Optimal Workload-based Weighted Wavelet Synopses

Yossi Matias and Daniel Urieli

School of Computer Science
Tel-Aviv University
{matias,daniel1}@tau.ac.il

Abstract. In recent years wavelets were shown to be effective data synopses. We are concerned with the problem of finding efficiently wavelet synopses for massive data sets, in situations where information about query workload is available. We present linear time, I/O optimal algorithms for building optimal workload-based wavelet synopses for point queries. The synopses are based on a novel construction of weighted inner-products and use weighted wavelets that are adapted to those products. The synopses are optimal in the sense that the subset of retained coefficients is the best possible for the bases in use with respect to either the mean-squared absolute or relative errors. For the latter, this is the first optimal wavelet synopsis even for the regular, non-workload-based case. Experimental results demonstrate the advantage obtained by the new optimal wavelet synopses, as well as the robustness of the synopses to deviations in the actual query workload.

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 [10]. 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 [18, 24, 22, 23, 2, 6, 9, 8], 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 sets where the purpose is 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 [18, 24]: in this case only *portions* of the data are reconstructed each time, in response to user queries, rather than the whole data at once. As a result, portions of the data that are used for answering frequent queries are reconstructed more frequently than portions of the data that correspond to rare queries. Therefore, the approximation error is measured over the *multi-set of actual queries*, rather than over the data itself. For more wavelet synopses basics see [18, 24].

Another aspect of the use of wavelets in database systems is that due to the large data-sizes in databases (giga-, tera- and peta-bytes), the efficiency of building wavelet synopses is of primary importance. Disk I/Os should be minimized as much as possible, and non-linear-time algorithms may be unacceptable.

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.

Generally speaking, we can characterize a wavelet approximation by three attributes: how the approximation error is measured, what wavelet basis is used and how coefficient thresholding is done. Many bases were suggested and used in traditional wavelets literature. Given a basis with respect to which the 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 some approximation-error measure. This subset is called an *optimal wavelet synopsis*, with respect to the chosen error measure.

While there has been a considerable work on wavelet synopses and their applications [18, 24, 22, 23, 2, 6, 14, 9, 8], so far there were only a few optimality results. The first one is a linear-time Parseval-based algorithm, which was used in traditional wavelets literature (e.g [12]), where the error was measured over the *data*. This algorithm minimizes the L_2 norm of the error vector, and equivalently it minimizes the mean-squared-absolute error over all possible point queries [18, 24]. No algorithm that minimizes the mean-squared-relative error over all possible point queries was known. The second one, introduced recently [9], is a polynomial-time ($O(N^2 M \log M)$) algorithm that minimizes the max relative or max absolute error over all possible point queries. Another optimality result is a polynomial time dynamic-programming algorithm that obtains an optimal wavelet synopsis over multiple measures [6]. The synopsis is optimal w.r.t. an error metric defined as weighted combination of L_2 norms over the multiple measures (this weighted combination has no relation with the notion of weighted wavelets of this paper).

Workload-based wavelet synopses. In recent years there is increased interest in workload-based synopses – synopses that are adapted to a given query workload, with the assumption that the workload represents (approximately) a probability distribution from which future queries will be taken. Chaudhuri et al [4] argue that identifying an appropriate precomputed sample that avoids large errors on an *arbitrary* query is virtually impossible. To minimize the effects of this problem, previous studies have proposed using the *workload* to guide the process of selecting samples [1, 3, 7]. By picking a sample that is tuned to the given workload, we can reduce the error over frequent (or otherwise "important") queries in the workload.

In [4], the authors formulate the problem of pre-computing a sample as an *optimization* problem, whose goal is to pick a sample that minimizes the error for the given workload.

Recently, *workload-based wavelet synopses* were proposed by Portman and Matias [14, 20]. Using an adaptive-greedy algorithm, the query-workload information was used during the thresholding process in order to build a wavelet synopsis that reduces the error w.r.t. to the query workload. These workload-based wavelet synopses demonstrate significant imporvement with respect to prior synopses. They are, however, not optimal w.r.t. the query workload.

In this paper, we address the problem of finding efficiently *optimal* workload-based wavelet synopses.

1.1 Contributions

We introduce efficient algorithms for finding optimal workload-based wavelet synopses using *weighted Haar (WH)* wavelets, for workloads of point queries. Our main contributions are:

- Linear-time, I/O optimal algorithms that find optimal Workload-based Weighted Wavelet (WWW) synopses[1]:
  - An optimal synopsis w.r.t. workload-based mean-squared *absolute*-error (*WB-MSE*).
  - An optimal synopsis w.r.t. workload-based mean-squared *relative*-error (*WB-MRE*).

  Equivalently, the algorithms minimize the *expected* squared, absolute or relative errors over a point query taken from a given distribution.
- The *WB-MRE* algorithm, used with uniform workload, is also the first algorithm that minimizes the mean-squared-relative-error over the *data values*, with respect to a wavelet basis.

[1] No relation whatsover to the world-wide-web.

- Both WWW synopses are also optimal with respect to *enhanced wavelet synopses*, which allow changing the values of the synopses coefficients to arbitrary values.
- Experimental results show the advantage of our synopses with respect to existing synopses.
- The synopses are robust to deviation from the pre-defined workload, as demonstrated by our experiments.

The above results were obtained using the following novel techniques.

- We define the problem of finding optimal workload-based wavelet synopses in terms of a *weighted norm*, a *weighted-inner-product* and a *weighted-inner-product-space*. This enables linear time I/O optimal algorithms for building optimal workload-based wavelet synopses.
  The approach of using a weighted inner product can also be used to the general case in which each data point is given different priority, representing its significance. This generalization is used to obtain the optimal synopses for mean relative error, where the weight of each point is normalized by its value. Using these weights, one can find a weighted-wavelet basis, and an optimal weighted wavelet synopsis in linear time, with $O(N/B)$ I/Os.
- We introduce the use of *weighted wavelets* for data synopses. Using weighted wavelets [5, 11] enables finding optimal workload-based wavelet synopses efficiently. In contrast, it is not known how to obtain optimal workload-based wavelet synopses with respect to the Haar basis efficiently. If we ignore the efficiency of finding a synopsis, the Haar basis is as good as the weighted Haar basis for approximation.

In the wavelets literature (e.g., [12]), wavelets are used to approximate a given signal, which is treated as a vector in an inner-product space. Since an inner-product defines an L_2 norm, the approximation error is measured as the L_2 norm of the error vector, which is the difference between the approximated vector and the approximating vector. Many wavelet bases were used for approximation, as different bases are adequate for approximating different collections of data vectors. By using an orthonormal wavelet basis, an optimal coefficient thresholding can be achieved in linear time, based on Parseval's formula. When using non-orthogonal wavelet basis, or measuring the error using other norms (e.g., L_∞), it is not known whether an optimal coefficient thresholding can be found efficiently, so usually non-optimal greedy algorithms are used in practice.

A weighted Haar (WH) basis is a generalization of the standard Haar basis, which is typically used for wavelet synopses due to its simplicity. There are several attributes by which a wavelet basis is characterized, which affects the quality of the approximations achieved using this basis (for full discussion, see [12]). These attribute are: the set of nested spaces of increasing resolution which the basis spans, the number of vanishing moments of the basis, and its compact support (if exists). Both Haar basis and a WH basis span the same subsets of nested spaces, have one vanishing moment, and a compact support of size 1.

Haar basis is orthonormal for uniform workload of point queries. Hence it is optimal for the MSE error measure. The WH basis is orthonormal with respect to the *weighted* inner-product defined by the problem of finding optimal workload-based wavelet synopses. As a result, an optimal workload-based synopses with respect to WH basis is achieved efficiently, based on Parseval's formula, while for the Haar basis no efficient optimal thresholding algorithm is known, in cases other than uniform workload.

1.2 Paper Outline

The rest of the paper is organized as follows. In Sec. 2 we describe our basic approach, including the workload-based error metrics and optimal thresholding in orthonormal bases. In Sec. 3 we define the problem of finding optimal workload-based wavelet synopses in terms of weighted inner product, and solve it using an orthonormal basis. In Sec. 4 we describe the optimal algorithm for minimizing *WB-MSE*, which is based on the construction of Sec. 3. In Sec. 5 we extend the algorithm to work for the *WB-MRE*, and in Sec. 6 we draw our conclusions. Due to space limitations, some technical proofs and additional experiments can be found in the full paper [17].

2 Basics

2.1 Workload-based Error Metrics

Let $D = (d_0, ..., d_{N-1})$ be a sequence with $N = 2^j$ values. Denote the set of point queries as $Q = (q_0, ..., q_{N-1})$, where q_i is a query which its answer is d_i. Let a workload $W = (c_0, ..., c_{N-1})$ be a vector of weights that represents the probability distribution from which future point queries are to be generated. Let $(u_0, ..., u_{N-1})$ be a basis of R^N, than $D = \sum_{i=0}^{N} \alpha_i u_i$. We can represent D by a vector of coefficients $(\alpha_0, ..., \alpha_{N-1})$.

Suppose we want to approximate D using a subset of the coefficients $S \subset \{\alpha_0, ..., \alpha_{N-1}\}$ where $|S| = M$. Then, for any subset S we can define a weighted norm WL_2 with respect to S, that provides a measure for the errors expected for queries drawn from the probability distribution represented by W, when using S as a synopsis. S is then referred to as a *workload-based wavelet synopsis*.

Denote $\hat{d}_i$ as an approximation of d_i using S. There are two standard ways to measure the error over the i'th data value (equivalently, *point query*):
The absolute error: $e_{\mathrm{a}}(i) = e_{\mathrm{a}}(q_i) = |d_i - \hat{d}_i|$; and *the relative error*: $e_{\mathrm{r}}(i) = e_{\mathrm{r}}(q_i) = \frac{|d_i - \hat{d}_i|}{max\{|d_i|, s\}}$, where s is a positive bound that prevents small values from dominating the relative error.

While the standard (non-workload-based) approach is to reduce the L_2 norm of the vector of errors $(e_1, ..., e_N)$ (where $e_i = e_{\mathrm{a}}(i)$ or $e_i = e_{\mathrm{r}}(i)$), here we would generalize the L_2 norm to reflect the query workload. Let W be a given *workload* consisting of a vector of queries' probabilities $c_1, ..., c_N$, where c_i is the probability that q_i occurs; that is, $0 < c_i \leq 1$, and $\sum_{i=0}^{N-1} c_i = 1$. The *weighted-L_2 norm* of the vector of (absolute or relative) errors $e = (e_1, ..., e_N)$ is defined as:

$$WL_2 (e) = \|e\|_{\mathrm{w}} = \sqrt{\sum_{i=0}^{N-1} c_i \cdot e_i^2}$$

where $0 < c_i \leq 1$, $\sum_{i=0}^{N-1} c_i = 1$. Thus, each data value d_i, or equivalently each point query q_i, is given some weight c_i that represents its significance. Note that WL_2 norm is the square-root of the mean squared error for a point query that is drawn from the given distribution. Thus, minimizing that norm of the error is equivalent to minimizing the *mean squared error of an answer to a query*.

In general, the weights given to data values need not necessarily represent a probability distribution of point queries, but any other significance measure. For example, in Sec. 5 we use weights to solve the problem of minimizing the mean-squared relative error measured over the *data values* (the non-workload-based case).

Notice that it is a generalization of the MSE norm: by taking equal weights for each query, meaning $c_i = \frac{1}{N}$ for each i and $e_i = e_{\mathrm{a}} (i)$, we get the standard MSE norm. We use the term *workload-based error* for the WL_2 norm of the vector of errors e. When e_i are absolute (resp. relative) errors the workload-based error would be called the *WB-MSE* (resp. *WB-MRE*).

2.2 Optimal Thresholding in Orthonormal Bases

The construction is based on Parseval's formula, and a known theorem that results from it (Thm. 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 than $e = \sum_{i \notin S} \alpha_i u_i$. 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 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 synopses by choosing the largest M coefficients.

2.3 Optimality Over Enhanced Wavelet Synopses

Notice that in the previous section 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.

Theorem 2. *When using an orthonormal basis, choosing the largest M coefficients with original values is an optimal enhanced 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 would change the first j coefficients, meaning, we would 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: $\|e\|^2 = \langle v - \tilde{v}', v - \tilde{v}' \rangle = \sum_{i=1}^{j} (\alpha_i - \alpha'_i)^2 + \sum_{i=M+1}^{N} \alpha_i^2 > \sum_{i=M+1}^{N} \alpha_i^2$.

3 The Workload-based Inner Product

In this section, we define the problem of finding an optimal workload-based synopses in terms of a weighted-inner-product space, and solve it relying on this construction. Here we deal with the case where e_i are the absolute errors (the algorithm minimizes the *WB-MSE*). An extension to relative errors (*WB-MRE*) is introduced in Sec. 5
Our development is as follows:

1. Transforming the data vector D into an equivalent representation as a function f in a space of piecewise constant functions over $[0, 1)$. (Sec. 3.1)
2. Defining the *workload-based inner product.* (Sec. 3.2)
3. Using the inner product to define an L_2 norm, showing that the newly defined norm is equivalent to the *weighted L_2 norm* (WL_2). (Sec. 3.3)
4. Defining a *weighted Haar basis* which is orthonormal with respect to the new inner product. (Sec. 3.4)

Based on Thm. 1 and Thm. 2 one can easily find an optimal workload-based wavelet synopses with respect to a weighted Haar wavelet basis.

3.1 Transforming the Data Vector into a Piecewise Constant Function

We assume that our approximated data vector D is of size $N = 2^j$. As in [21], we treat sequences (vectors) of 2^j points as piecewise constant functions defined on the half-open interval $[0, 1)$. In order to do so, we will use the concept of a vector space from linear algebra. A sequence of one point is just a function that is constant over the entire interval $[0, 1)$; we'll let V_0 be the space of all these functions. A sequence of 2 points is a function that has two constant parts over the intervals $[0, \frac{1}{2})$ and $[\frac{1}{2}, 1)$. We'll call the space containing all these functions V_1. If we continue in this manner, the space V_j will include all piecewise constant functions on the interval $[0, 1)$, with the interval divided equally into 2^j different sub-intervals. We can now think of every one-dimensional sequence D of 2^j values as being an element, or vector f, in V_j.

3.2 Defining a Workload-based Inner Product

The first step is to choose an inner product defined on the vector space V_j. Since we want to minimize a *workload based error* (and not the regular L_2 error), we started by defining a new *workload based inner product.* The new inner product is a generalization of the standard inner product. It is a sum of $N = 2^j$ weighted standard products; each of them is defined over an interval of size $\frac{1}{N}$:

$$\langle f, g \rangle = N \cdot \left(\sum_{i=0}^{N-1} c_i \int_{\frac{i}{N}}^{\frac{i+1}{N}} f(x)\, g(x)\, dx \right) \; where\; 0 < c_i \leq 1,\; \sum_{i=0}^{N-1} c_i = 1 \qquad (2)$$

Lemma 1. *$\langle f, g \rangle$ is an inner product.*

The proof of the lemma can be found in the full paper. As mentioned before, a coefficient c_i represents the probability (or a weight) for the i'th point query (q_i) to appear. Notice that the answer of which is the ith data value, which is function value at the i'th interval. When all coefficients c_i are equal to $\frac{1}{N}$ (a uniform distribution of queries), we get the standard inner product, and therefore this is a generalization of the standard inner product.

3.3 Defining a Norm Based on the Inner Product

Based on that inner product we define an inner-product-based (IPB) norm:

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

Lemma 2. *The norm $\|f\|_{\mathrm{IPB}}$ measured over the vector of absolute errors is the weighted L_2 norm of this vector, i.e $\|e\|^2_{\mathrm{IPB}} = \sum_{i=0}^{N-1} c_i e_i^2 = \|e\|^2_{\mathrm{w}}$.*

The proof of the lemma can be found in the full paper. Notice that when all coefficients are equal to $\frac{1}{N}$ we get the regular L_2 norm, and therefore this is a generalization of the regular L_2 norm (MSE).
Our goal is to minimize the *workload based error* which is the WL_2 norm of the vector of errors.

3.4 Defining an Orthonormal Basis

At this stage we would like to use Thm. 1. The next step would thus be finding an orthonormal (with respect to a workload based inner product) wavelet basis for the space V_j . The basis is a *Weighted Haar Basis.* For each workload-based inner product (defined by a given query workload) there is corresponding orthonormal weighted Haar basis, and our algorithm finds this basis in linear time, given the workload of point queries. We describe the bases here, and see how to find a basis based on a given workload of point queries. We will later use this information in the algorithmic part.

In order to build a weighted Haar basis, we take the Haar basis functions and for the k'th basis function we multiply its positive (resp. negative) part by some x_k (resp. y_k). We would like to choose such x_k and y_k so that we get an orthonormal basis with respect to our inner product. Thus, instead of using Haar basis functions (Fig. 1), we use functions of the kind illustrated in Fig. 2, where x_k and y_k are not necessarily (and probably not) equal, so our basis looks like the one in (Fig. 3). One needs to show how to choose x_k and y_k.

Let u_k be some Haar basis function as described above. Let $[a_{k_0}, a_{k_1})$ be the interval over which the basis function is positive and let $[a_{k_1}, a_{k_2})$ be the interval over which the function is negative. Recall that a_{k_0}, a_{k_1} and a_{k_2} are both multiples of $\frac{1}{N}$ and therefore the interval precisely contains some number of continuous intervals of the form $[\frac{i}{N}, \frac{i+1}{N}]$ (also $a_{k_1} = \frac{a_{k_0}+a_{k_2}}{2}$). Moreover, the size of the interval over which the function is positive (resp. negative) is $\frac{1}{2^i}$ for some $i < j$ (As we remember, $N = 2^j$). Recall that for the i'th interval of size $\frac{1}{N}$, meaning $[\frac{i}{N}, \frac{i+1}{N})$ there is a corresponding weight coefficient c_i which is the coefficient that is used in the inner product. Notice that each Haar basis function is positive (negative) over some number of (whole) such intervals. We can therefore associate the sum of coefficients of the intervals "under" the positive (negative) part of the function with the positive (negative) part of the function. Let us denote the sum of weight coefficients (c_i's) corresponding to intervals that are under the positive (resp. negative) as l_k (resp. r_k).

Lemma 3. *Suppose for each Haar basis function v_k we choose x_k and y_k such that*

$$x_k = \sqrt{\frac{r_k}{l_k r_k + l_k^2}} \quad y_k = \sqrt{\frac{l_k}{l_k r_k + r_k^2}}$$

and multiply the positive (resp. negative) part of v_k by x_k (resp. y_k); by doing that we get an orthonormal set of $N = 2^j$ functions, meaning we get an orthonormal basis.

The proof of the lemma can be found in the full paper. Again, notice that had all the workload coefficients been equal ($c_i = \frac{1}{N}$) we would get the standard Haar basis used to minimize the standard L_2 norm.

As we have seen, this is an orthonormal basis to our function space. In order to see that it is a wavelet basis, we can notice that for each $k = 1, ..., j$, the first 2^k functions are an orthonormal set belonging to V_k (its dimension is 2^k) and which is therefore a basis of V_k.

4 The Algorithm for the WWW Transform

In this section we describe the algorithmic part. Given a workload of point queries and a data vector to be approximated, we build workload-based wavelet synopses of the data vector using a weighted Haar basis. The algorithm has two parts:

1. Computing efficiently a *Weighted Haar basis*, given a workload of point queries. (Sec. 4.1)
2. Computing efficiently the *Weighted Haar Wavelet Transform* with respect to the chosen basis. (Sec. 4.2)

4.1 Computing Efficiently a Weighted Haar Basis

Note that at this point we already have a method to find an orthonormal basis with respect to a given workload based inner product. Recall that in order to know x_k and y_k for every basis function we need to know the corresponding l_k and r_k. We are going to compute all those partial sums in linear time. Suppose that the basis functions are arranged in an array like in a binary tree representation. The highest resolution functions are at indexes $\frac{N}{2}, ..., N-1$, which are the lowest level of the tree. The next resolution level functions are at indexes $\frac{N}{4}, ..., \frac{N}{2} - 1$, and so on, until the constant basis function is in index 0. Notice that for the lowest level (highest resolution) functions (indexes $\frac{N}{2}, ..., N-1$) we already have their l_k's and r_k's. These are exactly the workload coefficients. It can be easily seen in Fig. 3 for the lower four functions. Notice that after computing the accumulated sums for the functions at resolution level i, we have all the information to compute the higher level functions: let u_k be a function at resolution level i and u_{2k}, u_{2k+1} be at level $i+1$, where their supports included in u_k's support (u_k is their ancestor in the binary tree of functions). We can use the following formula for computing l_k and r_k:

$$l_k = l_{2k} + r_{2k} \quad r_k = l_{2k+1} + r_{2k+1}$$

See Fig. 3. Thus, we can compute in one pass only the lowest level, and build the upper levels bottom-up (in a way somewhat similar to the Haar wavelet transform). The algorithm consists of phases, where in each phase the functions of a specific level are computed. At the end of a phase, we keep a temporary array holding all the pairwise sums of all the l_k's and r_k's from that phase and use them for computing the next phase functions. Clearly, the running time is $\frac{N}{2} + \frac{N}{4} + ... + 1 = O(N)$. The number of I/Os is $O(N/B)$ I/Os (where B is the block size of the disk) – since the process is similar to the computation Haar wavelet transform. Recall that given r_k and l_k, one can easily compute the k'th basis function (its positive and negative parts) using the following formula:

$$x_k = \sqrt{\frac{r_k}{l_k r_k + l_k^2}} \quad y_k = \sqrt{\frac{l_k}{l_k r_k + r_k^2}}$$

4.2 Computing a Weighted Haar Wavelet Transform

Given the basis we would like to efficiently perform the wavelet transform with respect to that basis. Let us look at the case of $N = 2$ (Fig. 4). Suppose we would like to represent the function in Fig. 5. It is easy to compute the following result (denote α_i as the coefficient of f_i):

$$\alpha_0 = \frac{y v_0 + x v_1}{x + y} \qquad \alpha_1 = \frac{v_0 - v_1}{x + y}$$

(by solving 2x2 matrix). Notice that the coefficients are weighted averages and differences, since the transform generalizes the standard Haar transform (by taking $x = y = \sqrt{2^i}$ we get the standard Haar transform). It's easy to reconstruct the original function from the coefficients:

$$v_0 = \alpha_0 + x\alpha_1 \qquad v_1 = \alpha_0 - y\alpha_1$$

This implies a straightforward method to compute the wavelet transform (which is I/O efficient as well) according to the way we compute a regular wavelet transform with respect to the Haar basis: we go over the data, and compute the weighted differences which are the coefficients of the bottom level functions. We keep the weighted averages, which can be represented *solely* by the rest of the basis functions (the "lower resolution" functions - as in the regular Haar wavelet transform), in another array. We repeat the process over the averages time and time again until we have the overall average, which is added to our array as the coefficient of the constant function ($v_0(x) = \mathit{const}$). While computing the transform, in addition to reading the values of the signal, we need to read the proper basis function that is relevant for the current stage (in order to use the x_k and y_k of the function that is employed in the above formula). This is easy to do, since all the functions are stored in an array F and the index of a function is determined by the iteration number and is identical to the index of the corresponding currently computed coefficient. A pseudo code of the algorithm can be found in the full paper.

The steps of our algorithm are identical to the steps of the Haar algorithm, with the addition of reading the data at $F[i]$ (the x_k and y_k of the function) during the i'th iteration. Therefore the I/O complexity of that phase remains $O(N/B)$ (B is the disk block size) with $O(N)$ running time.

After obtaining the coefficient of the orthonormal basis we keep the largest M coefficients, along with their corresponding M functions, and throw the smallest coefficients. This can be done efficiently using an *M-approximate quantile algorithm* [13]. Based on Thm. 1 we obtain an optimal synopsis.

5 Optimal Synopsis for Mean Relative Error

We show how to minimize the weighted L_2 norm of the vector of *relative* errors, weighted by the query workload, by using weighted wavelets. As a special case, this minimizes the mean-squared-relative-error measured over the data values.

Recall that in order to minimize the weighted L_2 norm of relative errors, we need to minimize $\sum_{i=1}^{N} c_i \left(\frac{|d_i - \hat{d}_i|}{max\{d_i, s\}}\right)^2$. For simplicity, we show instead how to minimize $\sum_{i=1}^{N} c_i \left(\frac{|d_i - \hat{d}_i|}{d_i}\right)^2$; the extension to the above is straightforward. Since $D = d_1, ..., d_N$ is part of the input of the algorithm, it is fixed throughout the algorithm's execution. We can thus divide each c_i by d_i^2 and get a new vector of weights: $W = \left(\frac{c_1}{d_1^2}, ..., \frac{c_N}{d_N^2}\right)$. Relying on our previous results, and using the new vector of weights we minimize $\sum_{i=1}^{N} \frac{c_i}{d_i^2} \left(|d_i - \hat{d}_i|\right)^2 = \sum_{i=1}^{N} c_i \left(\frac{|d_i - \hat{d}_i|}{d_i}\right)^2$, which is the WL_2 norm of relative errors. Notice that in the case $b_i = \frac{1}{N}$ (the uniform case) the algorithm minimizes the mean-relative-error over all *data values*. As far as we know, this is the first algorithm that minimizes the mean-relative-error over the data values.

6 Conclusions

In this paper we introduce the use of weighted wavelets for building optimal workload-based wavelet synopses. We present two time-optimal and I/O-optimal algorithms for workload-based wavelet synopses, which minimize the WB-MSE and and the WB-MRE error measures, with respect to any given query workload. The advantage of optimal workload-based wavelet synopses, as well as their robustness, were demonstrated by experimentations (in the full paper).

Recently, and independently of our work, Muthukrishnan [19] presented an optimal workload-based wavelet synopsis with respect to the standard *Haar* basis. The algorithm for building the optimal synopsis is based on dynamic programming and takes $O(N^2M/\log M)$ time. As noted above, standard Haar basis is not orthonormal w.r.t. the workload-based error metric, and an optimal synopsis w.r.t. this basis is not necessarily also an optimal enhanced wavelet synopsis. Obtaining optimal enhanced wavelet synopses for the standard Haar wavelets may be an interesting open problem. Also, as quadratic time is too costly for massive data sets, it may be interesting to obtain a time efficient algorithm for such synopses. As far as approximation error is concerned, although in general optimal synopses w.r.t. the standard Haar and the weighted Haar bases are incomparable, both bases have the same characteristics. It would be interesting to compare the actual approximation errors of the two synopses for various data sets. This may indeed be the subject of a future work.

In a recent related paper [16], we show how to find optimal wavelet synopses for range-sum queries, using a framework similar to the one used in this paper. We define the problem of finding an optimal synopsis for range-sum queries in terms of a proper inner-product, and find an optimal synopsis, which minimizes the MSE measured over all possible range-sum queries, in linear time, with $O(N/B)$ I/Os.

Acknowledgments: We thank Leon Portman for helpful discussions and for his assistance in setting up the experiments on the τ-synopses system [15]. We also thank Prof. Nira Dyn for helpful discussions regarding the wavelets theory.

References

1. A. Aboulnaga and S. Chaudhuri. Self-tuning histograms: Building histograms without looking at data. In *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data*, pages 181–192, 1999.
2. 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.
3. S. Chaudhuri, G. Das, M. Datar, R. Motwani, and V. R. Narasayya. Overcoming limitations of sampling for aggregation queries. In *ICDE*, pages 534–542, 2001.
4. S. Chaudhuri, G. Das, and V. Narasayya. A robust, optimization-based approach for approximate answering of aggregate queries. In *Proceedings of the 2001 ACM SIGMOD international conference on on Management of data*, 2001.
5. R. R. Coifman, P. W. Jones, and S. Semmes. Two elementary proofs of the l2 boundedness of cauchy integrals on lipschitz curves. *J. Amer. Math. Soc.*, 2(3):553–564, 1989.
6. 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.
7. V. Ganti, M.-L. Lee, and R. Ramakrishnan. Icicles: Self-tuning samples for approximate query answering. *The VLDB Journal*, pages 176–187, 2000.
8. 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.

9. M. Garofalakis and A. Kumar. Deterministic wavelet thresholding for maximum-error metrics. In *Proceedings of the 2004 ACM SIGMOD international conference on on Management of data*, pages 166–176.
10. 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.
11. M. Girardi and W. Sweldens. A new class of unbalanced Haar wavelets that form an unconditional basis for L_p on general measure spaces. *J. Fourier Anal. Appl.*, 3(4), 1997.
12. S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 2nd edition, 1999.
13. 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.
14. Y. Matias and L. Portman. Workload-based wavelet synopses. Technical report, Department of Computer Science,Tel Aviv University, 2003.
15. 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.
16. Y. Matias and D. Urieli. Optimal wavelet synopses for range-sum queries. Technical report, Department of Computer Science, Tel-Aviv University, 2004.
17. Y. Matias and D. Urieli. Optimal workload-based weighted wavelet synopses. Technical report, Department of Computer Science, Tel-Aviv University, 2004.
18. 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.
19. S. Muthukrishnan. Workload-optimal wavelet synopsis. Technical report, May 2004.
20. L. Portman. Workload-based wavelet synopses. Msc thesis, Tel Aviv University, 2003.
21. E. J. Stollnitz, T. D. Derose, and D. H. Salesin. *Wavelets for Computer Graphics*. Morgan Kaufmann, 1996.
22. 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.
23. 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.
24. M. Wang. *Approximation and Learning Techniques in Database Systems*. PhD thesis, Duke University, 1999.

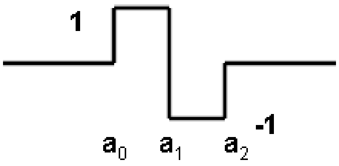

Fig. 1. An example for a Haar basis function

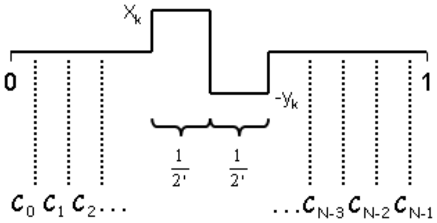

Fig. 2. An example for a Weighted Haar Basis function

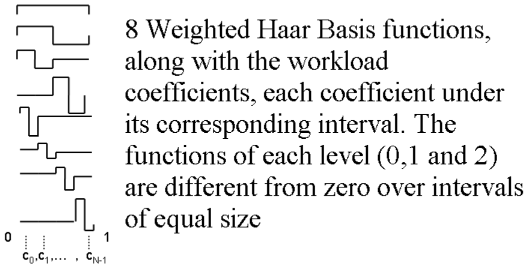
8 Weighted Haar Basis functions, along with the workload coefficients, each coefficient under its corresponding interval. The functions of each level (0,1 and 2) are different from zero over intervals of equal size

Fig. 3. the weighted Haar Basis along with the workload coefficients

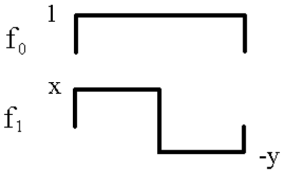

Fig. 4. An example for the Weighted Haar Transform

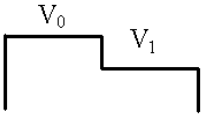

Fig. 5. a simple function with 2 values over $[0, 1)$