Optimal Distance Estimation Between Compressed Data Series

Nikolaos M. Freris* Michail Vlachos* Suleyman S. Kozat†

Abstract

Most real-world data contain repeated or periodic patterns. This suggests that they can be effectively represented and compressed using only a few coefficients of an appropriate complete orthogonal basis (e.g., Fourier, Wavelets, Karhunen-Loève expansion or Principal Components).

In the face of ever increasing data repositories and given that most mining operations are distance-based, it is vital to perform accurate distance estimation directly on the *compressed* data. However, distance estimation when the data are represented using different sets of coefficients is still a largely unexplored area. This work studies the optimization problems related to obtaining the *tightest* lower/upper bound on the distance based on the available information. In particular, we consider the problem where a distinct set of coefficients is maintained for each sequence, and the L_2-norm of the compression error is recorded. We establish the properties of optimal solutions, and leverage the theoretical analysis to develop a fast algorithm to obtain an *exact* solution to the problem. The suggested solution provides the tightest provable estimation of the L_2-norm or the correlation, and executes at least two order of magnitudes faster than a numerical solution based on convex optimization. The contributions of this work extend beyond the purview of periodic data, as our methods are applicable to any sequential or high-dimensional data as well as to any orthogonal data transformation used for the underlying data compression scheme.

Keywords: Distance estimation, Compression, Orthogonal bases, Time series, Fourier, Wavelets, KKT conditions, Water-filling algorithm

1 Introduction

A perennial problem in data analysis is the increasing dataset sizes. This trend dictates the need not only for more efficient compression schemes, but also for analytic operations that work directly on the compressed data. Efficient compression schemes can be designed based on exploiting inherent patterns and structures in the data. Data periodicity is one such characteristic that can significantly boost compression.

Periodic behavior is omnipresent; many types of collected measurements exhibit periodic patterns, including weblog data [1, 2], network measurements [3], environmental and natural processes [4, 5], medical and physiological measurements (e.g., ECG data). The aforementioned are only a few of the numerous scientific and industrial fields that handle periodic data. Examples from these areas are displayed in Fig. 1.

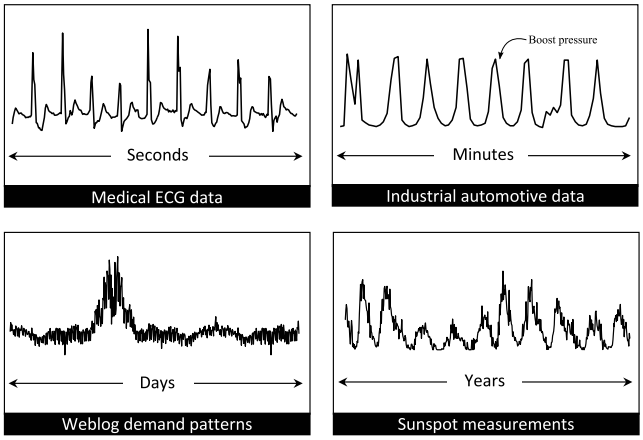

Figure 1: Many scientific fields entail periodic data. Examples from medical, industrial, web and astronomical measurements.

When data contain inherent structure, efficient compression can be performed with minimal loss in data quality (see Fig. 2 for an example). This is achievable by encoding the data using only few high-energy coefficients in a complete orthonormal basis representation, e.g., Fourier, Wavelets, Principal Component Analysis (PCA), etc. Our work focuses on the following problem: given data that are compressed in such a way, *how can we estimate distances among the original (uncompressed) data with the highest possible fidelity*?

Assuming two compressed objects, we address the problem of providing the tightest possible upper and lower bounds on the original distance between the uncompressed objects. By *tightest* we mean that given the information that we have no better estimate can be derived. Distance estimation is fundamental for data mining, because the majority of mining and learning tasks are distance-based, including clustering (e.g. k-Means or hierarchical), k-NN classification, outlier detection, pattern matching, etc. This work focuses on the case when the distance is the widely used Euclidean distance (L_2-norm of the difference), but makes further assertions for applicability to other distances. Our main **contributions** can be summarized as follows:

*Nikolaos M. Freris and Michail Vlachos are with the Department of Mathematical and Computational Sciences, IBM Research-Zürich, Switzerland.

†Suleyman S. Kozat is with the Department of Electrical and Electronics Engineering, Koc University, Istanbul, Turkey.

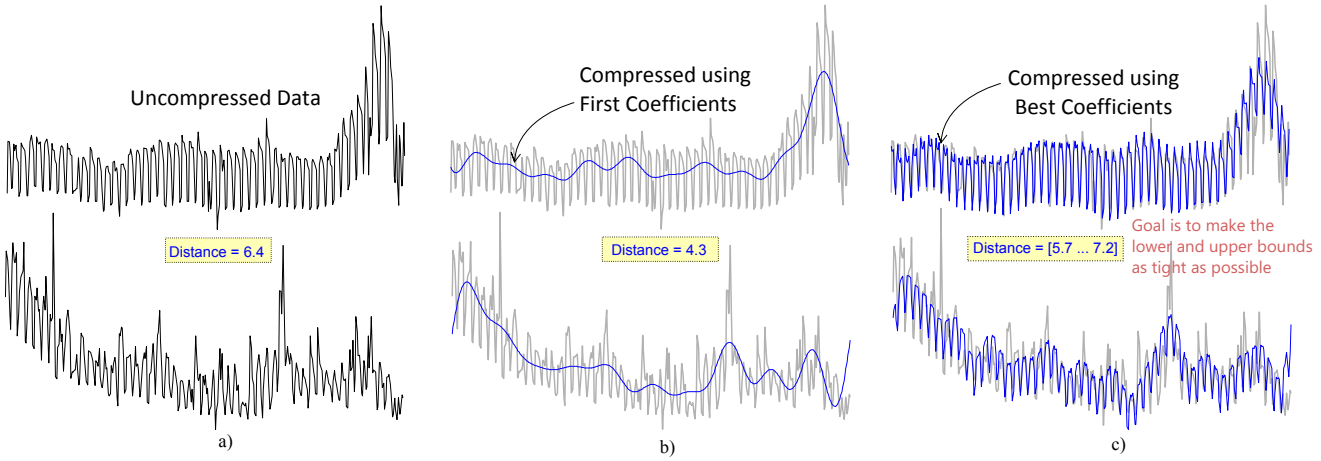

Figure 2: Motivation for using the high-energy (best) coefficients for compression. Using the best 10 coefficients (c) results in significantly better sequence approximation than when using the first coefficients (b). The goal of this work is to provide the *tightest* possible lower and upper distance estimates.

- We formulate the problem of tight distance estimation as two optimization problems for obtaining lower/upper bounds. We show that both problems can be solved simultaneously by solving a single convex optimization program.

- We provide the necessary and sufficient Karush-Kuhn-Tucker (KKT) conditions for an optimal solution and study the properties of optimal solutions.

- We use the analysis to derive exact algorithms for obtaining the optimal lower/upper bounds.

- We evaluate our analytical findings empirically; we compare the proposed algorithms with prevalent distance estimation schemes, and demonstrate significant improvements in terms of estimation accuracy. We further compare the performance of our optimal algorithm with that of a numerical scheme based on convex optimization, and show that our scheme is at least two orders of magnitude faster, while also providing more accurate results.

We emphasize that the estimated lower/upper bounds on the distance are *optimally* tight, so as to minimize the uncertainty on the distance estimation. This implies in turn that our scheme will least impact any distance-based mining operation operating directly on the compressed data.

2 Related Work

The majority of data compression techniques for sequential data use the *same* set of low-energy coefficients whether using Fourier [6, 7], Wavelets [8, 9] or Chebyshev polynomials [10] as the orthogonal basis for representation and compression. Using the same set of orthogonal coefficients has several advantages: a) it is immediate to compare the respective coefficients, b) space-partitioning indexing structures (such as R-trees) can be directly used on the compressed data, and c) there is no need to store also the indices of the basis functions that the stored coefficients correspond to. The disadvantage is that both object reconstruction and distance estimation may be *far from optimal* for a given fixed compression ratio.

One can also record side-information, such as the energy of the discarded coefficients, to better approximate the distance between compressed sequences by exploiting the Cauchy-Schwartz inequality [13]. This is shown in Figure 3a). In [11, 12], the authors advocated the use of high-energy coefficients and side-information on the discarded coefficients for weblog sequence repositories; in that setting one of the sequences was compressed, whereas the query was uncompressed, i.e., all coefficients were available as illustrated in Figure 3b). This work examines the most general and challenging case when both series are compressed. In that case, we record a (generally) different set of high-energy coefficients and also store aggregate side-information, such as the energy of the omitted data; this is depicted in Figure 3c). We are not aware of any previous art addressing this problem to derive either optimal or suboptimal bounds on distance estimation.

3 Searching Data Using Distance Estimates

We consider a database $\mathcal{DB}$ that stores sequences as V high-dimensional complex vectors $x^{(i)} \in \mathbb{C}^N, i = 1 \dots V$. The search problem that we examine can be abstracted as follows: a user is interested in finding the k most 'similar' sequences to a given query sequence $q \in \mathcal{DB}$, under a certain distance metric $d(\cdot,\cdot) : \mathbb{C}^{N\times N} \rightarrow \mathbb{R}_+$. This is the most basic yet the most

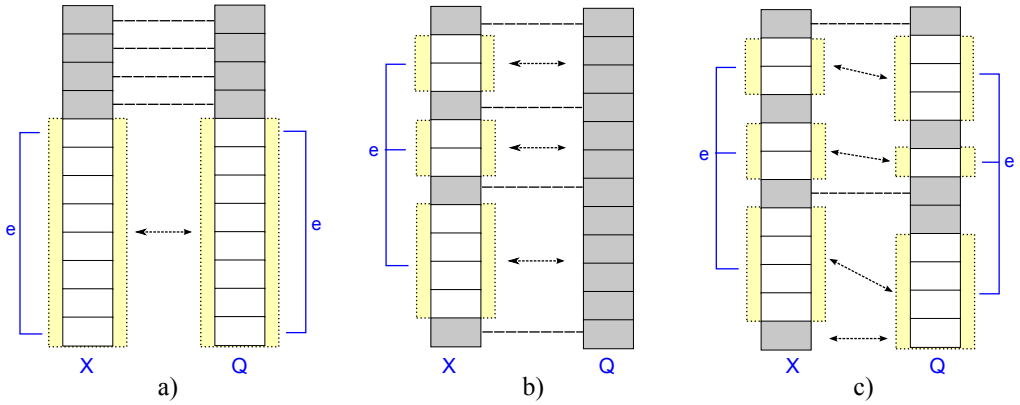

Figure 3: Comparison with previous work. Distance estimation between a compressed sequence (X) and a query (Q) represented in *any* complete orthonormal basis. A compressed sequence is represented by a set of stored coefficients (gray) as well as the error e incurred because of compression (yellow). a) Both X,Q are compressed by storing the first coefficients. b) Using the highest-energy coefficients for X, whereas Q is uncompressed as in [11, 12] , and c) the problem we address: both sequences are compressed using the highest-energy coefficients. Note that in general for each object a different set of coefficients is used.

fundamental search and mining operation, known as k-Nearest-Neighbor (k-NN) search. It is a core function in database querying, as well as a fundamental operation in a variety of data-mining and machine-learning algorithms including classification (NN-classifier), clustering, etc. In this paper, we focus on the case where d is the standard Euclidean distance, i.e., the L_2 norm on $\mathbb{C}^N$. We note that other measures, for example time-invariant matching, can be formulated as Euclidean distance on the periodogram [14]. Correlation r can also be expressed as an instance of Euclidean distance on properly normalized sequences [15]. Therefore, our approach is applicable on a wide range of distance measures with little or no modification. However, for ease of exposition, we focus on the Euclidean distance which is the most prevalent measure in the literature [16].

Search operations can be quite costly, especially for cases where the dimension N of the sequences is high, because sequences need to be retrieved from the disk for comparison against the query q. An effective way to mitigate this is to retain a compressed representation of the sequences to be used as an initial pre-filtering step. The set of compressed sequences could be small enough to keep in-memory, hence enabling a significant performance speedup. In essence, this is a *multilevel* filtering mechanism. With only the compressed sequences available, we obviously cannot infer the exact distance between the query q and a sequence $x^{(i)}$ in the database. However, it is still plausible to obtain under-estimates and over-estimates of the distance, i.e., *lower* and *upper bounds*. Using these bounds, a superset of the k-NN answers can be returned, which will be then verified using the uncompressed sequences that will need to be fetched and compared with the query, so that the exact distances can be computed. Such filtering ideas are used in the majority of the data-mining literature for speeding up search operations [6, 7, 17].

4 Notation

Consider a sequence $\mathbf{x} = \{x_1, x_2, \ldots, x_N\} \in \mathbb{R}^N$. For compression purposes, $\mathbf{x}$ is projected onto a subset of orthonormal bases $\{\mathbf{E}_1, \mathbf{E}_2, \ldots, \mathbf{E}_N\} \subset S$, where we restrict attention to the cases with $S = \mathbb{R}^N$ or $S = \mathbb{C}^N$ for most practical cases. We have

$$\mathbf{x} = \sum_{l=1}^{N} X_l \mathbf{E}_l$$

The vector $\mathbf{X} := \{X_1, X_2, \cdots, X_N\} \subset S$ is defined by

$$X_l =< \mathbf{x}, \mathbf{E}_l >= \mathbf{E}_l^* \mathbf{x} := \sum_{j=1}^{N} \mathbf{x}_k \bar{E}_{lj}$$

where we use the notation $< \cdot, \cdot >$ to denote the standard inner product in $\mathbb{C}^N$, "$*$" for complex transpose and "$\bar{}$" for the conjugate of a complex number; E_{lj} is the j-entry of vector $\mathbf{E}_l$. We denote the linear mapping $\mathbf{x} \rightarrow \mathbf{X}$ given by (4) by $\mathcal{F}$, and the inverse linear map $\mathbf{X} \rightarrow \mathbf{x}$ given by (4) by $\mathcal{F}^{-1}$, i.e., we say $\mathbf{X} = \mathcal{F}(\mathbf{x})$ and $\mathbf{x} = \mathcal{F}^{-1}(\mathbf{X})$. Examples for the invertible linear transformation that are of practical interest include e.g., Discrete Fourier Transform (DFT), PCA, Wavelets Karhunen-Loève expansion, etc.

As a running example for this paper we assume that a sequence is compressed using DFT. Therefore the basis represent sinusoids of different frequencies, and the

corresponding orthonormal basis is given by

$$\mathbf{E}_l = \left\{ \frac{1}{\sqrt{N}} e^{i2\pi kj/N} \right\}_{j=0}^{N-1}$$

In such a case, the pair $(\mathbf{x}, \mathbf{X})$, where $\mathbf{X} = DFT(\mathbf{x})$ and $\mathbf{x} = IDFT(\mathbf{X})$, the inverse DFT, satisfies

$$X_l = 1/\sqrt{N} \sum_{k=1}^{N} x_k e^{i2\pi(k-1)(l-1)/N},\ l = 1, \ldots, N$$

$$x_k = 1/\sqrt{N} \sum_{l=1}^{N} X_l e^{i2\pi(k-1)(l-1)/N},\ k = 1, \ldots, N$$

where i is the imaginary unit $i^2 = -1$. As distance between two sequences $\mathbf{x}$, $\mathbf{q}$ we assume the L_2-norm, which can easily be translated into distance in the frequency domain because of Parseval's theorem:

$$d(\mathbf{x}, \mathbf{q}) := ||\mathbf{x} - \mathbf{q}||_2 = ||\mathbf{X} - \mathbf{Q}||_2$$

5 Motivation

The choice of which coefficients to use has a direct impact on the data approximation quality. Although it has long been recognized that sequence approximation when using high-energy (i.e., best) coefficients is indeed superior [18, 11] - see also Figure 2 for an illustrative example - a barrier still has to be overcome: efficiency of solution for distance estimation.

Consider a sequence represented using its high-energy coefficients. Therefore, the compressed sequence $\mathbf{X}$ will be described by a set of C_x coefficients that hold the largest energy. We denote the vector describing the positions of those coefficients in $\mathbf{X}$ as p_x^+, and the positions of the remaining ones as p_x^- (that is $p_x^+ \cup p_x^- = [1, \ldots, N]$). For any sequence $\mathbf{X}$, we store in the database the vector $\mathbf{X}(p_x^+)$, which we denote simply by $\mathbf{X}^+ := \{X_i\}_{i \in p_x^+}$. We denote the vector of discarded coefficients by $\mathbf{X}^- := \{X_i\}_{i \in p_x^-}$.

In addition to the best coefficients of a sequence, we can also record one additional value for the energy of the compression error, $e_x = ||\mathbf{X}^-||_2^2$, i.e., the sum of squared magnitudes of the omitted coefficients.

Then one needs to solve the following minimization (maximization) problem for calculating the lower (upper) bounds on the distance between two sequences based on their compressed versions:

$$\begin{aligned} (5.1) \qquad \min(\max) \quad & ||\mathbf{X} - \mathbf{Q}||_2 \\ \text{s.t.} \quad & |X_l| \leq \min_{j \in p_x^+} |X_j|, \quad \forall l \in p_x^- \\ & |Q_l| \leq \min_{j \in p_q^+} |Q_j|, \quad \forall l \in p_q^- \\ \text{and} \quad & \sum_{l \in p_x^-} |X_l|^2 = e_x, \sum_{l \in p_q^-} |Q_l|^2 = e_q \\ & \mathbf{X}^- \in \mathbb{C}^{|p_x^-|}, \mathbf{Q}^- \in \mathbb{C}^{|p_q^-|} \end{aligned}$$

where the decision variables are the vectors $\mathbf{X}^-, \mathbf{Q}^-$. The constraints are due to the fact that we use the high-energy components for the compression. Hence, any of the omitted components must have energy lower than the minimum energy of any kept component.

The optimization problem presented is a complex-valued program: the minimization problem can easily be recast as an equivalent *convex program* by relaxing the equality constraints into $\leq$ inequality constraints, as will be justified. Hence, it can be solved efficiently with numerical methods [19], cf. Sec. 8.1. However, as we show in the experimental section, evaluating an instance of this problem just for a pair of sequences is not efficient in practice: it requires approximately one second on a modern CPU. Therefore, although a solution can be found *numerically*, it is generally costly and not tailored for large mining tasks where we would like to evaluate thousands or millions of such lower/upper bounds on compressed sequences. Here we show how solve this problem **analytically** by exploiting the derived optimality conditions. More importantly, we show in our experiments that our approach is more than two orders of magnitude faster than numerical solutions.

We solve this problem as a 'double water-filling' instance. Vlachos et al. have shown how the optimal lower and upper distance bounds between a compressed and an uncompressed sequence can be relegated to a single water-filling problem [11]. We revisit this approach as it will be used as a building block for our solution. In addition, we later derive optimality properties for our solution.

6 An Equivalent Convex Optimization Problem

For ease of notation, we consider the partition $\mathcal{P} = \{P_0, P_1, P_2, P_3\}$ of $\{1, \ldots, N\}$ (see Fig. 4), where we set the following:

- $P_0 = p_x^+ \cap p_q^+$ are the common known components in two compressed sequences $\mathbf{X}, \mathbf{Q}$.
- $P_1 = p_x^- \cap p_q^+$ are the components unknown for $\mathbf{X}$ but known for $\mathbf{Q}$.
- $P_2 = p_x^+ \cap p_q^-$ are the components known for $\mathbf{X}$ but unknown for $\mathbf{Q}$.
- $P_3 = p_x^- \cap p_q^-$ are the components unknown for both sequences.

Using the standard notation $\mathbf{x}^*$ for the *conjugate transpose* of a complex vector $\mathbf{x}$, $\Re$ to denote the real part of a complex number, and considering all vectors

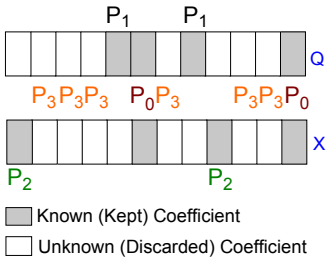

Figure 4: Visual illustration of sets P_0, P_1, P_2, P_3 between two compressed objects.

as column vectors, we have that the squared Euclidean distance is given by

$$\begin{aligned}||\mathbf{x}-\mathbf{q}||_2^2 &= ||\mathbf{X}-\mathbf{Q}||_2^2 = (\mathbf{X}-\mathbf{Q})^*(\mathbf{X}-\mathbf{Q}) \\ &= ||\mathbf{X}||_2^2 + ||\mathbf{Q}||_2^2 - 2\mathbf{X}^*\mathbf{Q} \\ &= ||\mathbf{X}||_2^2 + ||\mathbf{Q}||_2^2 - 4\sum_{i=1}^{N}\Re\{X_iQ_i\} \\ &= ||\mathbf{X}||_2^2 + ||\mathbf{Q}||_2^2 - 4(\sum_{l\in P_0}\Re\{X_lQ_l\} \\ &\quad +\sum_{l\in P_1}\Re\{X_lQ_l\} + \sum_{l\in P_2}\Re\{X_lQ_l\} \\ &\quad +\sum_{l\in P_3}\Re\{X_lQ_l\}).\end{aligned}$$

Note that $||\mathbf{X}||_2, ||\mathbf{Q}||_2$ can be inferred by summing the squared magnitudes of the known coefficients with the energy of the compression error. Also, the term $\sum_{l\in P_0}\Re\{X_lQ_l\}$ is known, whereas the last three sums are unknown. Considering the polar form, i.e., absolute value $|\cdot|$ and argument $\arg(\cdot)$

$$X_l = |X_l|e^{i\arg(X_l)}, \quad Q_l = |Q_l|e^{i\arg(Q_l)}$$

we have that the decision variables are vectors $|X_l|, \arg(X_l), l \in p_x^-$ as well as $|Q_l|, \arg(Q_l), l \in p_q^-$. Observe that for $x, y \in \mathbb{C}$ with $|x|, |y|$ known, we have that $-|x||y| \le \Re\{xy\} \le |x||y|$, where the upper bound is attained when $\arg(x) + \arg(y) = 0$ and the lower bound when $\arg(x) + \arg(y) = \pi$. Therefore, both problems (5.1) boil down to the real-valued optimization problem

$$\begin{aligned}(6.2)\qquad \min \quad & -\sum_{l\in P_1} a_lb_l - \sum_{l\in P_2} a_lb_l - \sum_{l\in P_3} a_lb_l \\ \text{s.t.} \quad & 0 \le a_l \le A, \quad \forall l \in p_x^- \\ & 0 \le b_l \le B, \quad \forall l \in p_q^- \\ & \sum_{l\in p_x^-} a_l^2 \le e_x \\ & \sum_{l\in p_q^-} b_l^2 \le e_q,\end{aligned}$$

where a_l, b_l represent $|X_l|, |Q_l|$, respectively, and $A := \min_{j\in p_q^+}|X_j|, B := \min_{j\in p_q^+}|Q_j|$. Note also that we have relaxed the equality constraints to inequality constraints as the objective function of (6.2) is decreasing in all a_i, b_i, so the optimum of (6.2) has to satisfy the relaxed inequality constraints with equality, because of the elementary property that $|p_x^-|A^2 \ge e_x, |p_q^-|B^2 \ge e_q$. Recall that in the first sum only $\{a_i\}$ are known, in the second only $\{b_i\}$ are known, and in the third all variables are unknown.

We have reduced the original problem to a single optimization program, which is, however, not convex unless $p_x^- \cap p_q^- = \emptyset$. It is easy to check that the constraint set is convex and compact, however, the bilinear function $f(x,y) := xy$ is convex in each argument alone, but *not* jointly. We consider the re-parametrization of the decision variables $z_i = a_i^2$, for $i \in p_x^-$ and $y_i = b_i^2$ for $i \in p_q^-$, we set $Z := A^2, Y := B^2$ and get the equivalent problem:

$$\begin{aligned}(6.3)\qquad \min \quad & -\sum_{i\in P_1} b_i\sqrt{z}_i - \sum_{i\in P_2} a_i\sqrt{y}_i - \sum_{i\in P_3}\sqrt{z}_i\sqrt{y}_i \\ \text{s.t.} \quad & 0 \le z_i \le Z, \quad \forall i \in p_x^- \\ & 0 \le y_i \le Y, \quad \forall i \in p_q^- \\ & \sum_{i\in p_x^-} z_i \le e_x \\ & \sum_{i\in p_q^-} y_i \le e_q\ .\end{aligned}$$

Existence of solutions and necessary and sufficient conditions for optimality:
The constraint set is a compact convex set, in fact, a compact *polyhedron*. The function $g(x,y) := -\sqrt{x}\sqrt{y}$ is convex but not strictly convex on $\mathbb{R}_+^2$. To see this, note that the *Hessian* exists for all $x, y > 0$ and equals

$$\nabla^2 g = \frac{1}{4}\begin{pmatrix} x^{-\frac{3}{2}}y^{-\frac{1}{2}} & -x^{-\frac{1}{2}}y^{-\frac{1}{2}} \\ -x^{-\frac{1}{2}}y^{-\frac{1}{2}} & x^{-\frac{1}{2}}y^{-\frac{3}{2}} \end{pmatrix}$$

with eigenvalues $0, \frac{1}{\sqrt{xy}}(\frac{1}{x}+\frac{1}{y})$, and hence is positive semi-definite, which in turn implies that g is convex [19]. Furthermore, $-\sqrt{x}$ is a strictly convex function of x so that the objective function of (6.3) is convex, and strictly convex only if $p_x^- \cap p_q^- = \emptyset$. It is also a continuous function so solutions exist, i.e., the optimal value is bounded and is attained. It is easy to check that the *Slater condition* holds, whence the problem satisfies *strong duality* and there exist Lagrange multipliers [19]. We skip the technical details for simplicity, but we want to highlight that this property is substantial because it guarantees that the Karush-Kuhn-Tucker (KKT) necessary conditions [19] for Lagrangian optimality are also *sufficient*. Therefore, if we can find a solution to satisfy the KKT conditions for the problem, we have

found an *exact* optimal solution and the *exact* optimal value of the problem. The Lagrangian is

(6.4)

$$L(\mathbf{y},\mathbf{z},\lambda,\mu,\boldsymbol{\alpha},\boldsymbol{\beta}) := -2\sum_{i\in P_1} b_i\sqrt{z}_i - 2\sum_{i\in P_2} a_i\sqrt{y}_i - 2\sum_{i\in P_3}\sqrt{z}_i\sqrt{y}_i + \lambda\Big(\sum_{i\in p_x^-}(z_i-e_x)\Big) + \mu\Big(\sum_{i\in p_q^-}(y_i-e_q)\Big) + \sum_{i\in p_x^-}\alpha_i(z_i-Z) + \sum_{i\in p_q^-}\beta_i(y_i-Y)\ .$$

The KKT conditions are as follows[1]:

(6.5a)
$$0\le z_i\le Z,\ 0\le y_i\le Y, \quad \text{(PF)}$$
$$\sum_{i\in p_x^-} z_i\le e_x,\ \sum_{i\in p_x^-} z_i\le e_Q$$

(6.5b)
$$\lambda,\mu,\alpha_i,\beta_i\ge 0 \quad \text{(DF)}$$

(6.5c)
$$\alpha_i(z_i-Z)=0,\ \ \beta_i(y_i-Y)=0 \quad \text{(CS)}$$
$$\lambda\Big(\sum_{i\in p_x^-}(z_i-e_x)\Big)=0,\ \ \mu\Big(\sum_{i\in p_q^-}(y_i-e_q)\Big)=0$$

(6.5d)
$$i\in P_1:\ \frac{\partial L}{\partial z_i}=-\frac{b_i}{\sqrt{z_i}}+\lambda+\alpha_i=0 \quad \text{(O)}$$
$$i\in P_2:\ \frac{\partial L}{\partial y_i}=-\frac{a_i}{\sqrt{y_i}}+\mu+\beta_i=0$$
$$i\in P_3:\ \frac{\partial L}{\partial z_i}=-\frac{\sqrt{y_i}}{\sqrt{z_i}}+\lambda+\alpha_i=0$$
$$\frac{\partial L}{\partial y_i}=-\frac{\sqrt{z_i}}{\sqrt{y_i}}+\mu+\beta_i=0\ ,$$

where we use shorthand notation for *Primal Feasibility* (PF), *Dual Feasibility* (DF), *Complementary Slackness* (CS), and *Optimality* (O) [19].

Let us denote the optimal value of (6.3) by $v_{\text{opt}}\le 0$. Then the optimal lower bound (LB) and upper bound (UB) for the distance estimation problem under consideration are given by

$$LB = \sqrt{\hat{D}+4v_{\text{opt}}} \tag{6.6}$$
$$UB = \sqrt{\hat{D}-4v_{\text{opt}}} \tag{6.7}$$
$$\hat{D} := ||X||_2^2+||Q||_2^2-4\sum_{l\in P_0}\Re\{X_lQ_l\}\ .$$

7 Exact Solutions

In this section, we study algorithms for obtaining exact solutions for the optimization problem (6.3). By *exact*, we mean that the optimal value is obtained in a finite number of computations as opposed to when using a numerical scheme for convex optimization. In the latter case, an approximate solution is obtained by means of an iterative scheme which converges with finite precision. Before addressing the general problem, we briefly recap a special case that was dealt with in [11], where the sequence $\mathbf{Q}$ was assumed to be uncompressed. In this case, an exact solution is provided via the *water-filling* algorithm, which will constitute a key building block for obtaining exact solutions to the general problem later on. We then proceed to study the properties of optimal solutions; our theoretical analysis gives rise to an *exact* algorithm, cf. Sec. 8.2.

7.1 Water-filling Algorithm.

The case that $\mathbf{Q}$ is uncompressed is a special instance of our problem with $p_q^-=\emptyset$, whence also $P_2=P_3=\emptyset$. The problem is strictly convex, and (6.5d) yields

$$z_i=\Big(\frac{b_i}{\lambda+\alpha_i}\Big)^2 \Leftrightarrow a_i=\frac{b_i}{\lambda+\alpha_i} \tag{7.8}$$

In such a case, the strict convexity guarantees the existence of a *unique* solution satisfying the KKT conditions as given by the *water-filling* algorithm, cf. Fig. 5. The algorithm progressively increases the unknown coefficients a_i until saturation, i.e., until they reach A, in which case they are fixed. The set C is the set of non-saturated coefficients at the beginning of each iteration, while R denotes the "energy reserve," i.e., the energy that can be used to increase the non-saturated coefficients; v_{opt} denotes the optimal value.

As a shorthand notation, we write $\mathbf{a}=\text{waterfill}(\mathbf{b},e_x,A)$. Note that in this case the problem (6.2) for $P_2=P_3=\emptyset$ is convex, so the solution can be obtained via the KKT conditions to (6.2), which are different from those for the re-parameterized problem (6.3); this was done in [11]. The analysis and straightforward extensions are summarized in Lemma 7.1.

Lemma 7.1. (Exact solutions)

1. *If either $p_x^-=\emptyset$ or $p_q^-=\emptyset$ (i.e., when at least one of the sequences is uncompressed) we can obtain an exact solution to the optimization problem (6.2) via the water-filling algorithm.*
2. *If $P_3=p_x^-\cap p_q^-=\emptyset$, i.e., when the two compressed sequences do not have any common unknown coefficients, the problem is decoupled in $\mathbf{a},\mathbf{b}$ and the water-filling algorithm can be used separately to obtain exact solutions to both unknown vectors.*
3. *If $P_1=P_2=\emptyset$, i.e., when both compressed sequences have the same discarded coefficients, the optimal value is simply equal to $-\sqrt{e_x}\sqrt{e_q}$, but there is no unique solution for $\mathbf{a},\mathbf{b}$.*

[1] The condition (6.5d) excludes the cases that for some i $z_i=0$, or $y_i=0$ which will be treated separately in the following.

Water-filling algorithm
Inputs: $\{b_i\}_{i\in p_x^-}, e_x, A$
Outputs: $\{a_i\}_{i\in p_x^-}, \lambda, \{\alpha_i\}_{i\in p_x^-}, v_{\text{opt}}, R$

1. Set $R = e_x,\ C = p_x^-$
2. **while** $R > 0$ and $C \neq \emptyset$ do
3. set $\lambda = \sqrt{\frac{\sum_{i\in C} b_i^2}{R}},\ \ a_i = \frac{b_i}{\lambda},\ i \in C$
4. **if** for some $i \in C,\ a_i > A$ then
5. $a_i = A,\ C \leftarrow C - \{i\}$
6. **else break**;
7. **end if**
8. $R = e_x - (|p_x^-| - |C|)A^2$
9. **end while**
10. Set $v_{\text{opt}} = -\sum_{i\in p_x^-} a_i b_i$ and

$$\alpha_i = \begin{cases} 0, & \text{if } a_i < A \\ \frac{b_i}{A} - \lambda, & \text{if } a_i = A \end{cases}$$

Figure 5: Water-filling algorithm for optimal distance estimation between a compressed and an uncompressed sequence

Proof. The first two cases are obvious. For the third one, note that it follows immediately from the Cauchy-Schwartz inequality that $-\sum_{l\in P_3} a_l b_l \geq -\sqrt{e_x}\sqrt{e_q}$ and in this is case this is also attainable, e.g., just consider $a_l = \sqrt{\frac{e_x}{|P_3|}}, b_l = \sqrt{\frac{e_q}{|P_3|}}$, which is feasible because $|p_x^-|A^2 \geq e_x, |p_q^-|B^2 \geq e_q$, as follows by compression with the high-energy coefficients. ■

We have shown how to obtain exact optimal solutions for special cases. To derive efficient algorithms for the general case, we first study end establish some properties of the optimal solution of (6.3).

THEOREM 7.1. (PROPERTIES OF OPTIMAL SOLUTIONS)

Let an augmented optimal solution of (6.2) be denoted by $(\mathbf{a}^{opt}, \mathbf{b}^{opt})$; where $\mathbf{a}^{opt} := \{a_i^{opt}\}_{i\in p_x^- \cup p_q^-}$ denotes the optimal solution extended to include the known values $|X_l|_{l\in P_2}$, and $\mathbf{b}^{opt} := \{b_i^{opt}\}_{i\in p_x^- \cup p_q^-}$ denotes the optimal solution extended to include the known values $|Q_l|_{l\in P_1}$. Let us further define $e'_x = e_x - \sum_{l\in P_1} a_l^2, e'_q = e_q - \sum_{l\in P_2} b_l^2$. We then have the following:

1. *The optimal solution satisfies*[2]

$$\text{(7.9a)} \qquad \mathbf{a}^{opt} = \textit{waterfill}\ (\mathbf{b}^{opt}, e_x, A)$$
$$\text{(7.9b)} \qquad \mathbf{b}^{opt} = \textit{waterfill}\ (\mathbf{a}^{opt}, e_q, B)$$

In particular, it follows that $a_i^{opt} > 0$ iff $b_i^{opt} > 0$ and that $\{a_i^{opt}\}, \{b_i^{opt}\}$ have the same ordering. In addition, $\min_{l\in P_1} a_l \geq \max_{l\in P_3} a_l, \min_{l\in P_2} b_l \geq \max_{l\in P_3} b_l$.

2. *If at optimality it holds that $e'_x e'_q > 0$ there exists a multitude of solutions. One solution $(\mathbf{a}, \mathbf{b})$ satisfies $a_l = \sqrt{\frac{e'_x}{|P_3|}}, b_l = \sqrt{\frac{e'_q}{|P_3|}}$ for all $l \in P_3$, whence*

$$\text{(7.10a)} \qquad \lambda = \sqrt{\frac{e'_q}{e'_x}} \quad \mu = \sqrt{\frac{e'_x}{e'_q}}$$
$$\text{(7.10b)} \qquad \alpha_i = \beta_i = 0\ \forall i \in P_3$$

In particular, $\lambda\mu = 1$ and the values e'_x, e'_q need to be solutions to the following set of nonlinear equations:

$$\text{(7.11a)} \qquad \sum_{l\in P_1} \min\left(b_l^2 \frac{e'_x}{e'_q}, A^2\right) = e_x - e'_x$$
$$\text{(7.11b)} \qquad \sum_{l\in P_2} \min\left(a_l^2 \frac{e'_q}{e'_x}, B^2\right) = e_q - e'_q$$

3. *At optimality, it is not possible to have $e'_x e'_q = 0$ unless $e'_x = e'_q = 0$.*
4. *Consider the vectors $\mathbf{a}, \mathbf{b}$ with $a_l = |X_l|,\ l \in P_2, a_l = |X_l|,\ l \in P_1$ and*

$$\text{(7.12a)} \qquad \{a_l\}_{l\in P_1} = \textit{waterfill}\ (\{b_l\}_{l\in P_1}, e_x, A)$$
$$\text{(7.12b)} \qquad \{b_l\}_{l\in P_2} = \textit{waterfill}\ (\{a_l\}_{l\in P_2}, e_q, B)$$

If $e_x \leq |P_1|A^2$ and $e_q \leq |P_2|B^2$, whence $e'_x = e'_q = 0$, then by defining $a_l = b_l = 0$ for $l \in P_3$, we obtain a globally *optimal solution $(\mathbf{a}, \mathbf{b})$.*

Proof. See Appendix.

REMARK 7.1. *One may be tempted to think that an optimal solution can be derived by water-filling for the coefficients of $\{a_l\}_{l\in P_1}, \{b_l\}_{l\in P_2}$ separately, and then allocating the remaining energies e'_x, e'_q to the coefficients in $\{a_l, b_l\}_{l\in P_3}$ leveraging the Cauchy-Schwartz inequality, the value being $-\sqrt{e'_x}\sqrt{e'_q}$. However, the third and fourth parts of Theorem 7.1 state that this is* not *optimal unless $e'_x = e'_q = 0$.*

[2] This has a natural interpretation as the *Nash equilibrium* of a 2-player game [20] in which Player 1 seeks to minimize the objective of (6.3) with respect to $\mathbf{z}$, and Player 2 seeks to minimize the same objective with respect to $\mathbf{y}$.

We have shown that there are two possible cases for an optimal solution of (6.2): either $e'_x = e'_q = 0$ or $e'_x, e'_q > 0$. The first case is easy to identify by checking whether (7.12) yields $e'_x = e'_q = 0$. If this is not the case, we are in the latter case and need to find a solution to the set of non linear equations (7.11).

Consider the mapping $T : \mathbb{R}^2_+ \to \mathbb{R}^2_+$ defined by

(7.13)
$$T((x_1, x_2)) := \left(e_x - \sum_{l \in P_1} \min\left(b_l^2 \frac{x_1}{x_2}, A^2\right),\ e_q - \sum_{l \in P_2} \min\left(a_l^2 \frac{x_2}{x_1}, B^2\right) \right)$$

The set of non linear equations of (7.11) corresponds to a positive *fixed point* of T, i.e., $(e'_x, e'_q) = T(e'_x, e'_q), e'_x, e'_q > 0$. As this problem is of interest only if $e'_x e'_q > 0$ at optimality, we know that we are not in the setup of Theorem 7.1.4, therefore we have the additional property that either $e_x > |P_1|A^2$, $e_q > |P_2|B^2$ or both. Let us define

(7.14)
$$\gamma_a := \min\left\{\gamma \geq 0 : \sum_{l \in P_2} \min\left(a_l^2 \frac{1}{\gamma}, B^2\right) \leq e_q\right\}$$
$$\gamma_b := \max\left\{\gamma \geq 0 : \sum_{l \in P_1} \min\left(b_l^2 \gamma, A^2\right) \leq e_x\right\}$$

Clearly if $e_x > |P_1|A^2$ then $\gamma_b = +\infty$ and for any $\gamma \geq \max_{l \in P_1} \frac{A^2}{b_l^2}$ we have $\sum_{l \in P_1} \min(b_l^2 \gamma, A^2) = |P_1|A^2$; similarly, if $e_q > |P_2|B^2$ then $\gamma_a = 0$, and for any $\gamma \leq \min_{l \in P_2} \frac{a_l^2}{B^2}$ we have $\sum_{l \in P_2} \min(a_l^2 \frac{1}{\gamma}, B^2) = |P_2|B^2$. If $\gamma_b < +\infty$, we can find the exact value of γ_b analytically by sorting $\{\gamma_l^{(b)} := \frac{A^2}{b_l^2}\}_{l \in P_1}$ in increasing order and considering

$$h_b(\gamma) := \sum_{l \in P_1} \min(b_l^2 \gamma_i^{(b)}, A^2) - e_x$$

and $v_i := h_b(\gamma_i^{(b)})$. In this case, $v_1 < \ldots < v_{|P_1|}$, and $v_{|P_1|} > 0$, and there are two possibilities: 1) $v_1 > 0$ whence $\gamma_b < \gamma_1^{(b)}$, or 2) there exists some i such that $v_i < 0 < v_{i+1}$ whence $\gamma_i^{(b)} < \gamma_b < \gamma_{i+1}^{(b)}$. For both ranges of γ, the function h becomes *linear* and strictly increasing, and it is elementary to compute its root γ_b. A similar argument applies for calculating γ_a if γ_a is strictly positive, by defining h_a .

THEOREM 7.2. (EXACT SOLUTION OF (7.11))
If either $e_x > |P_1|A^2$, $e_q > |P_2|B^2$ or both, then the non linear mapping T has a unique fixed point (e'_x, e'_q) with $e'_x, e'_q > 0$. The equation

(7.15)
$$\frac{e_x - \sum_{l \in P_1} \min(b_l^2 \gamma, A^2)}{e_q - \sum_{l \in P_2} \min(a_l^2 \frac{1}{\gamma}, B^2)} = \gamma$$

has a unique solution $\bar{\gamma}$ with $\gamma_a \leq \bar{\gamma}$ and $\gamma_a \leq \gamma_b$ when $\gamma_b < +\infty$. The unique fixed point of T (solution of (7.11)) satisfies

(7.16)
$$e'_x = e_x - \sum_{l \in P_1} \min\left(b_l^2 \bar{\gamma}, A^2\right)$$
$$e'_q = e_q - \sum_{l \in P_2} \min\left(a_l^2 \frac{1}{\bar{\gamma}}, B^2\right)$$

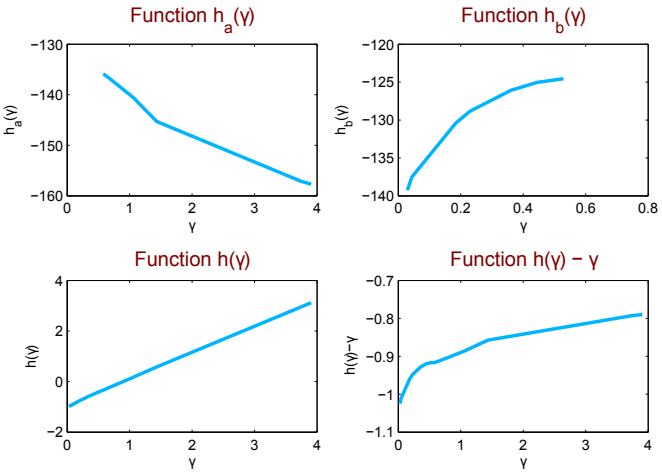

Figure 6: A plot of functions h_a, h_b, h; (top) h_a is a bounded decreasing function, which is piecewise linear in $\frac{1}{\gamma}$ with non-increasing slope in $\frac{1}{\gamma}$; h_b is a bounded increasing piecewise linear function of γ with non-increasing slope. (bottom) h is an increasing function; the linear term γ dominates the fraction term which is also increasing, see bottom right.

Proof. Existence[3] of a fixed point is guaranteed by existence of solutions and Lagrange multiplies for (6.3), as by assumption we are in the setup of Theorem 7.1.2. Define $\gamma := \frac{e'_x}{e'_q}$; a fixed point $(e'_x, e'_q) = T((e'_x, e'_q)), e'_x, e'_q > 0$, corresponds to a root of

(7.17)
$$h(\gamma) := -\frac{e_x - \sum_{l \in P_1} \min(b_l^2 \gamma, A^2)}{e_q - \sum_{l \in P_2} \min(a_l^2 \frac{1}{\gamma}, B^2)} + \gamma$$

For the range $\gamma \geq \gamma_a$ and $\gamma \leq \gamma_b$, if $\gamma_b < +\infty$, we have that $h(\gamma)$ is continuous and strictly increasing. The facts that $\lim_{\gamma \searrow \gamma_a} h(\gamma) < 0, \lim_{\gamma \nearrow \gamma_b} h(\gamma) > 0$ show existence of a unique root $\bar{\gamma}$ of h corresponding to a unique fixed point of T, cf. (7.16). ■

REMARK 7.2. (EXACT CALCULATION OF A ROOT OF h)
We seek to calculate the root of h exactly and efficiently. In doing so, consider the points $\{\gamma_l\}_{l \in P_1 \cup P_2}$

[3] An alternative and more direct approach of establishing existence of a fixed point is by considering all possible cases and defining an appropriate compact convex set $E \subset \mathbb{R}^2_+ \setminus (0,0)$ so that $T(E) \subset E$ whence existence follows by the Brower's fixed point theorem [20], since T is continuous.

where $\gamma_l := \frac{A}{b_l^2}$, $l \in P_1$, $\gamma_l := \frac{a_l^2}{B}$, $l \in P_2$. *Then, note that for any* $\gamma \geq \gamma_l, l \in P_1$ *we have that* $\min(b_l^2\gamma, A^2) = A^2$. *Similarly, for any* $\gamma \leq \gamma_l, l \in P_2$, *we have that* $\min(a_l^2 \frac{1}{\gamma}, B^2) = B^2$. *We order all such points in increasing order, and consider the resulting vector* $\boldsymbol{\gamma}' := \{\gamma_i'\}$ *excluding any points below* γ_a *or above* γ_b. *Let us define* $h_i := h(\gamma_i')$. *If for some* i, $h_i = 0$ *we are done. Otherwise there are three possibilities: 1) there is an* i *such that* $h_i < 0 < h_{i+1}$, *2)* $h_1 > 0$ *or 3)* $h_N < 0$. *In all cases, the numerator (denominator) of* h *is linear in* γ ($\frac{1}{\gamma}$) *for the respective range of* γ; $\bar{\gamma}$ *is obtained by solving the linear equation*

(7.18)

$$e_x - \sum_{l \in P_1} \min(b_l^2\gamma, A^2) = \gamma \left(e_q - \sum_{l \in P_2} \min\left(a_l^2 \frac{1}{\gamma}, B^2\right)\right)$$

and using the elementary property that for a linear function f *on* $[x_0, x_1]$ *with* $f(x_0)f(x_1) < 0$ *the unique root is given by*

$$\bar{x} = x_0 - \frac{x_1 - x_0}{f(x_1) - f(x_0)} f(x_0) \ .$$

8 Algorithm for Optimal Distance Estimation

In this section, we present an algorithm for obtaining the *exact optimal* upper and lower bounds on the distance between the original sequences, when fully leveraging all information available given their compressed counterparts. First, we present a simple numerical scheme using a convex solver such as cvx [21] and then use our theoretical findings to derive an analytical algorithm which we call 'double water-filling'.

8.1 Convex Programming

We let $M := N - |P_0|$, and consider the non-trivial case $M > 0$. Following the discussion in Sec. 6, we set the $2M \times 1$ vector $\mathbf{v} = (\{a_l\}_{l \in P_1 \cup P_2 \cup P_3}, \{b_l\}_{l \in P_1 \cup P_2 \cup P_3})$ and consider the following convex problem directly amenable to a numerical solution via a solver such as cvx:

$$\begin{array}{ll} \min & \sum_{l \in P_1 \cup P_2 \cup P_3} (a_l - b_l)^2 \\ \text{s.t.} & a_l \leq A, \ \forall l \in p_x^-, \quad b_l \leq B, \ \forall l \in p_q^- \\ & \sum_{l \in p_x^-} a_l^2 \leq e_x, \quad \sum_{l \in p_q^-} b_l^2 \leq e_q \\ & a_l = |X_l|, \ \forall l \in P_2, \quad b_l = |Q_l|, \ \forall l \in P_1 \end{array}$$

The lower bound (LB) can be obtained by adding $D' := \sum_{l \in P_0} |X_l - Q_l|^2$ to the optimal value of (5.1) and taking the square root; then the upper bound is given by $UB = \sqrt{2D' - LB^2}$, cf. (6.6).

8.2 Double Water-filling

Leveraging our theoretical analysis, we derive a simple efficient algorithm to obtain an *exact* solution to the problem of finding tight lower/upper bound on the distance of two compressed sequences; we call this the "double water-filling algorithm." The idea is to obtain an exact solution of (6.2) based on the results of Theorems 7.1, 7.2, and Remark 7.2; then the lower/upper bounds are given by (6.6), (7.7). The algorithm is described in Fig. 7; its proof of optimality follows immediately from the preceding theoretical analysis.

Double water-filling algorithm
Inputs: $\{b_i\}_{i \in P_1}, \{a_i\}_{i \in P_2}, e_x, e_q, A, B$
Outputs: $\{a_i, \alpha_i\}_{i \in p_x^-}, \{b_i, \beta_i\}_{i \in p_q^-}, \lambda, \mu, v_{\text{opt}}$

1. **if** $p_x^- \cap p_q^- = \emptyset$ **then** use water-filling algorithm (see Lemma 7.1 parts 1,2); **return**; **endif**
2. **if** $p_x^- = p_q^-$ **then** set $a_l = \sqrt{\frac{e_x}{|P_3|}}, b_l = \sqrt{\frac{e_q}{|P_3|}}$, $\alpha_l = \beta_l = 0$ for all $l \in p_x^-$, $v_{\text{opt}} = -\sqrt{e_x}\sqrt{e_q}$; **return**; **endif**
3. **if** $e_x \leq |P_1|A^2$ **and** $e_q \leq |P_2|B^2$ **then**
   $\{a_l\}_{l \in P_1} = \text{waterfill } (\{b_l\}_{l \in P_1}, e_x, A)$
   $\{b_l\}_{l \in P_2} = \text{waterfill } (\{a_l\}_{l \in P_2}, e_q, B)$
   with optimal values $v_{\text{opt}}^{(a)}, v_{\text{opt}}^{(b)}$, respectively.
4. Set $a_l = b_l = \alpha_l = \beta_l = 0$ for all $l \in P_3$, $v_{\text{opt}} = -v_{\text{opt}}^{(a)} - v_{\text{opt}}^{(b)}$; **return**;
5. **endif**
6. Calculate the root $\bar{\gamma}$ as in Remark 7.2 and define e_x', e_q' as in (7.16).
7. Set
   $\{a_l\}_{l \in P_1} = \text{waterfill } (\{b_l\}_{l \in P_1}, e_x - e_x', A)$
   $\{b_l\}_{l \in P_2} = \text{waterfill } (\{a_l\}_{l \in P_2}, e_q - e_q', B)$
   with optimal values $v_{opt}^{(a)}, v_{opt}^{(b)}$, respectively.
8. Set $a_l = \sqrt{\frac{e_x'}{|P_3|}}, b_l = \sqrt{\frac{e_q'}{|P_3|}}, \alpha_l = \beta_l = 0, l \in P_3$ and set $v_{\text{opt}} = -v_{opt}^{(a)} - v_{opt}^{(b)} - \sqrt{e_x'}\sqrt{e_q'}$

Figure 7: Double water-filling algorithm for optimal distance estimation between two compressed sequences

9 Experiments

Here we provide convincing experimental evidence on both the tightness of the proposed bounds compared with other approaches in the literature, and on the speed compared with the numerical scheme based on

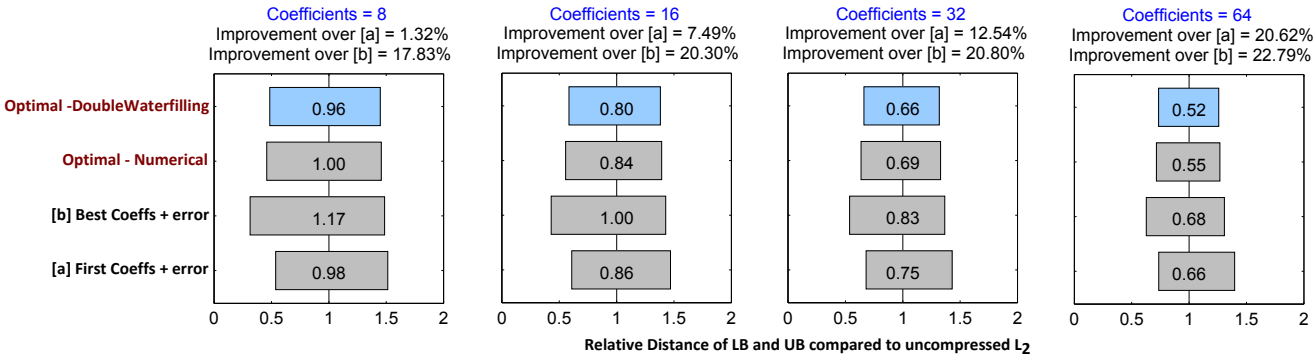

Figure 8: Comparison the Lower- (LB) and Upper-Bounds (UB) of various approaches. LB and UB are shown normalized by the original distance (vertical line) on the uncompressed data. Notice that the optimal bounds can provide more than 20% tighter bounds. We also observe that the analytical solution using the Double Water-filling approach always provides better estimates than the numerical solution.

convex optimization.

First, we recall a couple of approaches from the literature which use the known coefficients and solely apply the Cauchy-Schwartz inequality for the unknown ones to update upper/lower bounds. One approach is to compress all sequences using the same set of coefficients and also store the compression error [13]. Another option is to store only the highest-energy coefficients as well as the compression error [22]. In both cases, if we denote the set of common stored coefficients for two sequences $\mathbf{X}, \mathbf{Q}$ by P_{xq}, the bounds are given by

$$\begin{aligned} LB &= \sqrt{\hat{D} - 2\sqrt{\bar{e_x}}\sqrt{\bar{e_q}}} \\ UB &= \sqrt{\hat{D} - 2\sqrt{\bar{e_x}}\sqrt{\bar{e_q}}} \\ \hat{D} &:= ||\mathbf{X}||_2^2 + ||\mathbf{X}||_2^2 - 2\Re\mathbf{X}(P_{xq})^*\mathbf{Q}(P_{xq}) \\ \bar{e_x} &:= ||\mathbf{X}(P \setminus P_{pq})||_2^2 \\ \bar{e_q} &:= ||\mathbf{X}(Q \setminus P_{pq})||_2^2 \ , \end{aligned}$$

where for $\bar{P} \subset P$, $\mathbf{X}(\bar{P})$ denotes the vector containing the entries of $\mathbf{X}$ in P. We refer to these two schemes as "First Coeffs+Error" and "Best Coeffs+Error" in what follows. We also refer to the numerical solution obtained by `cvx` as "Optimal - Numerical", whereas the approach presented in this paper is referred to as "Optimal - DoubleWaterfilling".

We test all four algorithms using data using the weblog traces in [11]. These are time-series that represent daily demand patterns (i.e., how many queries were posed per day) at a search engine. We consider time-series of length 1024. For a random subset, we execute pairwise distance computations and compute the true Euclidean distance on the uncompressed data as well as the lower/upper bounds on the distance for various compression ratios (i.e., number of retained coefficients). Note that for the approaches that record the best coefficients, the *position* of the recorded coefficients must be explicitly stored; but this is not necessary for approaches that record the first coefficients. So, for reasons of fair comparison, for all approaches *we allocate the same amount of space per compressed sequence*. In essence, the "First Coeffs + error" approach will eventually use a few more coefficients than the techniques using the best coefficients. For a more in depth discussion on these issue, the interested reader is directed to [18, 11].

Tightness of distance bounds: The results on the lower and upper bounds are shown in Fig. 8. We can observe the both the Numerical and the Water-filling solutions always provide better distance estimates than existing state-of-the-art solution. Using the Double Water-filling we can achieve an up to 22% tighter distance estimation. We also conducted experiments on other widely available periodic datasets (e.g. ECG, sunspot) and we observed similar results. Due to space restrictions these experiments are omitted.

Runtime: Even though both the Water-filling and the Numerical solutions significantly decrease the uncertainty with respect to the distance estimate, they are not equally efficient. We present the runtime for each approach in Fig. 9. The graph reports the average running time (in msec) for computing the distance estimates between one pair of sequences. It is evident that the proposed analytical solution based on Double Water-filling presents a very lightweight solution for distance estimation: it is up to **300 times faster** than the numerical approach. More importantly, the optimal solution through Water-filling is not computationally burdening: competing approaches require 1-2 msec for computation, whereas the Water-filling approach takes up to 6 msec. The small additional time is attributed to the fact that the algorithm distributes the currently re-

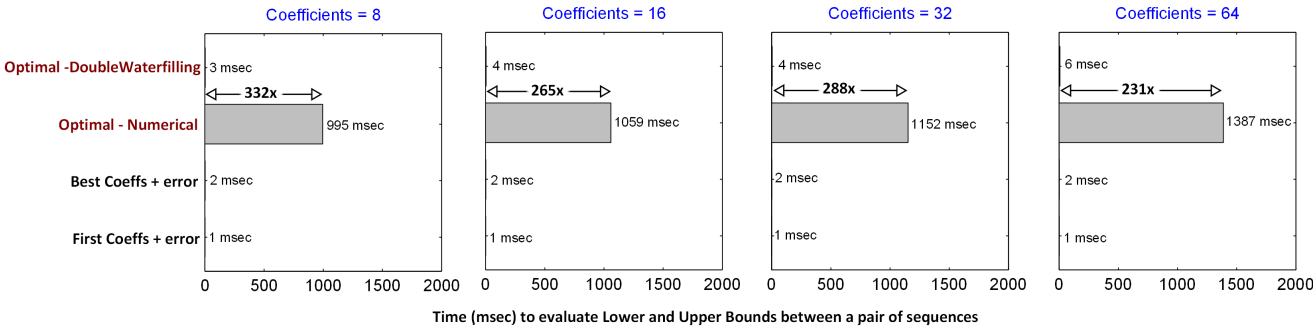

Figure 9: We depict the runtime for all approaches for computing the upper/lower bounds on the distance given a compressed representation with different number of coefficients. The optimal 'double-waterfilling' approach can be more than 300 times faster than the numerical approach.

maining energy over two-three iterations, thus incurring only minimal overhead. The numerical solution runs for more than 1 second and is considered impractical for large mining tasks.

10 Conclusions

In this work, we have presented an optimization approach for obtaining tight lower/upper bounds on the L_2 distance between objects that are compressed using a (potentially) different set of high-energy orthonormal coefficients. This problem has applications in a wide range of data management scenarios involving compression of sequential or high-dimensional data either for storage or transmission purposes with subsequent distance-based mining operations on the compressed domain. We have posed the problem as a convex optimization problem and have studied the properties of optimal solutions based on the KKT conditions. The proposed methodology is highly efficient, in that it requires only few iterations for termination and achieves significant speed-up over a numerical solution.

A wide gamut of applications currently challenged by storing and processing of large amounts of data can benefit from the outcome of this work. For example, web search behavior has been known to be periodic [23]; search engines aggregate and store such temporal patterns, e.g., Google Trends [24], to drive focused advertising campaigns. Other areas that deal with the storage and mining of massive periodic datasets can be found in: a) the profiling and latency estimation on large graphs of web/network hosts [3], b) large stream databases created for astronomical applications, such as the Low Frequency Array (LOFAR [25]) or the upcoming Square Kilometer Array telescope (SKA [26]).

References

[1] S. Chien and N. Immorlica, "Semantic similarity between search engine queries using temporal correlation," in *Proc. of WWW*, 2005.

[2] B. Lie, R. Jones, and K. Klinkner, "Measuring the Meaning in Time Series Clustering of Text Search Queries," in *Proc. of CIKM*, 2005.

[3] E. Nygren, R. K. Sitaraman, and J. Wein, "Networked systems research at Akamai," in *ACM Operating Systems Review (SIGOPS), 44(3)*, 2010.

[4] A. Souza and J.Pineda, "Tidal mixing modulation of sea surface temperature and diatom abundance in Southern California," in *Continental Shelf Research, 21(6-7)*, 2001, pp. 651–666.

[5] P. L. Noble and M. S. Wheatland, "Modeling the Sunspot Number Distribution with a Fokker-Planck Equation," *The Astrophysical Journal, 732(1)*, 2011.

[6] R. Agrawal, C. Faloutsos, and A. Swami, "Efficient Similarity Search in Sequence Databases," in *Proc. of FODO*, 1993.

[7] D. Rafiei and A. Mendelzon, "Efficient Retrieval of Similar Time Sequences Using DFT," in *Proc. of FODO*, 1998.

[8] K. Chan, A. W.-C. Fu, and C. T. Yu, "Haar Wavelets for Efficient Similarity Search of Time-Series: With and Without Time Warping," in *IEEE Trans. Knowl. Data Eng. 15(3)*, 2003, pp. 686–705.

[9] V. Eruhimov, V. Martyanov, P. Raulefs, and E. Tuv, "Combining unsupervised and supervised approaches to feature selection for multivariate signal compression," in *Intelligent Data Engineering and Automated Learning*, 2006, pp. 480–487.

[10] Y. Cai and R. Ng, "Indexing spatio-temporal trajectories with chebyshev polynomials." in *Proc. of ACM SIGMOD*, 2004.

[11] M. Vlachos, S. Kozat, and P. Yu, "Optimal Distance Bounds on Time-Series Data," in *Proc. of SDM*, 2009, pp. 109–120.

[12] P. Y. M. Vlachos, S.S. Kozat, "Optimal distance

bounds for fast search on compressed time-series query logs ," in *ACM Transactions on the Web, 4(2)*, 2010.

[13] C. Wang and X. S. Wang, "Multilevel filtering for high dimensional nearest neighbor search," in *ACM SIGMOD Workshop on Data Mining and Knowledge Discovery*, 2000.

[14] M. Vlachos, P. Yu, and V. Castelli, "On Periodicity Detection and Structural Periodic Similarity," in *Proc. of SDM*, 2005.

[15] A. Mueen, S. Nath, and J. Lie, "Fast Approximate Correlation for Massive Time-Series Data," in *Proc. of SIGMOD*, 2010.

[16] E. Keogh and S. Kasetty, "On the need for time series data mining benchmarks: A survey and empirical demonstration," in *Proc. of KDD*, 2002.

[17] A. Mueen, E. J. Keogh, and N. B. Shamlo, "Finding time series motifs in disk-resident data," in *Proc. of ICDM*, 2009, pp. 367–376.

[18] E. Keogh, K. Chakrabarti, S. Mehrotra, and M. Pazzani, "Locally adaptive dimensionality reduction for indexing large time series databases," in *Proc. of ACM SIGMOD*, 2001, pp. 151–162.

[19] S. Boyd and L. Vandenberghe, *Convex Optimization*, 1st ed. Cambridge University Press, 2004.

[20] T. Basar and G. J. Olsder, *Dynamic Noncooperative Game Theory*, 2nd ed. Academic Press, 1995.

[21] "CVX: Matlab software for disciplined convex programming. ver. 1.21," http://www.stanford.edu/~boyd/cvx/, 2011.

[22] M. Vlachos, C. Meek, Z. Vagena, and D. Gunopulos, "Identification of Similarities, Periodicities & Bursts for Online Search Queries," in *Proc. of SIGMOD*, 2004.

[23] A. Alkilany, "Visualise web usage mining: Spanning sequences' impact on periodicity discovery," in *Proc. of Intl. Conf. on Information Visualisation*, 2010, pp. 301–309.

[24] H. Choi and H. Varian, "Predicting the Present with Google Trends," *Google Technical Report*, 2009.

[25] "LOFAR: Low Frequency Array for radio astronomy," http://www.lofar.org/.

[26] "SKA: Square Kilometer Array Telescope," http://www.skatelescope.org.

APPENDIX: PROOF OF THEOREM 7.1

For the first part, note that the problem (6.3) is a double minimization problem over $\{z_i\}_{i\in p_x^-}$ and $\{y_i\}_{i\in p_q^-}$. If we fix one vector in the objective function of (6.3), then the optimal solution with respect to the other one is given by the Water-filling algorithm. In fact, if we consider the KKT conditions (6.5) or the KKT conditions to (6.2), they correspond exactly to (7.9). The Water-filling algorithm has the property that if $\mathbf{a} = \text{waterfill}\ (\mathbf{b}, e_x, A)$, then $b_i > 0$ implies $a_i > 0$. Furthermore, it has a monotonicity property in the sense that $b_i \le b_j$ implies $a_i \le a_j$. Assume that, at optimality, $a_{l_1} < a_{l_2}$ for some $l_1 \in P_1, l_2 \in P_3$; because $b_{l_1} \ge B \ge b_{l_3}$ we can swap these two values to decrease the objective function, which is a contradiction. The exact same argument applies for $\{b_l\}$, so $\min_{l\in P_1} a_l \ge \max_{l\in P_3} a_l, \min_{l\in P_2} b_l \ge \max_{l\in P_3} b_l$.

For the second part, note that $-\sum_{i\in P_3}\sqrt{z_i}\sqrt{y_i} \ge -\sqrt{e'_x}\sqrt{e'_q}$. If $e'_x e'_q > 0$, then at optimality this is attained with equality for the particular choice of $\{a_l, b_l\}_{l\in P_3}$. It follows that all entries of the optimal solution $\{a_l, b_l\}_{l\in p_x^- \cup p_q^-}$ are strictly positive, hence (6.5d) implies that

$$\text{(A-1a)} \quad a_i = \frac{b_i}{\lambda + \alpha_i}, \quad i \in P_1$$

$$\text{(A-1b)} \quad b_i = \frac{a_i}{\mu + \beta_i}, \quad i \in P_2$$

$$\text{(A-1c)} \quad \begin{aligned} a_i &= (\mu + \beta_i) b_i, \quad i \in P_3 \\ b_i &= (\lambda + \alpha_i) a_i, \quad i \in P_3 \end{aligned}$$

For the particular solution with all entries in P_3 equal $\left(a_l = \sqrt{e'_x/|P_3|}, b_l = \sqrt{e'_q/|P_3|}\right)$, (7.10a) is an immediate application of A-1.c. The optimal entries $\{a_l\}_{l\in P_1}, \{b_l\}_{l\in P_2}$ are provided by Water-filling with available energies $e_x - e'_x, e_q - e'_q$, respectively, so (7.11) immediately follow.

For the third part, note that the cases that either $e'_x = 0, e'_q > 0$ or $e'_x > 0, e'_q = 0$ are excluded at optimality by the first part, cf. (7.9).

For the last part, note that when $e'_x = e'_q = 0$, equivalently $a_l = b_l = 0$ for $l \in P_3$, it is not possible to take derivatives with respect to any coefficient in P_3, so the last two equations of (6.5) do not hold. In that case, we need to perform a standard perturbation analysis. Let $\boldsymbol{\epsilon} := \{\epsilon_l\}_{l\in P_1 \cup P_2}$ be a sufficiently small positive vector. As the constraint set of (6.3) is linear in z_i, y_i, any *feasible* direction (of potential decrease of the objective function) is of the form $z_i \leftarrow z_i - \epsilon_i, i \in P_1$, $y_i \leftarrow y_i - \epsilon_i, i \in P_2$, and $z_i, y_i \ge 0, i \in P_3$ such that $\sum_{i\in P_3} z_i = \sum_{i\in P_1}\epsilon_i, \sum_{i\in P_3} y_i = \sum_{i\in P_2}\epsilon_i$. The change in the objective function is then equal to (modulo an $o(||\boldsymbol{\epsilon}||^2)$ term)

(A-2)

$$\begin{aligned} g(\boldsymbol{\epsilon}) &\approx \frac{1}{2}\sum_{i\in P_1}\frac{b_i}{\sqrt{z_i}}\epsilon_i + \frac{1}{2}\sum_{i\in P_2}\frac{a_i}{\sqrt{y_i}}\epsilon_i - \sum_{i\in P_3}\sqrt{z_i}\sqrt{y_i} \\ &\ge \frac{1}{2}\sum_{i\in P_1}\frac{b_i}{\sqrt{z_i}}\epsilon_i + \frac{1}{2}\sum_{i\in P_2}\frac{a_i}{\sqrt{y_i}}\epsilon_i - \sqrt{\sum_{i\in P_1}\epsilon_i}\sqrt{\sum_{i\in P_2}\epsilon_i} \\ &\ge \frac{1}{2}\min_{i\in P_1}\frac{b_i}{\sqrt{z_i}}\epsilon_1 + \frac{1}{2}\min_{i\in P_2}\frac{a_i}{\sqrt{y_i}}\epsilon_2 - \sqrt{\epsilon_1}\epsilon_2 \end{aligned}$$

where the first inequality follows from an application of Cauchy-Schwartz inequality to the last term, and in the second one we have defined $\epsilon_j = \sum_{i\in P_j}\epsilon_i, i = 1, 2$. Let us define $\epsilon := \sqrt{\epsilon_1/\epsilon_2}$. From the last expression, it suffices to test for any $i \in P_1, j \in P_2$:

(A-3)

$$\begin{aligned} g(\epsilon_1, \epsilon_2) &= \frac{1}{2}\frac{b_i}{\sqrt{z_i}}\epsilon_1 + \frac{1}{2}\frac{a_j}{\sqrt{y_j}}\epsilon_2 - \sqrt{\epsilon_1}\sqrt{\epsilon_2} = \frac{1}{2}\sqrt{\epsilon_1}\sqrt{\epsilon_2}\, g_1(\epsilon) \\ g_1(\epsilon) &:= \frac{b_i}{\sqrt{z_i}}\epsilon + \frac{a_j}{\sqrt{y_j}}\frac{1}{\epsilon} - 2 \ge \frac{1}{\epsilon}g_2(\epsilon) \\ g_2(\epsilon) &:= \frac{b_i}{A}\epsilon^2 - 2\epsilon + \frac{a_i}{B} \end{aligned}$$

where the inequality above follows from the fact that $\sqrt{z_i} \le A, i \in P_1$ and $\sqrt{y_i} \le B, i \in P_2$. Note that $h(\epsilon)$ is a quadratic with a non-positive discriminant $\Delta := 4(1 - \frac{a_i b_i}{AB}) \le 0$ since, by definition, we have that $B \le b_i, i \in P_1$ and $A \le a_i, i \in P_2$. Therefore $g(\epsilon_1, \epsilon_2) \ge 0$ for any (ϵ_1, ϵ_2) both positive and sufficiently small, which is a necessary condition for local optimality. By convexity, the obtained vector pair $(\mathbf{a}, \mathbf{b})$ constitutes an optimal solution. ■