On the Optimality of the Greedy Heuristic in Wavelet Synopses for Range Queries

Yossi Matias

School of Computer Science
Tel Aviv University
Tel Aviv 69978, Israel
matias@tau.ac.il

Daniel Urieli

School of Computer Science
Tel Aviv University
Tel Aviv 69978, Israel
daniel1@post.tau.ac.il

(July 2004; rev. July 2005)

Abstract

In recent years wavelet based synopses were shown to be effective for approximate queries in database systems. The simplest wavelet synopses are constructed by computing the Haar transform over a vector consisting of either the raw-data or the prefix-sums of the data, and using a greedy-heuristic to select the wavelet coefficients that are kept in the synopsis. The greedy-heuristic is known to be optimal for point queries w.r.t. the mean-squared-error, but no similar optimality result was known for range-sum queries, for which the effectiveness of such synopses was only shown experimentally.

The optimality of the greedy-heuristic for the case of point queries is due to the Haar basis being orthonormal for this case, which allows using the Parseval-based thresholding. Thus, the main technical question we are concerned with in this paper is whether the Haar basis is orthonormal for the case of range-sum queries. We show that it is *not* orthogonal for the case of range-sum queries over the raw data, and that it *is* orthonormal for the case of prefix-sums. Consequently, we show that a slight variation of the greedy-heuristic over the prefix-sums of the data is an optimal thresholding w.r.t. the mean-squared-error. As a result, we obtain the first linear time construction of a provably optimal wavelet synopsis for range-sum queries. The crux of our proof is based on a novel construction of inner products, that define the error measured over *range-sum queries*.

1 Introduction

In recent years there has been increasing attention to the development and study of data synopses, as effective means for addressing performance issues in massive data sets. Data synopses are concise representations of data sets, that are meant to effectively support approximate queries to the represented data sets [5]. A primary constraint of a data synopsis is its size. The effectiveness of a data synopsis is measured by the accuracy of the answers it provides, as well as by its response time and its construction time. Several different synopses were introduced and studied, including random samples, sketches, and different types of histograms. Recently, wavelet-based synopses were introduced and shown to be a powerful tool for building effective data synopses for various applications, including selectivity estimation for query optimization in DBMS, approximate query processing in OLAP applications and more (see [12, 18, 16, 17, 1, 2, 4, 3], and references therein).

The general idea of wavelet-based approximations is to transform a given data vector of size N into a representation with respect to a wavelet basis (this is called a *wavelet transform*), and approximate it using only $M \ll N$ wavelet basis vectors, by retaining only M coefficients from the linear combination that spans the data vector (*coefficients thresholding*). The linear combination that uses only M coefficients (and assumes that all other coefficients are zero) defines a new vector that approximates the original vector, using less space. This is called *M-term approximation*, which defines a *wavelet synopsis* of size M.

Wavelet synopses. Wavelets were traditionally used to compress some data set where the purpose was to reconstruct, in a later time, an approximation of the *whole* data using the set of retained coefficients. The situation is a little different when using wavelets for building synopses in database systems [12, 18]: in this case different *portions* of the data are reconstructed each time, in response to user queries, and same portions of the data can be built several times in response to different queries. Thus, when building wavelet synopses in database systems, the approximation error is measured over *queries*, in contrast to the standard wavelet-based approximation techniques, where the error is measured over the data. Another aspect of the use of wavelet-based synopses is that due to the large data-sizes in modern DBMS (giga-, tera- and peta-bytes), the efficiency of building wavelet synopses is of primary importance. Disk I/Os should be minimized and non-linear-time algorithms may be unacceptable.

Wavelet synopses suggested in the database literature typically used the Haar wavelet basis due to its simplicity.

Optimal wavelet synopses. The main advantage of transforming the data into a representation with respect to a wavelet basis is that for data vectors containing similar values, many wavelet coefficients tend to have very small values. Thus, eliminating such small coefficients introduces only small errors when reconstructing the original data, resulting in a very effective form of lossy data compression.

After the wavelet transform is done, the selection of coefficients that are retained in the wavelet synopsis may have significant impact on the approximation error. The goal is therefore to select a subset of M coefficients that minimizes the approximation error. A subset that minimizes the approximation error for a given error metric w.r.t. the given basis is called an *optimal wavelet synopsis*.

While there has been a considerable work on wavelet synopses and their applications [12, 18, 16, 17, 1, 2, 4, 3], so far most known optimal wavelet synopses are with respect to *point queries*. The first one [12, 18] is based on a linear-time Parseval-based (PB) algorithm, which was typically used in traditional wavelet approximations (e.g [7]). The synopsis minimizes the mean-squared error (MSE) over all possible *point queries*, relying on the fact that the Haar basis is orthonormal.

The second synopsis, introduced recently [4], is constructed using a dynamic-programming based $O(N^2 M \log M)$ algorithm, and it minimizes the max relative or absolute error over all possible *point queries*. Another optimality result is an $O\left(N^2 M\left(D+M\right)\right)$ time dynamic-programming algorithm that obtains an optimal wavelet synopsis over multiple measures [2]. The synopsis is optimal w.r.t. an error metric defined as weighted combination of L_2 norms over the multiple measures, where each L_2 norm is measured over *point queries*.

Two recent optimality results deal with workload-based wavelet synopses, which are wavelet synopses that minimize an error measured based on the query workload. The first optimal workload-based wavelet synopsis [11] minimizes the workload-based-mean-squared absolute or relative error, measured over all possible point queries, using weighted Haar wavelets, in linear-time and optimal number of I/Os. The second optimal workload-based wavelet synopsis [13] minimizes the workload-based-mean-squared absolute error, measured over all possible point queries, using Haar wavelets, in $O(N^2 M / \log M)$ time.

Wavelet synopses for range-sum queries. A primary use of wavelet-based synopses in DBMS is answering range-sum queries. For such synopses, the approximation error is measured over the set of all possible range queries.

In the database literature (e.g., [12, 18, 16, 17]), two types of wavelet synopses for range-sum queries were presented. One over raw data and the other one over the vector of prefix-sums of the data. In both cases, a range-sum query can be expressed using point queries. In the prefix-sums case, the answer to a range query is a difference between two point queries; in the raw-data case the answer is a sum of all point queries in the range, or using a formula that depends on about $2\log N$ queries for pre-specified hierarchical ranges. Thus, suggested thresholding algorithms were based on optimization w.r.t. point queries.

The basic thresholding algorithms suggested in [12, 18] for range-sum queries are based on the greedy-heuristic, in which the coefficients are normalized based on their levels, and the highest normalized coefficients are retained in the synopsis. For the case of point queries, the greedy-heuristic is optimal as it is equivalent to the Parseval-based algorithm. For range-queries, however, no efficient optimality result has been known, yet the greedy-heuristic was selected for a lack of a better choice, and due to the simplicity and efficiency of its implementation.

It seems that optimality over points would give especially good results for the case of prefix-sums, where the answer to a range-query is a difference between only two point queries. Moreover, note that if we are interested only in range queries of the form $d_{0:i}$, that is, $\sum_{i=0}^{i} d_i$, then the greedy-heuristic is optimal, as a point query in this case is exactly a range query of the form $d_{0:i}$. However, it turns out that when using the greedy heuristic over prefix-sums for general queries $d_{i:j}$, the mean-squared-error could be larger than the optimal error by a factor of $\Theta\left(\sqrt{N}\right)$, as shown in this paper (Thm. 5). Nevertheless, we show here that a slight variation of the greedy heuristic is indeed an optimal thresholding for the case of prefix-sums.

We note that both synopses (prefix-sums-based and raw-data-based) were tested and experimental results showed that for range-sum queries, the approximations were better in the prefix-sums case than in the raw-data case [12, 18]. This gives a motivation for finding efficient (linear) optimal thresholding for the case of prefix-sums.

The only optimality result for range-sum queries that we are aware of is one mentioned in [6]; the authors mention that there exists an algorithm that computes optimal M-term wavelet synopses for range-sum queries over prefix-sums in $O\left(N\left(M\log N\right)^{O(1)}\right)$ time, but no further details about this synopsis were available.

1.1 Contributions

As pointed out above, the greedy heuristic is based on applying the Parseval-based thresholding, which is optimal for point queries, for the case of range-sum queries. The reason we can rely on Parseval's formula and get an optimal thresholding in the case of point queries, is because in this case the Haar basis is orthonormal. In fact, any *orthogonal* basis can be normalized, and thus in order to use Parseval's formula it suffices to show orthogonality. Thus, the main technical question we are concerned with in this paper is whether the Haar basis is orthogonal for the case of range-sum queries.

We show that the Haar basis *is not* orthogonal for the case of range-sum queries over the raw data, and we show that it *is* orthogonal for the case of range-sum queries over the prefix-sums of the data. Consequently, we show that a slight variation of the greedy-heuristic over the prefix-sums of the data is an optimal thresholding w.r.t. the mean-squared-error, obtaining the first linear time construction of a provably optimal wavelet synopsis for range-sum queries. As we show that the Haar basis is non-orthogonal in the case of raw data, Parseval's formula cannot be applied in this case for optimal thresholding.

A natural technical question is what is the notion of orthonormality for range-queries, and with respect to which inner-product. We base our result on a novel construction of inner-products related to the error measured over *range-sum* queries. Specifically, the main idea is to express the mean-squared-error measured over *range-sum* queries using an inner product, both for the raw-data case and the prefix-sums case. So far inner products were used in a more conventional way, to define an Euclidean error between two vectors, or a generalized Euclidean error (weigthed norm, see [11]). The main technical contributions with respect to this approach are:

- We define the case of range-sum queries in terms of an inner product space, and construct inner products for the cases the wavelet transform is done either over the prefix-sums or over the raw data.
- We show that the Haar basis is orthogonal with respect to the new inner product defined for the case of prefix-sums. This enables using Parserval-based thresholding.
- In contrast, we show that the Haar basis is not orthogonal with respect to the inner product defined for the case of range-sum queries over *raw data*, both analytically and empirically. For non-orthogonal bases no efficient optimal thresholding algorithm is known. Additionally, our empirical proof demonstrates an anomaly when using a non-orthogonal basis, where a larger synopsis may result with an increased error.

As a result of the above, we present an optimal wavelet-synopsis for the vector of prefix-sums of the data. Specifically:

- We show a wavelet synopsis that minimizes the MSE error measure, over all possible $O(N^2)$ range-sum queries (N is the size of the approximated vector).
- The synopsis is built by an $O(N)$ time algorithm. The algorithm is also I/O optimal with $O\left(N/B\right)$ I/Os for disk block of size B.
- The synopsis is also an optimal *enhanced wavelet synopsis.* Enhanced wavelet synopses are synopses that allow changing the values of their coefficients to *arbitrary values.*

1.2 Paper outline

In Sec. 2 we describe the basics of wavelet-based synopses. In Sec. 3 we describe some basics regarding Parseval Formula and its use. In Sec. 4 we describe the development of the optimal synopsis for prefix-sums. We build the inner-product for the case of prefix-sums, and then construct the optimal synopsis resulted from it. We then discuss the similarity and difference between our optimal wavelet synopsis and the greedy-heuristic given in [12, 18]. In Sec. 5 we show the non-orthogonality of Haar basis for range queries in the raw-data case. In Sec. 6 we present experimental results. Conclusions are given in Sec. 7.

2 Wavelets basics

In this section we start by presenting the Haar wavelets (Sec. 2.1). We continue with presenting wavelet based synopses, obtained by thresholding process (Sec. 2.2). The error tree structure is presented next (Sec. 2.3), along with a description of the reconstruction of the original data from the wavelet synopses (Sec. 2.4).

Wavelets are a mathematical tool for the hierarchical decomposition of functions in a space-efficient manner. Wavelets represent a function in terms of a coarse overall shape, plus details that range from coarse to fine. Regardless of whether the function of interest is an image, a curve, or a surface, wavelets offer an elegant technique for representing the various levels of detail of the function in a space-efficient manner.

2.1 One-dimensional Haar wavelets

Haar wavelets are conceptually the simplest wavelet basis functions, and were thus used in previous works of wavelet synopses. They are fastest to compute and easiest to implement. We focus on them for purpose of exposition in this paper. To illustrate how Haar wavelets work, we will start with a simple example borrowed from [12, 18].

Suppose we have one-dimensional "signal" of $N = 8$ data items: $S = [2, 2, 0, 2, 3, 5, 4, 4]$. We will show how the Haar wavelet transform is done over S. We first average the signal values, pairwise, to get a new lower-resolution signal with values $[2, 1, 4, 4]$. That is, the first two values in the original signal (2 and 2) average to 2, and the second two values 0 and 2 average to 1, and so on. We also store the pairwise differences of the original values (divided by 2) as detail coefficients. In the above example, the four detail coefficients are $(2-2)/2 = 0$, $(0-2)/2 = -1$, $(3-5)/2 = -1$, and $(4-4)/2 = 0$. It is easy to see that the original values can be recovered from the averages and differences.

This was one phase of the Haar wavelet transform. By repeating this process recursively on the averages, we get the Haar wavelet transform (Table 1). We define the *wavelet transform* (also called *wavelet decomposition*) of the original eighth-value signal to be the single coefficient representing the overall average of the original signal, followed by the detail coefficients in the order of increasing resolution. Thus, for the one-dimensional Haar basis, the wavelet transform of our signal is given by $\tilde{S} = [2\frac{3}{4}, -1\frac{1}{4}, \frac{1}{2}, 0, 0, -1, -1, 0]$

The individual entries are called the wavelet coefficients. The wavelet decomposition is very efficient computationally, requiring only *O(N)* CPU time and *O(N/B)* I/Os to compute for a signal of N values, where B is the disk-block size.

No information has been gained or lost by this process. The original signal has eight values, and so does the transform. Given the transform, we can reconstruct the exact signal by recursively adding

Resolution	Averages	Detail Coefficients
8	[2, 2, 0, 2, 3, 5, 4, 4]	
4	[2, 1, 4, 4]	[0,-1,-1, 0]
2	[1.5, 4]	[0.5, 0]
1	[2.75]	-1.25

Table 1: Haar Wavelet Decomposition

and subtracting the detail coefficients from the next-lower resolution. In fact we have transformed the signal S into a representation with respect to another basis of R^8, the Haar wavelet basis. A detailed discussion can be found, for instance, in [15].

2.2 Thresholding

Given a limited amount of storage for maintaining a wavelet synopsis of a data array A (or equivalently a vector S), we can only retain a certain number $M \ll N$ of the coefficients stored in the wavelet decomposition of A. The remaining coefficients are implicitly set to 0. The goal of coefficient thresholding is to determine the best subset of M coefficients to retain, so that some overall error measure in the approximation is minimized.

One advantage of the wavelet transform is that in many cases a large number of the detail coefficients turn out to be very small in magnitude. Truncating these small coefficients from the representation (i.e., replacing each one by 0) introduces only small errors in the reconstructed signal. We can approximate the original signal effectively by keeping only the most significant coefficients.

For a given input sequence $d_0, ..., d_{N-1}$, we can measure the error of approximation in several ways. We first discuss the error approximation for data values, and defer the definition of error approximation for range queries to Sec. 4.1. Let the i'th data value be d_i. Let q_i be the i'th point query, which it's value is d_i. Let $\hat{d_i}$ be the estimated result of d_i. We use the following error measure for the absolute error over the i'th data value:

$$e_i = e(q_i) = |d_i - \hat{d_i}|$$

Once we have the error measure for representing the errors of individual data values, we would like to measure the norm of the vector of errors $e = (e_0, ..., e_{N-1})$. The standard way is to use the L_2 norm of e divided by $\sqrt{N}$ which is called the *mean squared error*:

$$MSE(e) = \|e\| = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1} e_i^2}$$

We would use the terms MSE and L_2 norm interchangeably during our development since they are completely equivalent, to a positive multiplicative constant.

The basic thresholding algorithm, based on Parseval's formula, is as follows: let $\alpha_0, ..., \alpha_{N-1}$ be the wavelet coefficients, and for each α_i let $level(\alpha_i)$ be the resolution level of α_i. The detail coefficients are normalized by dividing each coefficient by $\sqrt{2^{level(a_i)}}$ reflecting the fact that coefficients at the lower resolutions are "less important" than the coefficients at the higher resolutions. This process actually turns the wavelet coefficients into an orthonormal basis coefficients (and is thus called "normalization"). The M largest normalized coefficients are retained. The remaining $N - M$ coefficients are implicitly replaced by zero. This deterministic process *provably* minimizes the L_2 norm of the vector of errors defined above, based on Parseval's formula (see Sec. 3).

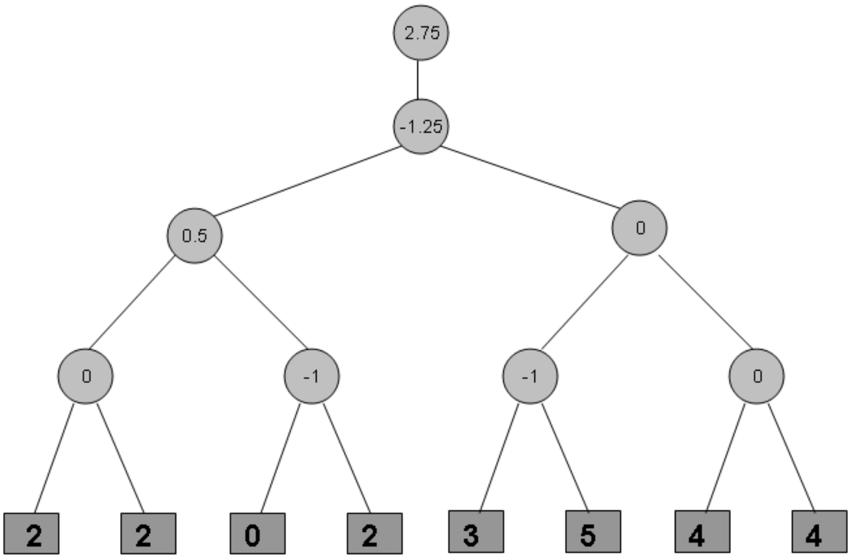


Figure 1: Error tree for $N = 8$

2.3 Error tree

The wavelet decomposition procedure followed by any thresholding can be represented by an *error tree* [12, 18].

Fig. 1 presents the error tree for the above example. Each internal node of the error tree is associated with a wavelet coefficient, and each leaf is associated with an original signal value. Internal nodes and leaves are labeled separately by $0, 1, ..., N-1$. For example, the root is an internal node with label 0 and its node value is 2.75 in Fig. 1. For convenience, we shall use "node" and "node value" interchangeably. The construction of the error tree exactly mirrors the wavelet transform procedure. It is a bottom-up process. First, leaves are assigned original signal values from left to right. Then wavelet coefficients are computed, level by level, and assigned to internal nodes. A coefficient is said to be at *resolution level* i, if its depth in the tree is i.

2.4 Reconstruction of original data

Given an error tree T and an internal node t of T, $t \neq a_0$, we let $leftleaves(t)$ $(rightleaves(t))$ denote the set of leaves (i.e., data) nodes in the subtree rooted at t's left (resp., right) child. Also, given any (internal or leaf) node u, we let $path(u)$ be the set of all (internal) nodes in T that are proper ancestors of u (i.e., the nodes on the path from u to the root of T, including the root but not u) with nonzero coefficients.

Finally, for any two leaf nodes d_l and d_k we denote $d(l : h)$ as the range sum $\sum_{i=l}^{k} d_i$

Using the error tree representation T, we can outline the following reconstruction properties of the Haar wavelet decomposition [12, 18].

2.4.1 Single value

The reconstruction of any data value d_i depends only on the values of the nodes in $path(d_i)$.

$$d_i = \sum_{\alpha_j \in path(d_i)} \delta_{ij} \cdot \alpha_j$$

where $\delta_{ij} = +1$ if $d_i \in leftleaves(\alpha_j)$ or $j = 0$, and $\delta_{ij} = -1$ otherwise. Thus, a reconstruction of a single data values involves the summation of at most $\log N + 1$ coefficients.

2.4.2 Range sum

When the transform is done over the prefix-sums, an answer to a range query is a difference between two point queries, and thus two points should be reconstructed, using the method described above.

When the transform is done over the raw data, an internal node α_j contributes to the range sum $d(l : h)$ only if $\alpha_j \in path(d_l) \cup path(d_k)$.

$$d(l : h) = \sum_{\alpha_j \in path(d_l) \cup path(d_h)} x_j$$

where

$$x_j = \begin{cases} (h - l) \cdot \alpha_j & \text{if } j = 0 \\ (|leftleaves(\alpha_j, l : h)| - |rightleaves(\alpha_j, l : h)|) \cdot \alpha_j & \text{otherwise} \end{cases}$$

and where $leftleaves(\alpha_j, l : h) = leftleaves(\alpha_j) \cap \{d_l, d_{l+1}, ..., d_h\}$ (i.e., the intersection of $leftleaves(\alpha_j)$ with the summation range) and $rightleaves(\alpha_j, l : h)$ is defined similarly.
Thus, reconstructing a range sum involves the summation of at most $2 \log N + 1$ coefficients, regardless of the width of the range.

3 Optimal thresholding in orthonormal bases

The efficient construction of optimal wavelet-synopses is commonly based on Parseval's formula.

3.1 Parseval's formula

Let V be a vector space, where $v \in V$ is a vector and $\{u_0, ..., u_{N-1}\}$ is an orthonormal basis of V. We can express v as $v = \sum_{i=0}^{N-1} \alpha_i u_i$. Then

$$\|v\|^2 = \sum_{i=0}^{N-1} \alpha_i^2 \tag{1}$$

An M-term approximation is achieved by representing v using a subset of coefficients $S \subset \{\alpha_0, ..., \alpha_{N-1}\}$ where $|S| = M$. The error vector is then $e = \sum_{i \notin S} \alpha_i u_i$. By Parseval's formula, $\|e\|^2 = \sum_{i \notin S} \alpha_i^2$. This proves the following theorem.

Theorem 1 (Parseval-based optimal thresholding) *Let V be a vector space, where $v \in V$ is a vector and $\{u_0, ..., u_{N-1}\}$ is an orthonormal basis of V. We can represent v by $\{\alpha_0, ..., \alpha_{N-1}\}$ where $v = \sum_{i=0}^{N-1} \alpha_i u_i$. Suppose we want to approximate v using a subset $S \subset \{\alpha_0, ..., \alpha_{N-1}\}$ where $|S| = M \ll N$. Picking the M largest (in absolute value) coefficients to S minimizes the L_2 norm of the error vector, over all possible subsets of M coefficients.*

Given an inner-product, based on this theorem one can easily find an optimal synopsis by choosing the largest M coefficients.

In fact, we can use the Parseval-based optimal thresholding even when using an *orthogonal* basis. Given a linear combination of a vector v with respect to an orthogonal basis $\tilde{U}$, $v = \sum \tilde{\alpha_i}\tilde{u_i}$, we can simply represent v with respect to an orthonormal basis U, $v = \sum \alpha_i u_i$, where $\alpha_i = \tilde{\alpha_i} \cdot \|\tilde{u_i}\|$ and $u_i = \frac{\hat{u_i}}{\|u_i\|}$. Parseval-based thresholding can then be applied on the normalized coefficients α_i. In particular, for Haar wavelet synopses we multiply a coefficient at level i by the norm of its corresponding basis vector, that is, by $\frac{1}{\sqrt{2^i}}$. Thus, the main technical question with which we are concerned in the rest of the paper is the *orthogonality* of the Haar basis in specific cases of interest.

Note that in order to show a negative result, that is, that Thm. 1 *cannot* be applied during the thresholding w.r.t. a given basis, it is sufficient to find an inner product that defines the desired L_2 norm, and show that the given basis is not orthogonal w.r.t. this inner product. This relies on the fact that if a norm is defined by some inner-product, then this inner-product is unique; that is, no other inner-product (w.r.t. which the basis *is* orthogonal) defines the same norm, as can easily be shown:

Lemma 1 *Let $\langle v, u\rangle$ be an inner product that defines an L_2 norm by $\sqrt{\langle v, v\rangle}$. There is no other inner product that defines the same norm.*

Proof: Suppose that there exists another inner product, $(\cdot, \cdot)$, with the same norm as that of $\langle \cdot, \cdot \rangle$; that is, for every vector x, $\sqrt{\langle x, x\rangle} = \sqrt{(x, x)}$ and consequently $\langle x, x\rangle = (x, x)$. Then, since by definition, for any two vectors u and w,

$$\langle u + w, u + w\rangle = \langle u, u\rangle + 2\langle u, w\rangle + \langle w, w\rangle$$

and

$$(u + w, u + w) = (u, u) + 2\,(u, w) + (w, w)$$

we obtain that $\langle u, w\rangle = (u, w)$, implying that the inner products $\langle \cdot, \cdot \rangle$ and $(\cdot, \cdot)$ are identical. □

Thus, if a basis is shown to be non-orthogonal w.r.t. an inner product $\langle \cdot, \cdot \rangle$ whose norm is $\| \cdot \| = \sqrt{\langle \cdot, \cdot \rangle}$, then it can be said to be *non-orthogonal w.r.t. the norm* $\| \cdot \|$.

3.2 Optimality over enhanced wavelet synopses

Note that in the definition of wavelet synopses we limited ourselves to picking subsets of coefficients with original values from the linear combination that spans v (as is usually done). In case $\{u_0, ..., u_{N-1}\}$ is a wavelet basis, these are the coefficients that results from the wavelet transform. We next show that the optimal thresholding according to Thm. 1 is optimal even according to an enhanced definition of M-term approximation. We define *enhanced wavelet synopses* as wavelet synopses that allow *arbitrary values* to the retained wavelet coefficients, rather than the original values that resulted from the transform. The set of possible standard synopses is a subset of the set of possible *enhanced* synopses, and therefore an optimal synopsis according to the standard definition is not necessarily optimal according to the enhanced definition. An enhanced wavelet synopsis is, for instance, the synopsis described in [3], where probabilistic techniques are used in order to determine a coefficient's value, so as to minimize the (expected) max-error. The following is a well known theorem about orthonormal transformations and enhanced synopses; we provide its proof for completeness.

Theorem 2 *When using an orthonormal basis, choosing the largest M coefficients with original values is an optimal enhanced wavelet synopses.*

Proof: The proof is based on the fact that the basis is orthonormal. It is enough to show that given some synopsis of M coefficients with original values, any change to the values of some subset of coefficients in the synopsis would only make the approximation error larger:
Let $u_1, ..., u_N$ be an orthonormal basis and let $v = \alpha_1 u_1 + ... + \alpha_N u_N$ be the vector we would like to approximate by keeping only M wavelet coefficients. Without loss of generality, suppose we choose the first M coefficients and have the following approximation for v: $\tilde{v} = \sum_{i=1}^{M} \alpha_i u_i$. According to Parseval's formula $\|e\|^2 = \sum_{i=M+1}^{N} \alpha_i^2$ since the basis is orthonormal. Now suppose we would change the values of some subset of j retained coefficients to new values. Let us see that due to the orthonormality of the basis it would only make the error larger. Without loss of generality we change the first j coefficients, that is, we change $\alpha_1, ..., \alpha_j$ to be $\alpha'_1, ..., \alpha'_j$. In this case the approximation would be $\tilde{v}' = \sum_{i=1}^{j} \alpha'_i u_i + \sum_{i=j+1}^{M} \alpha_i u_i$. The approximation error would be $v - \tilde{v}' = \sum_{i=1}^{j} (\alpha_i - \alpha'_i) u_i + \sum_{i=M+1}^{N} \alpha_i u_i$. It is easy to see that the error of approximation would be: $\|e\|^2 = \langle v - \tilde{v}', v - \tilde{v}' \rangle = \sum_{i=1}^{j} (\alpha_i - \alpha'_i)^2 + \sum_{i=M+1}^{N} \alpha_i^2 > \sum_{i=M+1}^{N} \alpha_i^2$. □

4 The synopsis construction

In this section we describe the development of our optimal synopsis. First we define the MSE error metrics by which we measure the approximation error over range-sum queries (Sec. 4.1), denoted here as MSE_{range}. Our goal is to efficiently build a synopsis that minimizes MSE_{range}.

Recall that Parseval-based thresholding is an efficient method to build optimal synopses with respect to an L_2 norm (of the error vector) in an inner-product vector-space. If we can show that the MSE_{range} is an L_2 norm (of the error vector) defined by some inner-product, and then find an orthonormal wavelet basis (with respect to this inner-product), we could use Parseval-based thresholding to build an optimal synopsis with respect to this basis. This is exactly what our construction does.

Our main idea is to define our problem in terms of an inner product space by constructing a range-sum-based inner product (Sec. 4.2), and to show that the L_2 norm defined by the new inner product is equivalent, up to a constant positive factor, to the MSE_{range} error measure when approximating a prefix-sums vector (Sec. 4.3). We then show that the Haar basis is orthogonal with respect to this inner product and normalize it (Sec. 4.4). Next, we discuss the complexity of the algorithm (Sec. 4.5). Finally we show the similarity to the greedy heuristic (Sec. 4.6).

4.1 The error metrics for range-sum queries

We define the error metrics by which the approximation error is measured. This is the mean-squared-error (MSE), measured over all possible *range-sum* queries.

Let $D = (d_0, ..., d_{N-1})$ be a sequence with $N = 2^j$ values. Let d_i be the i'th data value, and let $q_{l:r}$ be the range query $\sum_{i=l}^{r} d_i$. Let $d_{l:r}$ be the answer to the range query $q_{l:r}$ and let $\hat{d}_{l:r}$ be an approximated answer to the query $q_{l:r}$. The *absolute error* of the approximated answer is defined as $|e_{l:r}| = |d_{l:r} - \hat{d}_{l:r}|$. We can now define the mean-squared-error of any approximation that approximates all range-queries in some way. Such approximation defines a vector $\hat{R} = \left(\hat{d}_{1:1}, ..., \hat{d}_{1:N}, \hat{d}_{2:2}, ..., \hat{d}_{2:N}, ..., \hat{d}_{N:N}\right)$. A vector of approximated answers defines a vector of errors $E = (e_{1:1}, ..., e_{1:N}, e_{2:2}, ..., e_{2:N}, ..., e_{N:N})$. The MSE is defined as:

$$MSE_{range}\left(\hat{R}\right) = \frac{1}{(N+1)\,N/2} \sum_{i=1,...,N j=i,...,N} e_{i:j}^2$$

which is the sum of squared errors divided by the number of possible range-sum queries. Note that typically the sum of squared errors was measured only over point queries.

4.2 The prefix-sum based (PSB) inner product

We want to approximate a data vector $v \in R^N$ where $N = 2^j$. Our inner product, called *PSB inner product*, would be defined by the following symmetric bilinear form:

$$\mathbf{X} = \begin{pmatrix} N & -1 & \dots & -1 \\ -1 & N & \ddots & \vdots \\ \vdots & \ddots & \ddots & -1 \\ -1 & \dots & -1 & N \end{pmatrix} \tag{2}$$

That is, $\langle v, u \rangle := v^T X u$ where $v, u \in R^N$.

Lemma 2 *The PSB product is an inner product.*

Proof: Clearly $\langle v, u \rangle : R^N \times R^N \to R$, and the product is bilinear and symmetric. It remains to show that it is positive definite. The product is positive definite iff the matrix X is positive definite. A symmetric matrix is positive definite iff all its eigenvalues are positive. The only eigenvalues of X are $N+1$ (of dimension $N-1$) and 1 (of dimension 1) and therefore it is positive definite:
For $\lambda = N+1$:

$$A - \lambda I = \begin{pmatrix} -1 & -1 & \dots & -1 \\ -1 & -1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & -1 \\ -1 & \dots & -1 & -1 \end{pmatrix}$$

and thus $A - \lambda I = 0$ is an eigen space of dimension $N-1$.
For $\lambda = 1$:

$$A - \lambda I = \begin{pmatrix} N-1 & -1 & \dots & -1 \\ -1 & N-1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & -1 \\ -1 & \dots & -1 & N-1 \end{pmatrix}$$

and thus $A - \lambda I = 0$ is $sp\,(1, ..., 1)$, which is an eigen space of dimension $N-1$.

□

4.3 Defining a norm based on the PSB inner product

Based on the PSB inner product we define an Inner-Product-Based norm:

$$\|v\|_{IPB} = \sqrt{\langle v, v\rangle} \tag{3}$$

Our main lemma is:

Lemma 3 (main lemma) *Let $P = (p_1, ..., p_N)$ be a vector of prefix sums of the data. Let $\hat{P} = (\hat{p_1}, ..., \hat{p_N})$ be a vector that approximates it, by which we answer range-sum queries. Let $E_p = (p_1 - \hat{p_1}, ..., p_N - \hat{p_N}) = (e_{p_1}, ..., e_{p_N})$ be the error vector. Let $\hat{R}$ be the vector of approximations of all range-sum queries, answered using $\hat{P}$.*

Then,

$$\|E_p\|_{IPB}^2 = \frac{N\,(N+1)}{2} MSE_{range}\left(\hat{R}\right).$$

Proof:

Let $v \in R^N$ where $v = (v_1, \ldots, v_N)$. Then (note that at each stage we use part of the underbraced terms to create the overbraced terms of next stage):

$$\|v\|_{IPB}^2 = \langle v, v\rangle = v^T X v = \sum_{i,j} v_i \cdot X_{ij} \cdot v_j = \underbrace{\sum_{i=1}^{N} N \cdot v_i^2}_{} - \sum_{i=1...N, j>i} 2v_i v_j =$$

$$\overbrace{\sum_{i=1}^{N} v_i^2}^{} + \underbrace{\sum_{i=1}^{N} (N-1) \cdot v_i^2 - \sum_{i=1...N, j>i} 2v_i v_j}_{} =$$

$$\sum_{i=1}^{N} v_i^2 + \overbrace{\sum_{i=2}^{N} (v_i - v_1)^2}^{} + \underbrace{\sum_{i=2}^{N} (N-2) \cdot v_i^2 - \sum_{i=2...N, j>i} 2v_i v_j}_{} =$$

$$\sum_{i=1}^{N} v_i^2 + \sum_{i=2}^{N} (v_i - v_1)^2 + \overbrace{\sum_{i=3}^{N} (v_i - v_2)^2}^{} + \underbrace{\sum_{i=3}^{N} (N-3) \cdot v_i^2 - \sum_{i=3...N, j>i} 2v_i v_j}_{} =$$

$$\vdots$$

$$\sum_{i=1}^{N} v_i^2 + \sum_{i=2}^{N} (v_i - v_1)^2 + \sum_{i=3}^{N} (v_i - v_2)^2 + \cdots + (v_N - v_{N-1})^2$$

Now, let $D = (d_1, ..., d_N)$ be a vector of data values, which $P = (p_1, \ldots, p_N)$ is the vector of its prefix-sums ($p_i = \sum_{i=1}^{i} d_i$). Each range-sum query $d_{l:r}$ is computed by $d_{l:r} = p_r - p_{l-1}$ (p_{-1} is defined as 0 and is not part of the vector). Therefore the absolute error of a specific range sum query approximation is:

$$|e_{l:r}| = |d_{l:r} - \hat{d}_{l:r}| = |\,(p_r - p_{l-1}) - (\hat{p}_r - \hat{p}_{l-1})\,| =$$

$$|\,(p_r - \hat{p}_r) - (p_{l-1} - \hat{p}_{l-1})\,| = |e_{p_r} - e_{p_{(l-1)}}|$$

Let us compute the norm of the vector E as defined by the PSB inner product:

$$\|E_p\|^2_{IPB} = \langle E_p, E_p \rangle = \sum_{i=1}^{N} e^2_{p_i} + \sum_{i=2}^{N} (e_{p_i} - e_{p_1})^2 + \sum_{i=3}^{N} (e_{p_i} - e_{p_2})^2 + \cdots + \left(e_{p_N} - e_{p_{(N-1)}}\right)^2 =$$

$$\sum_{i=1}^{N} e^2_{1:i} + \sum_{i=2}^{N} e^2_{2:i} + \sum_{i=3}^{N} e^2_{3:i} + \cdots + e^2_{N:N} = \frac{N\,(N+1)}{2} MSE_{range}\left(\hat{R}\right)$$

This concludes our proof. □

Minimizing the inner-product-based norm is equivalent to minimizing the $MSE_{range}\left(\hat{R}\right)$ norm, since $\frac{N(N+1)}{2}$ is always positive and constant. We could equivalently normalize the inner product defined above by $\sqrt{\frac{N(N+1)}{2}}$. Our goal would be to minimize the above inner-product-based-norm of the vector of the prefix-sums approximations $\hat{P} = (\hat{p}_1, \ldots, \hat{p}_N)$. By proving the Haar basis is orthogonal with respect to the PSB inner product, we would be able to use Thm. 1: choosing the M largest normalized coefficients to our synopses (where M is the space limitation) would minimize the norm of the vector P, and thus the error of the approximation $MSE_{range}\left(\hat{R}\right)$.

4.4 Orthonormality of the Haar basis with respect to the PSB inner product

In this section we show the orthonormality of the Haar basis w.r.t. the PSB inner product. We show it in two stages. First we show that the Haar basis is orthogonal w.r.t. the PSB inner product. Then we show how to normalize the basis.

The Haar basis is usually treated as a basis for the piecewise-constant functions over the interval $[0,1)$, which are constant over intervals of the form $[\frac{i}{N}, \frac{i+1}{N})$ (e.g in [15]). We can equivalently treat the Haar basis functions as vectors in R^N, where the ith coordinate in the vector would be the value of the function over the ith interval. Thus there is a $1:1$ mapping between the space of piecewise constant functions defined in [15], and the space R^N. The Haar basis vectors (and equivalently functions) can be divided into "levels" according to the number of non-zero values in each one of them: in level i $(i = 0, ..., \log N - 1)$ there are $\frac{N}{2^i}$ non-zero coordinates, and their corresponding wavelet coefficient is at resolution level i in the error tree. For example, when $N = 4$ the basis vectors and the levels are:

$$\underbrace{\begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ -1 \\ -1 \end{pmatrix}}_{0} \underbrace{\begin{pmatrix} 1 \\ -1 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ -1 \end{pmatrix}}_{1}$$

(Actually, these are "unnormalized" Haar basis vectors where each non-zero value is ±1, instead of $\pm\sqrt{2^i}$. For simplicity and clarity, call this "unnormalized" Haar basis a "Haar basis").

Theorem 3 *The Haar basis is orthogonal with respect to the PSB inner product.*

Proof: In order to show that the Haar basis is orthogonal, let

$$u = (0, ..., 0, 1, ..., 1, -1, ..., -1, 0, ..., 0)$$

be a Haar basis vector at resolution level i. It is enough to show that u is orthogonal to all vectors at levels $\leq i$. We show it separately for levels $< i$ and for level i:

For level $<$ i. Let v be a vector at level $< i$. Recall that $\langle v, u\rangle = v^T X u$. It is easy to verify that

$$Xu = (0, ..., 0, (N+1), ..., (N+1), -(N+1), ..., -(N+1), 0, ..., 0)$$

where for each index j such that $u_j = 1$ we have $(Xu)_j = N+1$, and for each index j such that $u_j = -1$ we have $(Xu)_j = -(N+1)$. Note that the range of indexes of the non-zero coefficients is the same in u and in Xu. Denote this range as $l, ..., r$. Note that for each vector v at level $< i$, $v_l = v_{l+1} = ... = v_r$. Therefore:

$$v^T(Xu) = v_l \cdot (N+1) + \cdots + v_r \cdot -(N+1) =$$

$$v_l((N+1) + \cdots + (N+1) - (N+1) - \cdots - (N+1)) = 0$$

This completes the proof for vectors v at level $< i$.

For level $=$ i. Let v be a vector at level i. As shown in the previous part of the proof, the range of indexes of the non-zero coefficients in Xu, denoted $l, ...r$, is the same as in u. By the Haar basis definition, there is no overlapping between these ranges of v and u (two Haar basis vectors at the same level), so there is no overlap between these ranges of Xu and v, and therefore $v^T(Xu) = 0$.

Note. there is a special case in level 0, where the non-zero ranges overlaps. In this case the proof of the previous case should be used since $v_1 = ... = v_n$. □

As we have seen, the Haar basis is orthogonal with respect to our PSB inner product. We normalize each basis vector in order to have an orthonormal basis. For the first basis vector $u_1 = (1, \ldots, 1)$ it is easy to verify that its norm is $\|u_1\|_{IPB} = \sqrt{\langle u_1, u_1\rangle} = \sqrt{N}$. For any other basis vector v at level i its norm is $\|u\|_{IPB} = \sqrt{\frac{N}{2^i}(N+1)}$. In order to normalize the basis, we divide each basis vector by its norm. Transforming the basis w.r.t. the orthonotmal basis still takes linear time.

4.5 Building the optimal synopsis

First, the algorithm transforms the vector of prefix-sums with respect to the normalized Haar basis. Equivalently, the algorithm could transform the vector w.r.t. the *orthogonal* Haar basis and then normalize the wavelet coefficients (Sec. 3). The vector of prefix-sums, if not built yet, can be computed *during* the wavelet transform. Computing the Haar wavelet transform takes linear time using $O(N/B)$ I/Os. Next, the algorithm chooses the largest M coefficients which can be done in linear time using the *M-approximate quantile* algorithm [8]. Note that although there are $O(N^2)$ range-sum queries, our algorithm didn't use at any stage the vector of all possible queries. It was used just during the proof that $\|E_p\|_{IPB} = MSE_{range}\left(\hat{R}\right)$.

The following theorem follows from our construction, together with Thm. 1 and Thm. 3:

Theorem 4 *An optimal wavelet synopses for a vector of size N, which minimizes the* MSE *measured over all possible range-sum queries, can be constructed in linear-time, using $O(N/B)$ I/Os, for a disk block of size B.*

4.6 Comparison between the optimal thresholding and the greedy heuristic

The greedy heuristic for wavelet-synopses thresholding is commonly described as a two-stage process. First, the transform is computed w.r.t. the *orthogonal* Haar basis, where all non-zero coordinates are ±1. Then, the coefficients are normalized to be the coefficients of the linear combination w.r.t. the *normalized* basis (Sec. 3). We show that the greedy-heuristic thresholding is nearly identical to the optimal thresholding described in Thm. 4. Specifically, the resulting synopses may defer in at most a single coefficient.

The greedy heuristic transforms the data with respect to the orthogonal Haar basis, and normalizes each coefficient as follows: a coefficient of a vector at level i by multiplied by $\frac{1}{\sqrt{2^i}}$. Suppose that we scale all the coefficients of the greedy heuristic by multiplying them with the same factor $\sqrt{N(N+1)}$. Clearly, the greedy thresholding will still select the same coefficients to be part of the computed synopsis. Recall that the optimal synopsis computes the same Haar transform, and normalizes each coefficient as follows: a coefficient of the first basis vector is multiplied by $\sqrt{N}$, and any other coefficient is multiplied by $\sqrt{\frac{N}{2^i}(N+1)}$. As can be easily verified, except for the first coefficient, all coefficients in the optimal synopsis construction are identical to the scaled coefficients in the greedy heuristic. Therefore, the only possible difference between the optimal synopsis and the standard synopsis (obtained by the greedy heuristic) is a situation where the coefficient of v_0 is included in the standard synopsis but not in the optimal synopsis.

While the optimal synopsis and the standard synopsis are nearly identical, the difference in their error can be significant in extreme situations:

Theorem 5 *When using the greedy-heuristic that is based on point queries, instead of the above optimal thresholding, the mean-squared-error might be $\Theta\left(\sqrt{N}\right)$ times larger than the optimal error.*

Proof: Consider a wavelet transform that results in the following coefficients, normalized according to the greedy heuristic: $[\alpha_0, \ldots, \alpha_{N-1}] = [m, m, m, m, m-1, \epsilon, \epsilon, \ldots, \epsilon]$ and suppose that we have a synopsis consisting of 4 coefficients. The greedy heuristic would keep the first 4 coefficients, resulting with a mean-squared-error of $\sqrt{(m-1)^2 + (N-5)\cdot\epsilon^2}$, which is $\Theta(m)$ for $\epsilon = O(1/\sqrt{N})$. The optimal algorithm would normalize the first coefficient to $\frac{m}{\sqrt{N+1}}$, and consequently not keep it in the synopsis, but instead keep in the synopsis the next 4 coefficients: $m, m, m, m-1$. The error in this case is $\sqrt{\left(m/\sqrt{N+1}\right)^2 + (N-5)\cdot\epsilon^2}$, which is $\Theta\left(m/\sqrt{N}\right)$ for $\epsilon = O(1/\sqrt{N})$; that is, smaller by a factor of $\Theta(\sqrt{N})$ than that of the standard greedy heuristic. □

Comment: Vectors of prefix-sums tend to be monotone increasing in database-systems, as in many cases the raw-data has non-negative values (for example in selectivity estimation). In this case we should slightly change the proof so that the wavelet coefficients would be of a non-decreasing vector. We would fix the "small" coefficients to be $\epsilon, ..., \epsilon, \frac{\epsilon}{2}, ..., \frac{\epsilon}{2}, \frac{\epsilon}{4}, ...,$ according to their levels in the tree (in level i the "ϵ" coefficient would be divided by 2^i $(i < \log N)$. One can easily verify that the resulting vector would be monotone non-decreasing, and yet the wavelet coefficients are small enough, such that the proof stands.

5 The non-orthogonality of the Haar basis over raw data

In this section we define the inner product that corresponds to the MSE when answering range-sum queries over the raw data. We then show that the Haar basis is not orthogonal with respect to this inner product. Consequently, based on Lemma 1, the Haar basis is non-orthogonal w.r.t. the desired norm and Parseval's formula cannot be applied for optimal thresholding. We prove the non-orthogonality in two different ways. First we give an analytical proof, and then give a different, empirical proof. The latter also demonstrates an anomaly when using a non-orthogonal basis, where a larger synopsis may result with an increased error.

5.1 The inner product for range-sum queries

Lemma 4 (raw-data inner product) *Let $D = (d_1, ..., d_N)$ be a data vector. Let $\hat{D} = \left(\hat{d_1}, ..., \hat{d_N}\right)$ be a vector that approximates the raw data, D, built using the Haar-based wavelet synopsis. An answer to a range query $d_{l:r}$ is approximated as $\hat{d}_{l:r} = \sum_{i=l}^{r} \hat{d}_i$.*
As above, define $\hat{R} := \left(\hat{d}_{1:1}, ..., \hat{d}_{1:N}, \hat{d}_{2:2}, ..., \hat{d}_{2:N}, ..., \hat{d}_{N:N}\right)$. Denote:

$$X_{l:r} := \begin{array}{cc} \left(\begin{array}{ccc} & & \\ 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 1 \\ & & \end{array}\right) & \begin{array}{c} l \\ \\ r \end{array} \end{array}$$

and $X := \sum_{l=1,...,N \; r=l,...,N} X_{l:r}$. Let $E = (e_1, ..., e_N) = \left(d_1 - \hat{d_1}, ..., d_N - \hat{d_N}\right)$ be the vector of errors. Then:

1. *$E^T X E = \frac{N(N+1)}{2} MSE_{range}\left(\hat{R}\right)$.*

2. *$\langle v, u\rangle := v^T X u$ is an inner product.*

Proof:

1. When building the synopsis over the raw data, an answer to a range query that is approximated using $\hat{D}$ is $\hat{d_{l:r}} = \sum_{i=l}^{r} \hat{d_i}$. The absolute error over a range query $d_{l:r}$ is $|e_{l:r}| = |e_l + ... + e_r|$. Note that $E^T X_{l:r} E = e_{l:r}^2$. Thus,

$$E^T X E = E^T \left(\sum_{l=1,...,N \; r=l,...,N} X_{l:r}\right) E = \sum E^T X_{l:r} E = \sum e_{l:r}^2 = \frac{N(N+1)}{2} MSE_{range} \hat{R}$$

2. Clearly $\langle v, u\rangle : R^N \times R^N \to R$, and the product is Bilinear and symmetric, as a sum of symmetric matrices. It remains to show that it is positive definite, meaning that $\langle v, v\rangle \geq 0$ and $\langle v, v\rangle = 0 \iff v = 0$. Relying on the previous part of the proof, we get that $v^T X v \geq 0$, as a sum of squares. Denote $v_{l:r} = v_l + ... + v_r$. Then $v^T X v = \sum v_{l:r}^2$. Suppose that $v = 0$, then clearly $v^T X v = 0$. On the other hand, suppose $\langle v, v\rangle = 0$. That is, $0 = v^T X v = \sum v_{l:r}^2 = \sum_{l=r} v_{l:r}^2 + \sum_{l \neq r} v_{l:r}^2$. We get that $0 = \sum_{l=r} v_{l:r}^2 = \sum_{i=1}^{N} v_i^2$, meaning that $v = 0$.

□

5.2 The non-orthogonality of Haar basis - analytical proof

We show that the Haar basis is not orthogonal with respect to the above inner product. Consequently, based on Lemma 1, the Haar basis is non-orthogonal w.r.t. the desired norm and Parseval's formula cannot be applied for optimal thresholding. First, we find an expression for X_{ij}.

Lemma 5 *Let $1 \leq i \leq j \leq N$. Then, $X_{ij} = X_{ji} = (N - j + 1) \cdot i$*

Proof: Recall that $X := \sum_{l=1,...,N\ r=l,...,N} X_{l:r}$. Let us first assume that $i < j$. Each matrix $X_{l:r}$ in the sum has an additive contribution of 1 for each entry X_{ij}, with $l \leq i \leq j \leq r$. Given that $i < j$, the question is how many matrices $X_{l:r}$ contributes to the entry X_{ij}. A matrix $X_{l:r}$ contributes to an entry X_{ij} if $1 \leq l \leq i$ and $j \leq r \leq N$. The number of matrices that contributes for X_{ij} is thus $(i - 1 + 1) \cdot (N - j + 1) = (N - j + 1)\, i$. Note that $X_{ij} = X_{ji}$ since the matrix X is symmetric. □

Knowing the general expression of an entry X_{ij}, many examples can be given in order to show that the Haar basis is not orthogonal with respect to this inner product. For example, let us take the basis vector $u = (0, ..., 0, -1, 1)$, and another basis vector $v = (0, ..., 0, -1, 1, 0, ..., 0)$. It can be easily verified that $Xu = (1, 2, 3, ..., N - 1, -1)$. Thus, $\langle v, u \rangle = vXu \neq 0$, for every v in the level of u. This implies that more than $\left(\frac{N}{4}\right)^2$ pairs are non-orthogonal w.r.t. each other. Therefore, there is no easy modification to the Haar basis, based only on changes of a few vectors, that would make it orthogonal w.r.t. the inner product defined for the raw data case.

5.3 The non-orthogonality of Haar basis - empirical proof

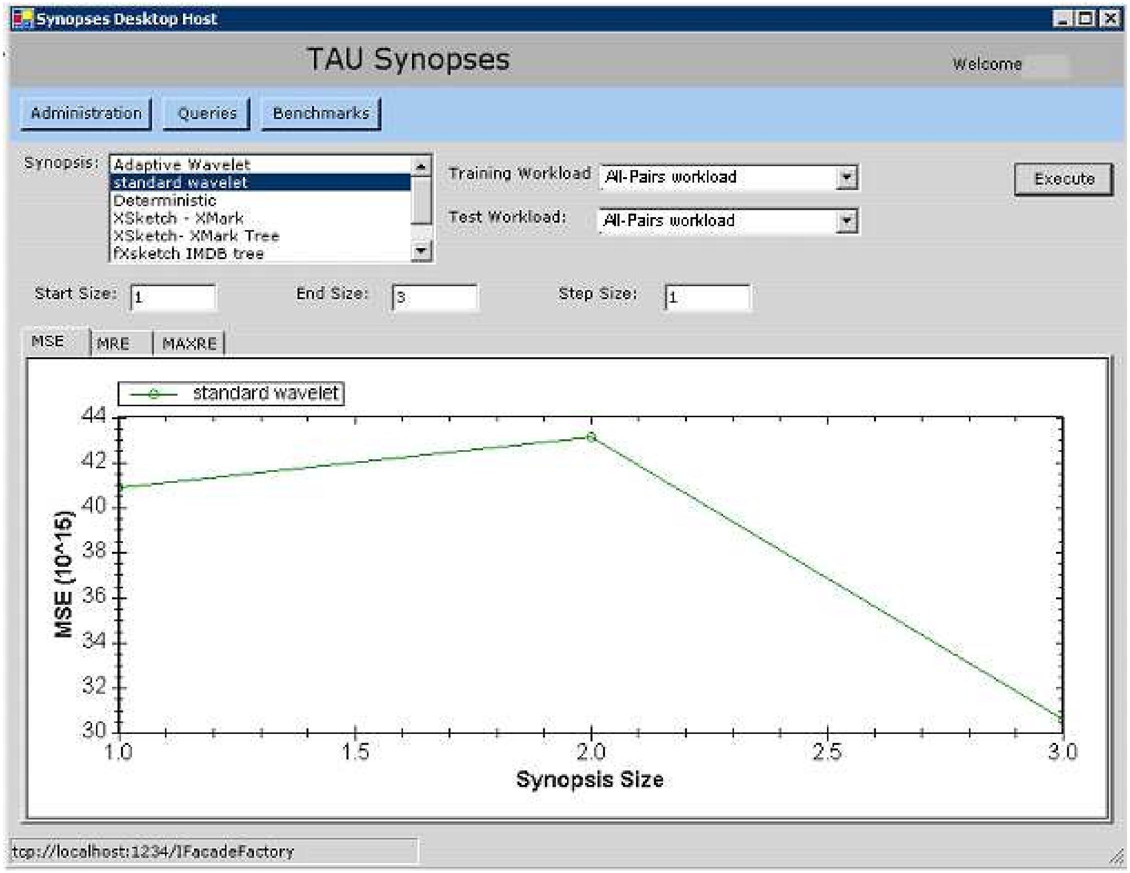
We give an additional, empirical proof for the non-orthogonality of the Haar basis. The following lemma is a straightforward implication of Thm. 1, and its generalization to orthogonal bases (Sec. 3).

Lemma 6 *Let S_1 and S_2 be two synopses built w.r.t. an orthogonal basis. Suppose $S_1 \subset S_2$, where $|S_1| = m - 1$ and $|S_2| = m$ for some m. The L_2 norm of the error for S_1 is greater than or equal to the L_2 norm of the error for S_2.*

Proof: As we have seen above, a linear combination of a vector v with respect to an orthogonal basis $\tilde{U}$, $v = \sum \tilde{\alpha_i} \tilde{u_i}$, can be represented with respect to the corresponding orthonormal basis U: $v = \sum \alpha_i u_i$, where $\alpha_i = \tilde{\alpha_i} \cdot \|\tilde{u_i}\|$ and $u_i = \frac{\tilde{u_i}}{\|u_i\|}$. Thus, when removing from S_2 any coefficient $\tilde{\alpha_i}$, the square of the error increases by $(\tilde{\alpha_i} \tilde{u_i})^2$ □

The above lemma implies that when starting with an empty synopsis, and adding coefficients gradually, the error as a function of synopsis size must be non-increasing.

We tested the monotonicity by conducting the following experiment. Our data vector was taken from a data set provided by KDD data of the University of California (http://kdd.ics.uci.edu). Specifically, we used data attribute Elevation from table CovTypeAgr filtered by Aspect, of size 512. The experiment was done using a small data set, and with synopsis sizes 1, 2 and 3, for purposes of demonstration. We built a sequence of synopses, each obtained from its preceding synopsis by adding one coefficient. For each synopsis, we measured the MSE over the set of all possible range-sum queries, which is the L_2 norm of the error vector; that is, the norm that is defined by the raw-data inner product. The measured errors are depicted in Fig. 2. As can be observed, the error is not a monotone, non-increasing function. Based on Lemma 6, this implies that the Haar basis is not orthogonal in this case, as the transform was done w.r.t. the Haar basis. The experiment also demonstrates an anomaly when using a non-orthogonal basis, where a larger synopsis may result with an increased error. Note that when the greedy heuristic is used over the prefix-sums, such a phenomenon cannot happen as the Haar basis is orthogonal.



Synopsis: standard wavelet

Training Workload All-Pairs workload

Test Workload: All-Pairs workload

Start Size: 1

End Size: 3

Step Size: 1

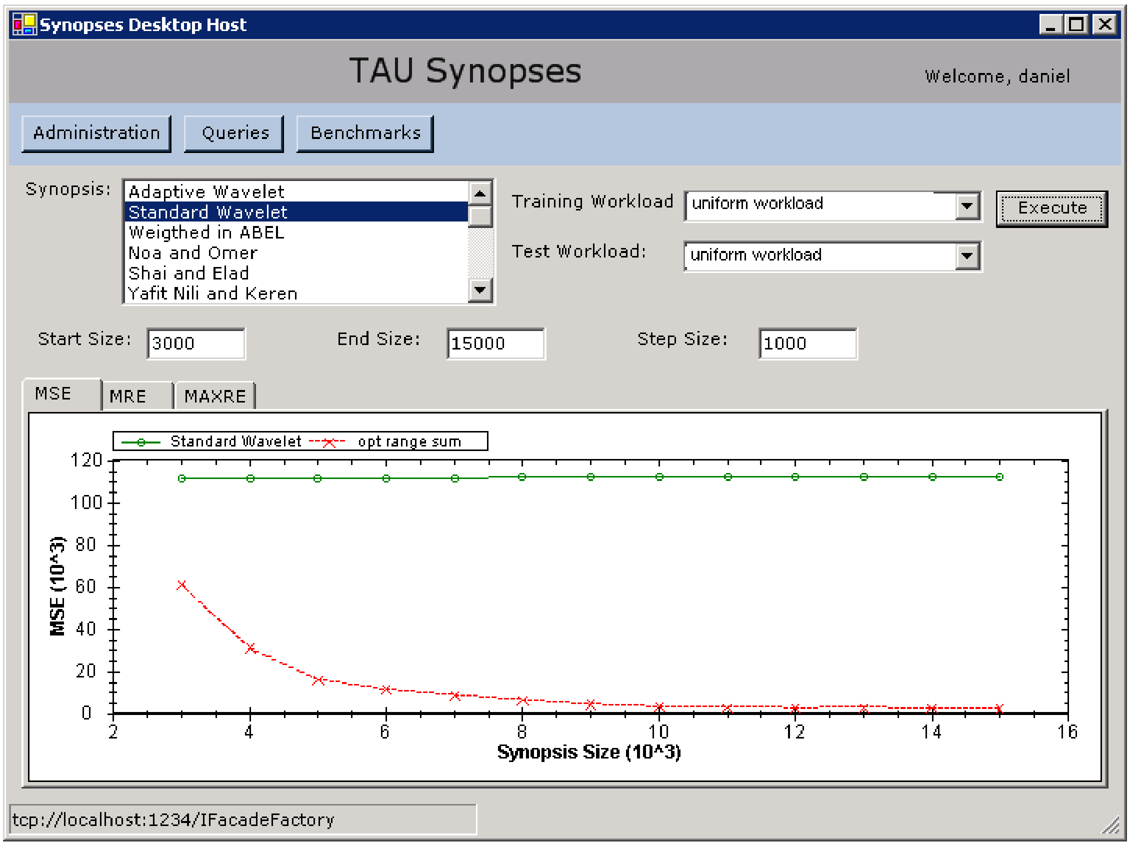
Figure 2: Wavelet transform over the raw data. The experimental results figure demonstrate that the error is not necessarily decreasing as synopsis size increases, for workloads of range-sum queries.

6 Experiments

In this section we demonstrate two points. First, we demonstrate cases where the prefix-sums based synopsis has a significant advantage over the raw-data based synopsis. Second, we compare the greedy heuristic with the optimal thresholding and demonstrate their possible difference for small synopses, and their similarity for sufficiently large synopses. All our experiments were done using the τ-synopses system [10].

Wavelet synopses built over the prefix-sums using the standard thresholding for point queries were shown through experimentation to be effective for answering range-sum queries [12, 18]. Recall that the main advantage of transforming the data into a representation with respect to a wavelet basis is that for data vectors containing similar values, many wavelet coefficients tend to have very small values. Eliminating such small coefficients introduces only small errors when reconstructing the original data, resulting in a very effective form of lossy data compression.

The above statement leads to the following two conclusions. First, when a wavelet synopsis is built over the raw data, a good approximation is achieved for data containing many *similar* values regardless of the actual values' sizes. Second, when a wavelet synopsis is built over the prefix sums of a data vector, a good approximation is achieved for data containing many *small* values, regardless of the smoothness of the original raw data vector. This is simply because the prefix-sums vector is



Synopsis: Standard Wavelet

Training Workload uniform workload

Test Workload: uniform workload

Start Size: 3000

End Size: 15000

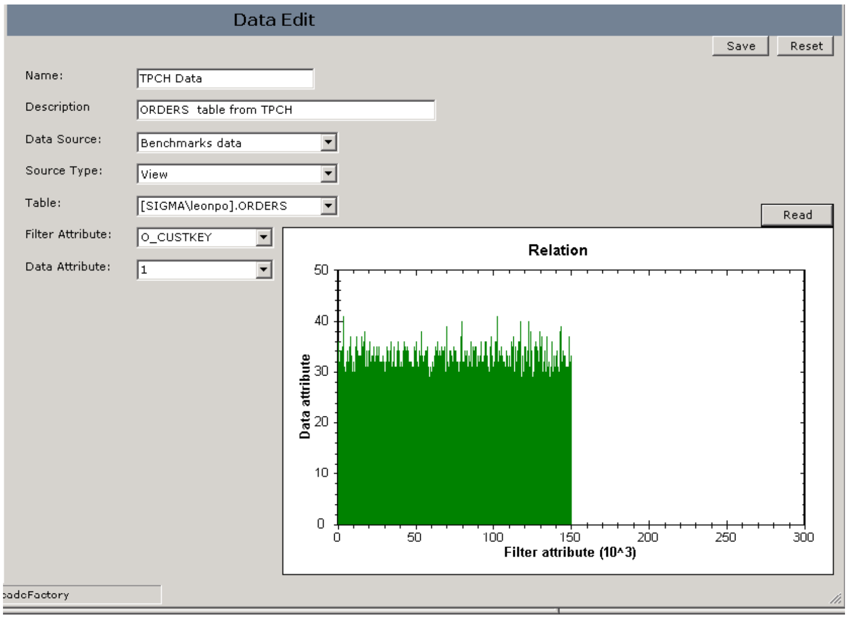
Step Size: 1000

tcp://localhost:1234/IFacadeFactory

Figure 3: Comparison between the standard greedy heuristic built over the *raw-data* ("standard wavelet"), and the optimal synopsis built over *prefix-sums* ("opt range sum"). The figure depicts the MSE as a function of synopsis size. The advantage of the prefix-sums-based synopsis over raw-data-based wavelet synopsis is demonstrated, in a case of a non-smooth data with relatively small values.

monotone where a difference between a pair of adjacent values is exactly *a value from the original data vector*, and thus we want it to be small. Experimental results demonstrate this advantage of the prefix-sums-based wavelet synopses over the raw-data-based synopses, as can be seen in Fig. 3. The experiment was done over data attribute 1 from the table ORDERS of the TPCH data, filtered by the attribute O_CUSTKEY, and which consists of about 150,000 distinct values (Fig. 4). The workload was generated uniformly in a quasi-random fasion. This data set has two properties that makes it suitable for prefix-sums based synopses, and not suitable for raw-data based synopses. It contains many small values, and adjacent values are significantly different from each other, as the data set is noisy.

The next experiment compares the greedy heuristic with the optimal synopsis, over a vector of prefix-sums of the original data. The experiment demonstrated the similarity of the two synopses. As pointed out, the standard synopsis is almost identical to the optimal synopsis presented here. Indeed, when normalizing the coefficients, the normalized values are all the same in both methods, except for the value of α_0 – the overall average, which according to the optimal method is divided by $\sqrt{N+1}$, thus making it smaller.



Name: TPCH Data

Description ORDERS table from TPCH

Data Source: Benchmarks data

Source Type: View

Table: [SIGMA\leonpo].ORDERS

Filter Attribute: O_CUSTKEY

Data Attribute: 1

Figure 4: The attribute of TPCH data over which the experiment was done. It can be seen that the data is non-smooth with relatively small values, making it more suitable for prefix-sums-based wavelet synopses than for raw-data-based wavelet synopses.

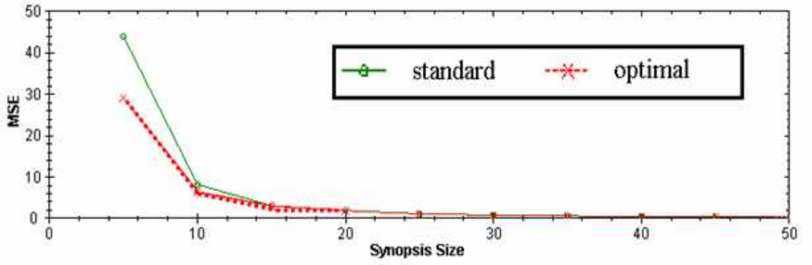


Figure 5: Error as a function of synopses size: Our optimal synopsis vs. the greedy heuristic-based standard synopsis; $rank(\alpha_0) = 15$. For small synopses sizes the standard synopsis' error is about 1.5 times larger than the optimal synopsis' error. Data size is 2048. Synopses sizes: 5, 10,...,50.

Recall that for both the standard synopsis and for the optimal synopsis, we select the M largest normalized coefficients to the synopses. Let k be the rank of α_0 among the normalized coefficients in the optimal synopsis; that is, α_0 is the k'th largest normalized value. Then, an optimal synopsis of size $\ell \geq k$ is the same as a standard synopsis of the same size. This is demonstrated in Fig. 5, which depicts the MSE as a function of synopses sizes for both methods. Here the rank of α_0 is 15. In this case we see that the difference between two synopses is about a factor of 2 for small synopses sizes. The experiment was done using the *Forest CoverType* data provided by KDD. Specifically, we used data attribute Aspect from table CovTypeAgr filtered by Elevation from the KDD data, with a total of 2048 distinct values. The experiment was done over a relatively small data set for purposes of illustration. The workload was generated uniformly in a quasi-random fasion.

7 Conclusions

In this paper we proved the near optimality of the greedy heuristic used for building wavelet synopses for range-sum queries over the vector of prefix-sums. Consequently, we introduced the first linear time construction of a provably optimal wavelet synopsis for range-sum queries. The technique we used for finding optimal synopses for range-sum queries is based on defining a suitable norm, a corresponding inner product, and showing the Haar basis is orthogonal with respect to this inner product. This enabled us to find an optimal wavelet synopsis efficiently, using a simple Parseval based thresholding algorithm. We have also presented the inner product that corresponds to the raw-data case, and we showed that the Haar basis is not orthogonal w.r.t. this inner product. Thus Parseval's formula cannot be applied for optimal thresholding over the vector of the raw data.

This paper leads to two interesting open problems. The first one is finding an optimal wavelet synopsis for range-sum queries over the raw-data representation. As we already defined the inner-product that defines the mean-squared error over the raw data, a future work can be finding an orthogonal, and consequently orthonormal, basis in order to find an optimal synopsis. The second one is finding an optimal workload-based wavelet synopsis for a workload of *range queries*. Recall that effective, yet non-optimal workload-based synopses for range queries were presented in [9, 14], and that efficient workload-based wavelet synopses for point queries were given in [11].

Acknowledgement. We thank Leon Portman for helpful discussions, and for his help with the τ-synopses system. We thank Yariv Matia for all his help in conducting our experiments.

References

[1] K. Chakrabarti, M. Garofalakis, R. Rastogi, and K. Shim. Approximate query processing using wavelets. In *VLDB 2000, Proceedings of 26th International Conference on Very Large Data Bases, 2000*, pages 111–122.

[2] A. Deligiannakis and N. Roussopoulos. Extended wavelets for multiple measures. In *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*, pages 229–240.

[3] M. Garofalakis and P. B. Gibbons. Wavelet synopses with error guarantees. In *Proceedings of the 2002 ACM SIGMOD International Conference on Management of Data*, 2002.

[4] M. Garofalakis and A. Kumar. Deterministic wavelet thresholding for maximum-error metrics. In *Proceedings of the 2004 ACM PODS international conference on on Management of data*, pages 166–176.

[5] P. B. Gibbons and Y. Matias. Synopsis data structures for massive data sets. In *DIMACS: Series in Discrete Mathematics and Theoretical Computer Science: Special Issue on External Memory Algorithms and Visualization, A*, 1999.

[6] A. C. Gilbert, Y. Kotidis, S. Muthukrishnan, and M. J. Strauss. Optimal and approximate computation of summary statistics for range aggregates. In *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 227–236. ACM Press, 2001.

[7] S. Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, 2nd edition, 1999.

[8] G. S. Manku, S. R., and B. G. Lindsay. Approximate medians and other quantiles in one pass and with limited memory. In *Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data*, pages 426–435, New York, 1998.

[9] Y. Matias and L. Portman. Workload-based wavelet synopses. Technical report, Department of Computer Science, Tel Aviv University, 2003.

[10] Y. Matias and L. Portman. τ-synopses: a system for run-time management of remote synopses. In *International conference on Extending Database Technology (EDBT), Software Demo, 865-867 & ICDE'04, Software Demo*, March 2004.

[11] Y. Matias and D. Urieli. Optimal workload-based weighted wavelet synopses. In *Proceedings of the 2005 ICDT conference*, Edinburgh, January 2005.

[12] Y. Matias, J. S. Vitter, and M. Wang. Wavelet-based histograms for selectivity estimation. In *Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data*, pages 448–459, Seattle, WA, June 1998.

[13] S. Muthukrishnan. Nonuniform sparse approximation using haar wavelet basis. Technical report, DIMACS, May 2004.

[14] L. Portman. Workload-based wavelet synopses. Master's thesis, School of Computer Science, Tel Aviv University, 2003.

[15] E. J. Stollnitz, T. D. Derose, and D. H. Salesin. *Wavelets for Computer Graphics.* Morgan Kaufmann, 1996.

[16] J. S. Vitter and M. Wang. Approximate computation of multidimensional aggregates of sparse data using wavelets. In *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data*, pages 193–204, Phildelphia, June 1999.

[17] J. S. Vitter, M. Wang, and B. Iyer. Data cube approximation and histograms via wavelets. In *Proceedings of Seventh International Conference on Information and Knowledge Management*, pages 96–104, Washington D.C., November 1998.

[18] M. Wang. *Approximation and Learning Techniques in Database Systems.* PhD thesis, Duke University, 1999.