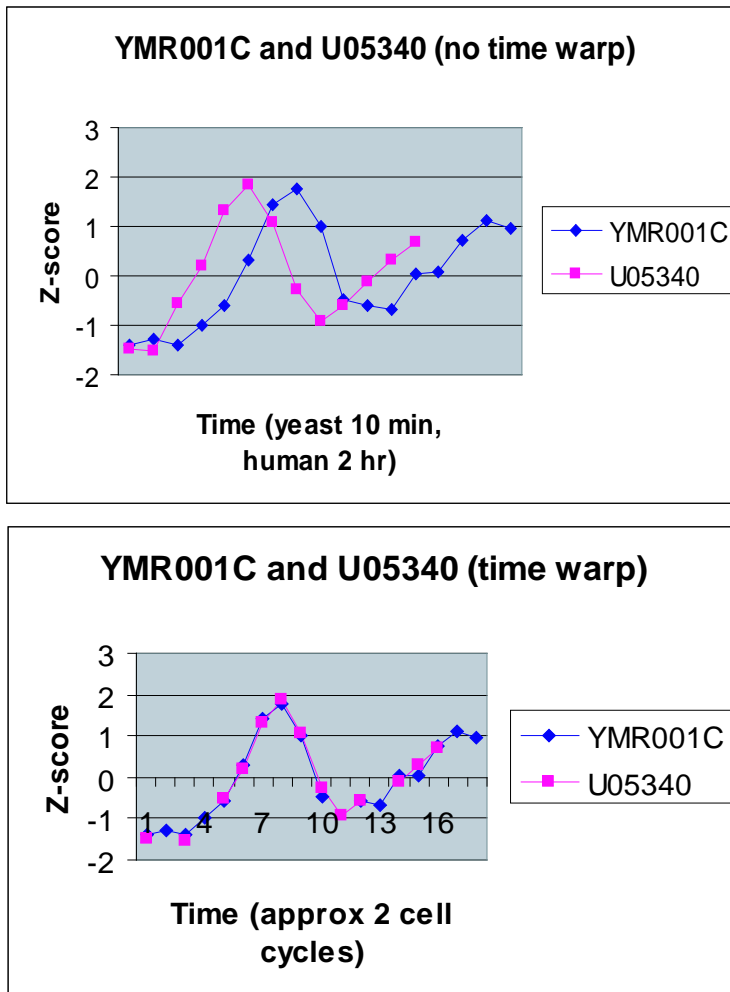
Fig. 3. *(i)* Superposition of time series for YMR001C and U05340 (without time warping). *(ii)* Time warping of YMR001C and U05340.

instance, Pearson correlation coefficient, because of the different number of time points. In Figure 3(ii), an optimal (symmetric) time warping of U05340 with YMR001C is graphed, with time warp distance 26.18.

In Figure 4(ii), the normalized time series data, i.e. Z-scores, are depicted from those ten genes from human expression data of Cho et al. [6], whose time warp distance to YMR001C is the least among human genes. In this case, the time warp distances between YMR001C and those 10 human genes having smallest time warp distance range from 26.18 to 50.99.

Scrutiny of the biology of YMR001C, U05340, and also yeast CDC20 leads to the conclusion that U05340 and YMR001, alias CDC5, are not homologous proteins, as measured by BLAST sequence homology; additionally, since YRM001C is a kinase and U05340 is not, there is no likely homology.[9]

The previous example suggests a cautionary tale, where small time warp distances may indicate a role in the same or a related metabolic pathway, rather than protein homology. This might be of promise when considering time series expression data from two closely related species, one of which is not yet annotated (e.g. among funghi which have potential medical interest because of human yeast infections).

Human gene U05340 has the smallest symmetric time warp distance to yeast gene YMR001C, among all human genes from [6], though YMR001C is not homologous to U05340. However, yeast protein YGL116W is homologous to U05340, and U05340 has the 10-th smallest symmetric time warp distance to YGL116W, that being 51.17 – see Figure 5.

---

[9] Accessions for the yeast gene YMR001C are Genbank M84220, SwissProt P32562, GI 172186, and accessions for human gene U05340 are GI 468032 (protein), GI 468031 (nucleotide), GB NM_001255, SwissProt Q12834.

5. Boltzmann probability

The time warping *partition function* Z between sequences $a = a_1, \ldots, a_n$ and $b = b_1, \ldots, b_m$ is defined to be

$$Z = \sum e^{-S(\mathcal{A})/kT} \tag{13}$$

where the sum is over all possible time warpings $\mathcal{A}$ of sequences a and b, $S(\mathcal{A})$ is the score of time warping $\mathcal{A}$, k is the Boltzmann constant, and T is temperature.[10] Though the sum is over exponentially many time warpings, Algorithm 4 allows the computation of Z in time $O(n \cdot m)$. This is done by computing the *forward partition function* $FZ_{i,j}$, defined as the sum of $e^{-S(\mathcal{A})/kT}$ over all time warpings $\mathcal{A}$ of $a_1, \ldots, a_i$ and $b_1, \ldots, b_j$. Let $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$ be given sequences of time series values, where a_i [resp. b_j] is the value at time $\tau \cdot (i - 1)$ [resp. $\mu \cdot (j - 1)$].

Algorithm 4 (Forward partition function)

$FZ_{0,0} = 1; \ a_0 = a_1; \ b_0 = b_1; \ a_{n+1} = a_n; \ b_{m+1} = b_m$

for i = 1 to n

$\quad FZ_{i,0} = FZ_{i-1,0} \cdot e^{-(\frac{\tau}{2} \cdot \rho(a_i, b_0))/kT}$

---

[10] The *Boltzmann probability* of time warping $\mathcal{A}$ is defined as $Pr[\mathcal{A}] = \frac{e^{-S(\mathcal{A})/kT}}{Z}$. In the present setting, we can take kT to be any constant. If T is infinite, then Z equals the total number of time warpings and the Boltzmann probability of any time warping $\mathcal{A}$ equals $\frac{1}{Z}$. If T is positive, but close to 0, then the Boltzmann probability of any time warping $\mathcal{A}$ equals 0, if $\mathcal{A}$ is not optimal, and $\frac{1}{M}$, if there exist M optimal time warpings. See Clote-Backofen [2] for proofs of these assertions.

```
for j = 1 to m
```

$$FZ_{0,j} = FZ_{0,j-1} \cdot e^{-(\frac{\mu}{2} \cdot \rho(a_0, b_j))/kT}$$

```
for i = 1 to n
```

```
  for j = 1 to m
```

$$case_1 = FZ_{i-1,j-1} \cdot e^{-(\frac{\tau+\mu}{2} \cdot \rho(a_i, b_j))/kT}$$

$$case_2 = FZ_{i-1,j} \cdot e^{-(\frac{\tau}{2} \cdot \frac{\rho(a_i, b_j) + \rho(a_i, b_{j+1})}{2})/kT}$$

$$case_3 = FZ_{i,j-1} \cdot e^{-(\frac{\mu}{2} \cdot \frac{\rho(a_i, b_j) + \rho(a_{i+1}, b_j)}{2})/kT}$$

$$FZ_{i,j} = case_1 + case_2 + case_3$$

In a similar fashion, we compute the *backwards partition function* $BZ_{i,j}$, defined as the sum of $e^{-\mathcal{A}/kT}$ over all *reverse* time warpings $\mathcal{A}$ between $a_i, \ldots, a_n$ and $b_j, \ldots, b_m$.

Algorithm 5 (Backward time warping)

$$BZ_{n+1,m+1} = 1; \ a_0 = a_1; \ b_0 = b_1; \ a_{n+1} = a_n; \ b_{m+1} = b_m$$

```
for i = n down to 1
```

$$BZ_{i,m+1} = BZ_{i+1,m+1} \cdot e^{-(\frac{\tau}{2} \cdot \rho(a_i, b_{m+1}))/kT}$$

```
for j = m down to 1
```

$$BZ_{n+1,j} = BZ_{n+1,j+1} \cdot e^{-(\frac{\mu}{2} \cdot \rho(a_{n+1}, b_j))/kT}$$

```
for i = n down to 1
```

```
  for j = m down to 1
```

$$case_1 = BZ_{i+1,j+1} \cdot e^{-(\frac{\tau+\mu}{2} \cdot \rho(a_i, b_j))/kT}$$

$$case_2 = BZ_{i+1,j} \cdot e^{-(\frac{\tau}{2} \cdot \frac{\rho(a_i, b_j) + \rho(a_i, b_{j+1})}{2})/kT}$$

$$case_3 = BZ_{i,j+1} \cdot e^{-(\frac{\mu}{2} \cdot \frac{\rho(a_i, b_j) + \rho(a_{i+1}, b_j)}{2})/kT}$$

$$BZ_{i,j} = case_1 + case_2 + case_3$$

Straightforward modification of the proof of symmetry in Section 3 yields

an inductive proof of

$$FZ_{i,j} \cdot BZ_{i+1,j+1} \geq FZ_{n,m}$$

$$FZ_{i,j} \cdot BZ_{i+1,j+1} \geq BZ_{1,1}$$

for all $0 \leq i \leq m$, $0 \leq j \leq m$, and hence $FZ_{n,m} = Z = BZ_{1,1}$. We define

the *Boltzmann probability* $Pr[\mathcal{A}]$ of time warping $\mathcal{A}$ by

$$Pr[\mathcal{A}] = \frac{e^{-S(\mathcal{A})/kT}}{Z}$$

It follows from the proof that $FZ_{i,j} \cdot BZ_{i+2,j+2}$ is the sum of $e^{-\mathcal{A}/kT}$

over all time warpings $\mathcal{A}$ of $a_1, \ldots, a_i$ with $b_1, \ldots, b_j$ and of $a_{i+2}, \ldots, a_n$ with

$b_{j+2}, \ldots, b_m$. Since there are only three manners to extend time warpings of

$a_1, \ldots, a_i$ with $b_1, \ldots, b_j$ to time warpings of $a_1, \ldots, a_{i+1}$ with $b_1, \ldots, b_{j+1}$,

we can define the *Boltzmann pair probability* $Pr_{\mathcal{A}}[a_{i+1}, b_{j+1}]$ that a_{i+1} is

warped against b_{j+1}.

Suppose that $\mathcal{A}$ is a time warping between $a_1, \ldots, a_n$ with $b_1, \ldots, b_m$.

To define $Pr_{\mathcal{A}}[a_{i+1}, b_{j+1}]$, we consider four cases.

1. Case 1: a_i is aligned with b_j in $\mathcal{A}$, a_{i+1} is aligned with b_{j+1} in $\mathcal{A}$ and

   a_{i+2} is aligned with b_{j+2} in $\mathcal{A}$.

2. Case 2: a_i is aligned with b_j in $\mathcal{A}$, a_{i+1} is aligned with b_{j+1} in $\mathcal{A}$ and

   a_{i+1} is aligned with b_{j+2} in $\mathcal{A}$.

3. Case 3: a_i is aligned with b_j in $\mathcal{A}$, a_{i+1} is aligned with b_{j+1} in $\mathcal{A}$ and

   a_{i+2} is aligned with b_{j+1} in $\mathcal{A}$.

4. Case 4: a_{i+1} is not aligned with b_{j+1} in $\mathcal{A}$.

Then we define $Pr_{\mathcal{A}}[a_{i+1}, b_{j+1}]$ by

$$Pr_{\mathcal{A}}[a_{i+1}, b_{j+1}] = \begin{cases} \dfrac{FZ_{i,j} \cdot e^{-(\frac{\tau+\mu}{2} \cdot \rho(a_{i+1},b_{j+1}))/kT)} \cdot BZ_{i+2,j+2}}{Z} & \text{if Case 1} \\[3ex] \dfrac{FZ_{i,j} \cdot e^{-(\frac{\mu}{2} \cdot \frac{\rho(a_i,b_{j+1})+\rho(a_{i+1},b_{j+1})}{2})/kT} \cdot BZ_{i+1,j+2}}{Z} & \text{if Case 2} \\[3ex] \dfrac{FZ_{i,j} \cdot e^{-(\frac{\tau}{2} \cdot \frac{\rho(a_{i+1},b_j)+\rho(a_{i+1},b_{j+1})}{2})/kT} \cdot BZ_{i+2,j+1}}{Z} & \text{if Case 3} \\[3ex] 0 & \text{if Case 4} \end{cases}$$

Table 2 displays an optimal time warping $\mathcal{A}$ of gene expression data from YGL116W (17 time points) and U05340 (13 time points) along with Boltzmann pair probabilities of time warping a_i with b_j with respect to $\mathcal{A}$. This time warping is graphically depicted in Figure 5. Dashes in the first two columns of Table 2 mean that the gene expression value above the dash in the same column is warped against the corresponding gene expression value in the adjacent column. Though the optimal time warping does not depend on values of parameters k, T (Boltzmann constant k, temperature T), the Boltzmann probabilities necessarily do depend on these parameters. The sum $\sum_{\mathcal{X}} Pr[\mathcal{X}]$ over all exponentially many possible time warpings $\mathcal{X}$ equals 1, regardless of choice of parameters k, T; however, the pair probabilities $Pr_{\mathcal{A}}[a_i, b_j]$ depend on the time warping $\mathcal{A}$ as well as parameters k, T.

Figure 6 graphs the Boltzmann probabilities given in Table 2 along with those obtained by our algorithm when $T = 0.5$ and $T = 1$.

Some discussion of choice of temperature is necessary. In the context of RNA secondary structure, Zuker's algorithm [17,16] is a dynamic program-

YGL116W	U05340	$k = 1, T = 0.1$	$k = 1, T = 2$
-0.994588897451	-1.366	0.518910	0.193674
-0.976486101673	-1.4175	0.776216	0.104172
-0.867869327005	-	0.997866	0.382842
-0.885972122783	-	0.999523	0.565344
-0.777355348115	-	0.999989	0.486676
-0.252374270552	-0.4265	0.999986	0.273227
-	0.3265	0.999987	0.585945
1.30446616636	1.444	1.000000	0.872255
2.40873670882	1.999	0.999993	0.509778
1.64841928614	1.198	1.000000	0.678834
-0.0713463127722	-0.1625	1.000000	0.668327
-0.324785453665	-	1.000000	0.723832
-0.723046960781	-0.79	1.000000	0.373339
-0.632532981891	-	1.000000	0.551806
-0.578224594557	-0.472	0.999999	0.374425
0.254504011232	-0.0005	0.999882	0.402818
0.489840356347	0.441	0.999993	0.441802
0.978615842353	0.8285	0.998077	0.382370

Table 2. Optimal time warping $\mathcal{A}$ of gene expression data from YGL116W (17 time points) and U05340 (13 time points) along with Boltzmann pair probabilities of time warping a_i with b_j with respect to $\mathcal{A}$. Dashes in the first two columns mean that the gene expression value above the dash in the same column is warped against the corresponding gene expression value in the adjacent column. Column 3 [resp. 4] contains Boltzmann probabilities when $k = 1$, $T = 0.1$ [resp. $k = 1$, $T = 2$]. Though the optimal time warping does not depend on values of k, T, the Boltzmann probabilities necessarily do depend on these parameters. The sum $\sum_{\mathcal{X}} Pr[\mathcal{X}]$ over all exponentially many possible time warpings $\mathcal{X}$ always equals 1, regardless of choice of parameters k, T; however, the pair probabilities $Pr_{\mathcal{A}}[a_i, b_j]$ depend on the time warping $\mathcal{A}$ as well as parameters k, T.

ming method which determines the optimal secondary structure for a given
RNA sequence with respect to the nearest neighbor model using experimen-
tally measured free energies at $37^\circ C$ [15]. The use of such experimentally
measured energies at specific temperatures ensures a natural significance for
temperature in the computation of Boltzmann probabilities for base pairs in
the low energy ensemble of RNA secondary structure, as computed by Mc-
Caskill's algorithm [8]. In contrast to the natural significance of temperature
in RNA secondary structure determination, in the context of time warping
temperature has no intrinsic significance – indeed warping factors $\tau/2$, $\mu/2$
and $(\tau + \mu)/2$ depend on time, rather than experimentally measured ener-
gies at certain temperatures. For that reason, we compute Boltzmann pair
probabilities for several temperatures, from which the user can see the more
significant portions of the alignment.

6. Conclusion

In this paper, we develop a new symmetric time warping algorithm, which
yields the same time warp distance when computed from left to right as
from right to left, and prove the symmetry of Algorithm 1 as well as that
of Aach's algorithm `genewarp.exe` with parameter `-a 2`. This symmetry
leads to a dynamic programming algorithm to compute the forward and
backwards partition function, and hence the Boltzmann probability $Pr[\mathcal{A}]$ of
time warping $\mathcal{A}$, as well as the Boltzmann probability $Pr_{\mathcal{A}}[a_i, b_j]$ of warping
a_i with b_j with respect to time warping $\mathcal{A}$. A possible application of the

Boltzmann probability is to indicate which portions of an optimal time warping $\mathcal{A}$ may be biologically more significant than others.

We designed and implemented the forward and reverse time warping and partition function Algorithms 1, 2, 4, 5 in Python on a Linux platform.

7. Acknowledgements

References

1. J. Aach and G. Church. Aligning gene expression time series with time warping algorithms. *Bioinformatics*, 17(6):495–508, 2001.

2. P. Clote and R. Backofen. *Computational Molecular Biology: An Introduction.* John Wiley & Sons, 2000. 286 pages.

3. M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA*, 95:14863–14868, 1998.

4. P. Spellman et al. Comprehensive identification of cell cycle-regulated genes of the yeast *saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell*, 9:3273–3297, 1998.

5. R. Cho et al. A genome-wide transcriptional analysis of the mitotic cell cycle. *Molecular Cell*, 2:65–73, 1998.

6. R. Cho et al. Transcriptional regulation and function during the human cell cycle. *Nature Genetics*, 27:48–54, 2001.

7. J.B. Kruskal and M. Liberman. The symmetric time-warping problem: From continuous to discrete. In J.B. Kruskal and D. Sankoff, editors, *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*, pages 125–161. CSLI Publications,Stanford, 1999. Text originally published by Addison-Wesley in 1983.

8. J.S. McCaskill. The equilibrium partition function and base pair binding probabilities for rna secondary structure. *Biopolymers*, 29:1105–1119, 1990.

9. U. Mückstein, I. Hofacker, and P. Stadler. Stochastic pairwise alignments. *Bioinformatics*, 18:S153–S160, 2002.

10. S. Myazawa. A reliable sequence alignment method based on probabilities of residue correspondences. *Protein Eng.*, 8:999–1009, 1994.

11. S.B. Needleman and C.D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Bio.*, 48:443–453, 1970.

12. T.F. Smith and M.S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.*, 147:195–197, 1981.

13. M. Vingron and P. Argos. Determination of reliable regions in protein sequence alignments. *Protein Eng.*, 3(7):565–569, 1990.

14. M. S. Waterman. *Introduction to Computational Biology - Maps, Sequences and Genomes.* Chapman & Hall, 1995.

15. T. Xia, Jr. J. SantaLucia, M.E. Burkard, R. Kierzek, S.J. Schroeder, X. Jiao, C. Cox, and D.H. Turner. Thermodynamic parameters for an expanded nearest-neighbor model for formation of RNA duplexes with Watson-Crick base pairs. *Biochemistry*, 37:14719–35, 1999.

16. M. Zuker. Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Res.*, 31(13):3406–3415, 2003.

17. M. Zuker and P. Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Res.*, 9:133–148, 1981.

Appendix

Symmetry of Aach-Church time warping

.

In this section, we make a change of notation. Time series $a_0, \ldots, a_{N-1}$ [resp. $b_0, \ldots, b_{M-1}$] consists of values a_i [resp. b_j] measured at time $t_1(i)$ [resp. $t_2(j)$] for $0 \leq i < N$ [resp. $0 \leq j < M$]. Our goal in this section to show that Aach's time warping [1] algorithm is *symmetric* in the sense that the time warping distance between $a_0, \ldots, a_{N-1}$ with $b_0, \ldots, b_{M-1}$ is the same whether computed from *left to right* or *right to left*. Let's look at the pseudocode[11] underlying the time warping algorithm [1] corresponding to the command

```
genewarp.exe -i1 input1.txt -i2 input2.txt -a 2 -o out.txt
```

Here `input1.txt` and `input2.txt` are text files, whose first line consists of `>N|comment|`, whose second line consists of tab separated time instances, beginning with a tab, and all other lines consist of a gene name with gene expression values, all tab separated, followed by a new line marker. Both `input1.txt` and `input2.txt` are required to have precisely the same gene names in the same order, although the time intervals and number of gene expression values may differ between both files.

---

[11] The pseudocode for this algorithm did not appear in either [1] or the corresponding web supplement. For that reason, we describe the pseudocode corresponding to `genewarp` with parameter `-a 2`. We'd like to thank John Aach for sharing his pseudocode.

Given sequences $a_0, \ldots, a_{N-1}$ and $b_0, \ldots, b_{M-1}$, let $t_1(i)$ denote the time value of time point i of first time series and $t_2(j)$ denote the time value of time point j of second time series. Without loss of generality, we will assume that $t_1(0) = 0 = t_2(0)$. Let $d(i,j)$ denote the Euclidean distance between feature vectors of time point i in first series and time point j in second series. In the following, we define the matrix

$$D = (D(i,j) : 0 \leq i < N, 0 \leq n < M)$$

where $D(i,j)$ is the optimal time warping distance between $a_0, \ldots, a_i$ and $b_0, \ldots, b_j$, as measured by Aach's algorithm. Let $D(0,0) = 0$. For $1 \leq j < M$ define

$$D(0,j) = D(0, j-1) + \left(\frac{t_2(j) - t_2(j-1)}{2} \cdot \frac{d(0,j) + d(0, j-1)}{2} \right) \quad (14)$$

and for $1 \leq i < N$ define

$$D(i,0) = D(i-1, 0) + \left(\frac{t_1(i) - t_1(i-1)}{2} \cdot \frac{d(i,0) + d(i-1, 0)}{2} \right). \quad (15)$$

For $1 \leq i < M$ and $1 \leq j < N$, define

$$D(i,j) = \min \begin{cases} D(i-1, j-1) + \left(\frac{(t_1(i) - t_1(i-1)) + (t_2(j) - t_2(j-1))}{2} \cdot \frac{d(i-1,j-1) + d(i,j)}{2} \right) \\ D(i-1, j) + \left(\frac{t_1(i) - t_1(i-1)}{2} \cdot \frac{d(i-1,j) + d(i,j)}{2} \right) \\ D(i, j-1) + \left(\frac{t_2(j) - t_2(j-1)}{2} \cdot \frac{d(i,j-1) + d(i,j)}{2} \right) \end{cases} \quad (16)$$

To investigate the symmetry of this version of time warping, we define the matrix

$$E = (E(i,j) : 0 \leq i < N, 0 \leq j < M)$$

by reverse induction. Let $E(N-1, M-1) = 0$. For $0 \leq j < M-1$ define $E(N-1, j)$ to be

$$E(N-1, j+1) + \left(\frac{t_2(j+1) - t_2(j)}{2} \cdot \frac{d(N-1, j) + d(N-1, j+1)}{2} \right) \tag{17}$$

and for $0 \leq i < N-1$ define $E(i, M-1)$ to be

$$E(i+1, M-1) + \left(\frac{t_1(i+1) - t_1(i)}{2} \cdot \frac{d(i, M-1) + d(i+1, M-1)}{2} \right) \tag{18}$$

For $0 \leq i < M-1$ and $0 \leq j < N-1$, define

$$E(i,j) = \min \begin{cases} E(i+1, j+1) + \left(\frac{(t_1(i+1)-t_1(i))+(t_2(j+1)-t_2(j))}{2} \cdot \frac{d(i,j)+d(i+1,j+1)}{2} \right) \\ E(i+1, j) + \left(\frac{t_1(i+1)-t_1(i)}{2} \cdot \frac{d(i,j)+d(i+1,j)}{2} \right) \\ E(i, j+1) + \left(\frac{t_2(j+1)-t_2(j)}{2} \cdot \frac{d(i,j)+d(i,j+1)}{2} \right). \end{cases} \tag{19}$$

Notice that in (17), (18), and (19), we have expressions involving terms $t_1(i+1) - t_1(i)$ [resp. $t_2(j+1) - t_2(j)$] instead of expressions involving terms $t_1(i) - t_1(i+1)$ [resp. $t_2(j) - t_2(j+1)$]. With this definition, the reader should see that $E(i,j)$ is the optimal time warping distance, according to Aach's algorithm, when proceeding from *left to right* between sequences $a_{N-1}, \ldots, a_i$ and $b_{M-1}, \ldots, b_j$ where time instances are $t_1'(N-1) = 0$, and for $i = N-2, \ldots, 0$ we have $t_1'(i) = t_1(i+1) - t_1(i)$, and similarly $t_2'(M-1) = 0$, and for $j = M-2, \ldots, 0$ we have $t_2'(j) = t_2(j+1) - t_2(j)$. Equivalently, we consider $E(i,j)$ to be the optimal time warping distance, according to Aach's algorithm, between $a_i, \ldots, a_{N-1}$ and $b_j, \ldots, b_{M-1}$ when proceeding from *right to left*. We now would like to show that $D(N-1, M-1) = E(0,0)$.

CLAIM C0: For $0 \leq i \leq N - 1$,

$$D(i, M - 1) + E(i, M - 1) \geq D(N - 1, M - 1)$$

$$(20)$$

$$D(i, M - 1) + E(i, M - 1) \geq D(N - 1, M - 1).$$

PROOF. The first assertion is proved by reverse induction on i. When $i = N - 1$, by definition $E(N - 1, M - 1) = 0$ so the base case holds. Assume by the induction hypothesis that $D(i + 1, M - 1) + E(i + 1, M - 1) \geq D(N - 1, M - 1)$. Then $D(i, M - 1) + E(i, M - 1)$ equals

$$= D(i, M - 1) + \left(E(i + 1, M - 1) + (\frac{t_2(i + 1) - t_2(i)}{2} \cdot \frac{d(i, M - 1) + d(i + 1, M - 1)}{2}) \right)$$

$$= E(i + 1, M - 1) + \left(D(i, M - 1) + (\frac{t_2(i + 1) - t_2(i)}{2} \cdot \frac{d(i, M - 1) + d(i + 1, M - 1)}{2}) \right)$$

$$\geq E(i + 1, M - 1) + D(i + 1, M - 1) = D(i + 1, M - 1) + E(i + 1, M - 1)$$

$$\geq D(N - 1, M - 1)$$

where the first equality holds because of (18), the first inequality holds because of (16) and the last inequality arises from the induction hypothesis.

The second assertion is proved by reverse induction on j. When $j = M - 1$, by definition $E(N - 1, M - 1) = 0$ so the base case holds. Assume by the induction hypothesis that $D(N - 1, j + 1) + E(N - 1, j + 1) \geq D(N - 1, M - 1)$. Then $D(N - 1, j) + E(N - 1, j)$ equals

$$= D(N - 1, j) + \left(E(N - 1, j + 1) + (\frac{t_2(j + 1) - t_2(j)}{2} \cdot \frac{d(N - 1, j) + d(N - 1, j + 1)}{2}) \right)$$

$$= E(N - 1, j + 1) + \left(D(N - 1, j) + (\frac{t_2(j + 1) - t_2(j)}{2} \cdot \frac{d(N - 1, j) + d(N - 1, j + 1)}{2}) \right)$$

$$\geq E(N - 1, j + 1) + D(N - 1, j + 1) = D(N - 1, j + 1) + E(N - 1, j + 1)$$

$$\geq D(N - 1, M - 1)$$

where the first equality holds because of (18), the first inequality holds because of (16) and the last inequality is justified by the induction hypothesis. This establishes Claim C0. Q.E.D.

CLAIM C: For $0 \leq i \leq N - 1$, $0 \leq j \leq M - 1$,

$$D(i, j) + E(i, j) \geq D(N - 1, M - 1). \qquad (21)$$

PROOF. By reverse induction on $i + j$. When either $i = N - 1$ or $j = M - 1$, the assertion holds by (20) of Claim C0, so we may assume that $i < N$ and $j < M - 1$. Assume the induction hypothesis

$$D(i, j - 1) + E(i, j - 1) \geq D(N - 1, M - 1)$$
$$D(i - 1, j) + E(i - 1, j) \geq D(N - 1, M - 1). \qquad (22)$$

Now by (19) $E(i, j)$ is the minimum value over three cases.

CASE 1: $E(i, j) = E(i+1, j+1) + \left(\frac{(t_1(i+1)-t_1(i))+(t_2(j+1)-t_2(j))}{2} \cdot \frac{d(i,j)+d(i+1,j+1)}{2} \right)$. In this case, temporarily let Δ denote $\frac{(t_1(i+1)-t_1(i))+(t_2(j+1)-t_2(j))}{2}$. Then $D(i, j) + E(i, j)$ equals

$$= D(i, j) + \left(E(i + 1, j + 1) + (\Delta \cdot \frac{d(i, j) + d(i + 1, j + 1)}{2}) \right)$$
$$= E(i + 1, j + 1) + \left(D(i, j) + (\Delta \cdot \frac{d(i, j) + d(i + 1, j + 1)}{2}) \right)$$
$$\geq E(i + 1, j + 1) + D(i + 1, j + 1) = D(i + 1, j + 1) + E(i + 1, j + 1)$$
$$\geq D(N - 1, M - 1)$$

where the first inequality arises from (16) and the last inequality is justified by the induction hypothesis.

CASE 2: $E_{i,j} = E(i+1, j) + (\frac{t_1(i+1)-t_1(i)}{2} \cdot \frac{d(i,j)+d(i+1,j)}{2})$. In this case, $D(i, j) + E(i, j)$ equals

$$
\begin{aligned}
&= D(i,j) + \left(E(i+1,j) + (\frac{t_1(i+1)-t_1(i)}{2} \cdot \frac{d(i,j)+d(i+1,j)}{2}) \right) \\
&= E(i+1,j) + \left(D(i,j) + (\frac{t_1(i+1)-t_1(i)}{2} \cdot \frac{d(i,j)+d(i+1,j)}{2}) \right) \\
&\geq E(i+1,j) + D(i+1,j) = D(i+1,j) + E(i+1,j) \\
&\geq D(N-1, M-1)
\end{aligned}
$$

where the first inequality arises from (16) and the last inequality is justified by the induction hypothesis.

CASE 3: $E(i,j) = E(i, j+1) + (\frac{t_2(j+1)-t_2(j)}{2} \cdot \frac{d(i,j)+d(i,j+1)}{2})$. In this case, $D(i, j) + E(i, j)$ equals

$$
\begin{aligned}
&= D(i,j) + \left(E(i,j+1) + (\frac{t_2(j+1)-t_2(j)}{2} \cdot \frac{d(i,j)+d(i,j+1)}{2}) \right) \\
&= E(i,j+1) + \left(D(i,j) + (\frac{t_2(j+1)-t_2(j)}{2} \cdot \frac{d(i,j)+d(i,j+1)}{2}) \right) \\
&\geq E(i,j+1) + D(i,j+1) = D(i,j+1) + E(i,j+1) \\
&\geq D(N-1, M-1)
\end{aligned}
$$

where the first inequality arises from (16) and the last inequality is justified by the induction hypothesis. This establishes the Claim C. Q.E.D.

CLAIM D0: For $0 \leq i \leq N-1$ and $0 \leq j \leq M-1$,

$$
\begin{aligned}
D(0,j) + E(0,j) &\geq E(0,0) \\
D(i,0) + E(i,0) &\geq E(0,0).
\end{aligned}
\tag{23}
$$

PROOF. The first assertion is proved by induction on j. When $j = 0$, by definition $D(0,0) = 0$ so the base case holds. Assume by the induction

hypothesis that $D(0, j - 1) + E(0, j - 1) \geq E(0, 0)$. Then $D(0, j) + E(0, j)$

equals

$$
\begin{aligned}
&= \left(D(0, j - 1) + \left(\frac{t_2(j) - t_2(j - 1)}{2} \cdot \frac{d(0, j) + d(0, j - 1)}{2} \right) \right) + E(0, j) \\
&= D(0, j - 1) + \left(\left(\frac{t_2(j) - t_2(j - 1)}{2} \cdot \frac{d(0, j) + d(0, j - 1)}{2} \right) + E(0, j) \right) \\
&\geq D(0, j - 1) + E(0, j - 1) \\
&\geq E(0, 0)
\end{aligned}
$$

where the first equality holds because of (14), the first inequality holds

because of (19) and the last inequality arises from the induction hypothesis.

The second assertion is proved by induction on i. When $i = 0$, by defini-

tion $D(0, 0) = 0$ so the base case holds. Assume by the induction hypothesis

that $D(i - 1, 0) + E(i - 1, 0) \geq E(0, 0)$. Then $D(i, 0) + E(i, 0)$ equals

$$
\begin{aligned}
&= \left(D(i - 1, 0) + \left(\frac{t_1(i) - t_1(i - 1)}{2} \cdot \frac{d(i, 0) + d(i - 1, 0)}{2} \right) \right) + E(i, 0) \\
&= D(i - 1, 0) + \left(\left(\frac{t_1(i) - t_1(i - 1)}{2} \cdot \frac{d(i, 0) + d(i - 1, 0)}{2} \right) + E(i, 0) \right) \\
&\geq D(i - 1, 0) + E(i - 1, 0) \\
&\geq E(0, 0)
\end{aligned}
$$

where the first equality holds because of (15), the first inequality holds

because of (19) and the last inequality arises from the induction hypothesis.

This establishes Claim D0. Q.E.D.

CLAIM D: For $0 \leq i \leq N - 1$ and $0 \leq j \leq M - 1$,

$$
D(i, j) + E(i, j) \geq E(0, 0). \tag{24}
$$

PROOF. By induction on $i+j$, where $i+j$ takes values from 0 to $N+M-2$.

When either $i = 0$ or $j = 0$, the assertion holds by (23) of Claim D0. Thus

we may assume that $1 \le i, j$. Assume the induction hypothesis

$$D(i, j-1) + E(i, j-1) \ge E(0,0)$$
$$D(i-1, j) + E(i-1, j) \ge E(0,0).$$

(25)

Now by (16), for $1 \le i < N$, $1 \le j < M$, it is the case that $D(i, j)$ is the

minimum value over three cases.

CASE 1: $D(i, j) = D(i-1, j-1) + \left(\frac{(t_1(i)-t_1(i-1))+(t_2(j)-t_2(j-1))}{2} \cdot \frac{d(i-1,j-1)+d(i,j)}{2} \right)$.

In this case, temporarily let Δ denote $\frac{(t_1(i)-t_1(i-1))+(t_2(j)-t_2(j-1))}{2}$. Then

$D(i, j) + E(i, j)$ equals

$$= \left(D(i-1, j-1) + (\Delta \cdot \frac{d(i-1, j-1) + d(i, j)}{2}) \right) + E(i, j)$$
$$= D(i-1, j-1) + \left(E(i, j) + (\Delta \cdot \frac{d(i-1, j-1) + d(i, j)}{2}) \right)$$
$$\ge D(i-1, j-1) + E(i-1, j-1)$$
$$\ge E(0,0)$$

where the first inequality arises from (16) and the second inequality is justified by the induction hypothesis.

CASE 2: $D(i, j) = D(i-1, j) + \left(\frac{t_1(i)-t_1(i-1)}{2} \cdot \frac{d(i-1,j)+d(i,j)}{2} \right)$.

In this case, $D(i, j) + E(i, j)$ equals

$$= \left(D(i-1, j) + (\frac{t_1(i) - t_1(i-1)}{2} \cdot \frac{d(i-1, j) + d(i, j)}{2}) \right) + E(i, j)$$
$$= D(i-1, j) + \left((\frac{t_1(i) - t_1(i-1)}{2} \cdot \frac{d(i-1, j) + d(i, j)}{2}) + E(i, j) \right)$$

$$\geq D(i-1,j) + E(i-1,j)$$

$$\geq E(0,0)$$

where the first inequality arises from (16) and the second inequality is justified by the induction hypothesis.

CASE 3: $D(i,j) = D(i,j-1) + (\frac{t_2(j)-t_2(j-1)}{2} \cdot \frac{d(i,j-1)+d(i,j)}{2})$.

In this case, $D(i,j) + E(i,j)$ equals

$$= \left(D(i,j-1) + (\frac{t_2(j)-t_2(j-1)}{2} \cdot \frac{d(i,j-1)+d(i,j)}{2}) \right) + E(i,j)$$

$$= D(i,j-1) + \left((\frac{t_2(j)-t_2(j-1)}{2} \cdot \frac{d(i,j-1)+d(i,j)}{2}) + E(i,j) \right)$$

$$\geq D(i,j-1) + E(i,j-1)$$

$$\geq E(0,0)$$

where the first inequality arises from (19) and the second inequality is justified by the induction hypothesis. This establishes Claim D. Q.E.D.

**Theorem 6.** *Given sequences $a_0, \dots, a_{N-1}$ and $b_0, \dots, b_{M-1}$, we have $D(N-1, M-1) = E(0,0)$.*

PROOF. From Claims C0,C,D0,D it follows that for $0 \leq i \leq N-1$, $0 \leq j \leq M-1$, we have $D(i,j) + E(i,j) \geq D(N-1, M-1)$ and $D(i,j) + E(i,j) \geq E(0,0)$. It follows that $E(0,0) \geq D(N-1, M-1)$ and $D(N-1, M-1) \geq E(0,0)$, hence $D(N-1, M-1) = E(0,0)$. ∎

Theorem 6 asserts that Aach's algorithm [1] with parameter -a 2 is *symmetric*, in the sense that time warping distance (computed according to Aach's algorithm) between sequences $a = a_0, \dots, a_{N-1}$ and $b =$

$b_0, \ldots, b_{M-1}$ is the same when computed left to right as when computed from right to left. We can denote this as $D(a, b) = E(a, b)$, i.e.

$$D(a_0, \ldots, a_{N-1}, b_0, \ldots, b_{M-1}) = E(a_0, \ldots, a_{N-1}, b_0, \ldots, b_{M-1}).$$

Since Aach's algorthm `genewarp.exe` with parameter `-a 2` is symmetric, following the approach of Section 5, it is possible to define forward and backward partition functions and related Boltzmann probabilities with respect to Aach's approach. Since details are obvious and analogous to those given in Section 5, we leave this to the reader.
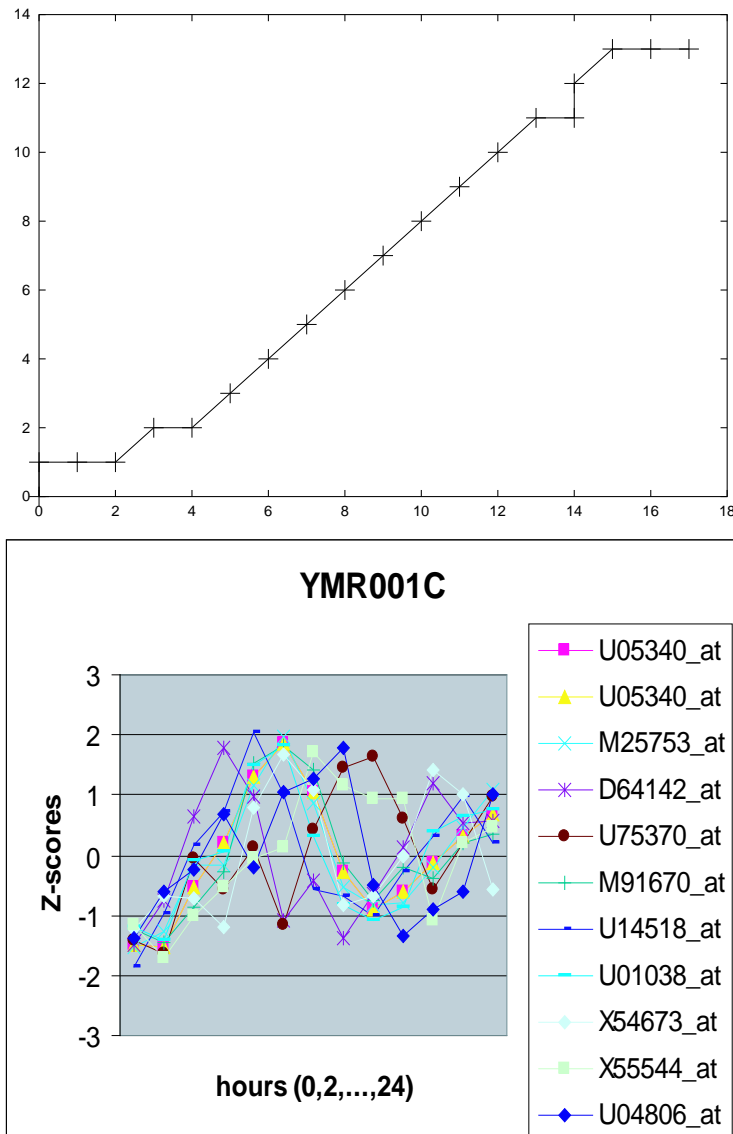
Fig. 4. *(i)* Path graph of YMR001C with U05340 produced by our algorithm, indicating regions of contraction/expansion (vertical/horizontal lines) and matches (diagonal lines). *(ii)* Ten human genes with smallest time warp distance to YMR001C, as computed from expression data of Cho et al. [6]. Note that there appears to be little correlation pairwise between these genes, although each has small time warp distance to YMR001C.
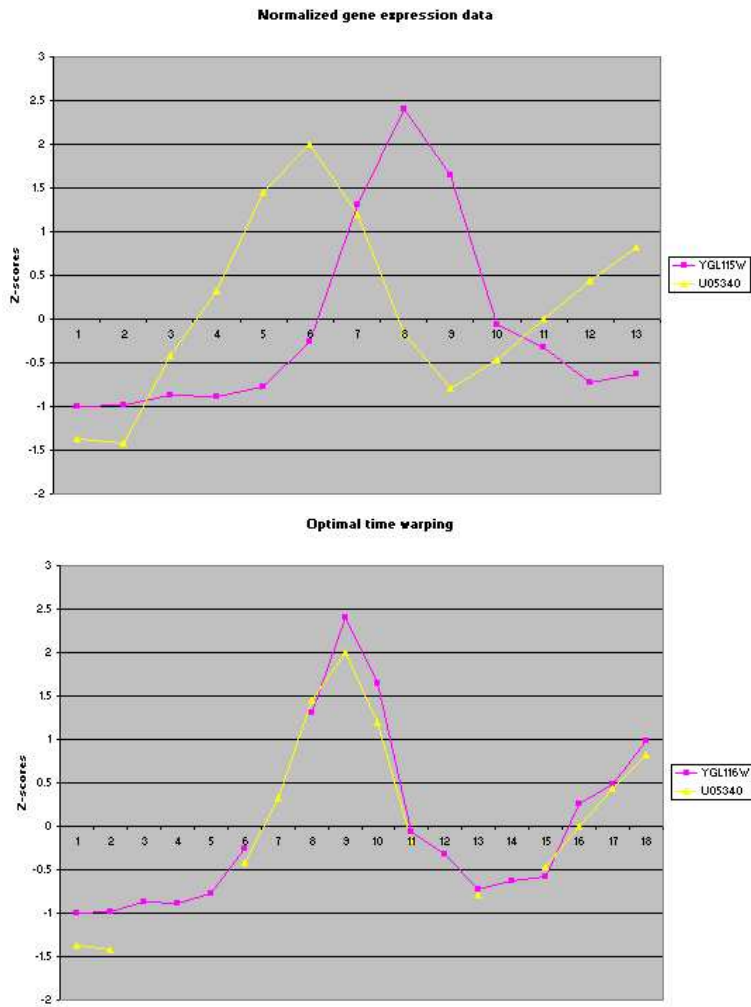
Fig. 5. *(i)* Superposition of time series for homologous genes YGL116W and U05340 (without time warping). *(ii)* Time warping of homologous genes YGL116W and U05340.
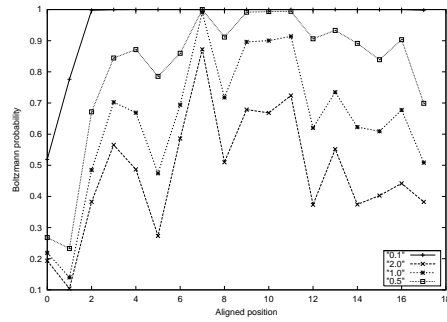
Fig. 6. Boltzmann pair probabilities for alignment positions in optimal time warping between YGL116W (17 time points) and U05340 (13 time points). Graphs are displayed when Boltzmann probabilities are computed with $k = 1$ and for the values 0.1, 0.5, 1.0 and 2.0 of T. With increasing values of T, the Boltzmann probability values decrease; this enhances the depiction of presumed biological significance of aligned positions in this optimal time warping.