DIMACS Technical Report 2004-25
May 2004

Workload-Optimal Wavelet Synposis

by

S. Muthukrishnan[1,2]
Dept. of Computer Science
Rutgers University
New Brunswick, New Jersey 08903

[1]Permanent Member
[2]Supported by NSF ITR 0220280.

DIMACS is a collaborative project of Rutgers University, Princeton University, AT&T Labs–Research, Bell Labs, NEC Laboratories America and Telcordia Technologies, as well as affiliate members Avaya Labs, HP Labs, IBM Research and Microsoft Research. DIMACS was founded as an NSF Science and Technology Center.

ABSTRACT

We present the first known polynomial time for the problem of finding B wavelet vectors to represent a signal of length N so that the representation has the smallest error, averaged over a given workload of the probability of each point in the signal being queried. Previously known results trivially determined the B wavelet vectors with the largest (in absolute value) coefficients; by Parseval's inequality, this is the representation with least error for *uniform* workload where all points are equally likely to be queried. This is no longer true for arbitrary, non-uniform workloads, and our first known polynomial time algorithm makes use of two different kinds of exhaustive searches to determine the workload optimal B-term representation in time $O(N^2B/\log B)$.

1 Introduction

We will present background on wavelets, how to represent functions using wavelets, our problem and our results.

1.1 Background on Wavelets

We first define wavelets. There are many kinds of wavelets. We work with one of the simplest and most popular of them, ie., Haar wavelets.

Definition 1 *Let N be a power of 2. We define $N-1$ proper wavelet functions on $[0,N)$ as follows. For integer j, $0 \leq j < \log(N)$ and integer k, $0 \leq k < 2^j$, we define a* proper wavelet *by*

$$\psi(x)[j,k] = \begin{cases} -\sqrt{2^j/N}, & x \in [kN/2^j, kN/2^j + N/2^{j+1}) \\ +\sqrt{2^j/N}, & x \in [kN/2^j + N/2^{j+1}, (k+1)N/2^j) \\ 0, & \textit{otherwise.} \end{cases}$$

Additionally, we define a wavelet function ϕ, also known as a scaling function*, that takes the value $+1/\sqrt{N}$ over the entire $[0,N)$ linear grid. We will also denote $\phi(x)[j,k]$ as $\psi_{j\times k}$ and ϕ as ψ_{N-1}.*

The *support* of a vector v, denoted $\text{supp}(v)$, is $\{t : v(t) \neq 0\}$. Thus the support of a wavelet vector is either the entire interval $[0,N)$ or, recursively, the left half or right half of the support of some other wavelet (its parent in a naturally-defined binary tree). Each wavelet is constant on the left half and right half of its support. The wavelet ϕ is constant on its support; each other wavelet takes values on its left and right halves that are negatives of each other. The set of possible supports of a wavelet is the set of dyadic intervals of length at least 2. (A *dyadic* interval is the whole space a power of two in length, or recursively, a dyadic interval comprising the left or right half.)

1.2 Wavelet Synopsis for Signals

Wavelets can be used to represent signals. Consider a signal $\mathbf{A}[0...N-1]$, where N is a power of 2 for convenience and $\mathbf{A}[i] \in R$. Any signal $\mathbf{A}$ is exactly recoverable using the wavelet basis, ie.,

$$\mathbf{A} = \sum_i \langle \mathbf{A}, \psi_i \rangle \, \psi_i.$$

Typically, we are not interested in recovering the signal exactly using all the N wavelet coefficients $\langle \mathbf{A}, \psi_i \rangle$; instead, we want to represent the signal using no more than B wavelet coefficients for some $B \ll N$. Say Λ is a set of wavelets of size at most B. Signal $\mathbf{A}$ can be represented as $\mathbf{R}$ using these coefficients as follows:

$$\mathbf{R} = \sum_{i \in \Lambda} \langle \mathbf{A}, \psi_i \rangle \, \psi_i.$$

Necessarily $\mathbf{R}$ is an approximation of $\mathbf{A}$. The *best B-term representation* (aka *wavelet synopsis*) of $\mathbf{A}$ is the choice of Λ that minimizes the error $\epsilon_{\mathbf{R}}$ of representing $\mathbf{A}$ which is typically the sum-squared-error, i.e., $\epsilon_{\mathbf{R}} = ||\mathbf{R} - \mathbf{A}||_2^2 = \sum_i (\mathbf{R}[i] - \mathbf{A}[i])^2$.

A fundamental result is due to Parseval that the best B-term representation for signal $\mathbf{A}$ is $\mathbf{R} = \sum_{i \in \Lambda} c_i \psi_i$, where $c_i = \langle \mathbf{A}, \psi_i \rangle$ and Λ of size k maximizes $\sum_{i \in \Lambda} c_i^2$. This is because the error of $\mathbf{R}$ is

$$\|\mathbf{A} - \mathbf{R}\|^2 = \sum_{i \in \Lambda} (c_i - \langle \mathbf{A}, \psi_i \rangle)^2 + \sum_{i \notin \Lambda} \langle \mathbf{A}, \psi_i \rangle^2 .$$

Hence, choosing the B largest $|c_i|$'s suffices to get the best B term representation for any signal.

1.3 Our Problem

How do wavelet synopses get used in databases and data analysis? A signal $\mathbf{A}$ is represented using a B-term wavelet synopsis. Then its representation $\mathbf{R}$ is used as the (succinct) approximation or the surrogate of $\mathbf{A}$. When the signal is queried using *point queries*, i.e., queries are of the form "$\mathbf{A}[i] =?$", the approximation $\mathbf{R}[i]$ is returned instead. *If all point queries are equally likely*, then average of the squared error of using $\mathbf{R}[i]$ rather than $\mathbf{A}[i]$ is precisely $\epsilon_{\mathbf{R}}$, thus, the rationale to minimize $\epsilon_{\mathbf{R}}$.

Our problem here arises in the same setting. However, in practice, all point queries are *not* equally likely. In general, there is bias in how points are queried that depends on the application and the nature of the signal. So, there is a *workload* π where $\pi[i]$ is the *probability* of i being queried.[1] The uniform workload case is the one when $\pi[i] = 1/N$ for all i. In the general workload case then, one wants to minimize, analogous to the uniform workload case, the total squared-error over all i's, but now normalized for the probability of a point i being queried (equivalently the probability of incurring the error on i). More precisely, we wish to minimize

$$\epsilon_{\mathbf{R}}^{\pi} = \sum_i \pi[i](\mathbf{R}[i] - \mathbf{A}[i])^2.$$

To summarize, our problem is as follows.

Problem 1 *Given signal $\mathbf{A}[0, \ldots, N-1]$, N a power of 2, an integer $B \ll N$, and workload $\pi[0, \ldots, N-1]$, determine set Λ of wavelets such that $|\Lambda| \leq B$, and $\epsilon_{\mathbf{R}}^{\pi} = \sum_i \pi[i](\mathbf{R}[i] - \mathbf{A}[i])^2$ is minimized where $\mathbf{R} = \sum_{i \in \Lambda} \langle \mathbf{A}, \psi_i \rangle \psi_i$.*

It is trivial to observe that the B i's with the largest $|c_i|$'s no longer form the best B-term representation and indeed can be arbitrarily far away from the optimal in specific instances. So, algorithmically the problem is nontrivial.

This problem is fairly central to query optimization in databases. Databases maintain workload information by profiling queries in a number of ways and would prefer to use

[1] One may use any nonnegative weights for each of the $\pi[i]$'s; for the results in this paper, it does not matter if they are normalized to 1.

wavelet synopsis that accurately reflects their cost for the given workload rather than ignore this information. In database literature, there has been a lot of work related to this problem. In [1, 4], the general concept of synopses that adapt to the nonuniform workload is studied; the synopses the authors study there are histograms and samples respectively, but the issue is equally important with wavelet synopses since they have been shown to be powerful for database applications in [7]. In [8], authors adapted wavelet based estimation to different workloads, however, they studied the order of retrieval of the full set of wavelet coefficients in order to be efficient over a whole set of queries but provide full reconstruction of a signal; they did not study the problem of finding the best B-term representation. In [2], authors studied how to obtain B-term representation for a set of signals with cumulative error bound.

Our problem is also interesting from a Mathematical Approximation Theory [3] point of view where researchers focus on how accurately functions (here, signals) can be reconstructed using different dictionaries (here, Haar wavelet basis). The precise problem we study is that of reconstructing functions where the reconstruction of individual points have different importance specified by the workload. Of particular interest to this community may be an extremal version of our problem where we only want to reconstruct the inner product of the signal with say k specified spikes. This would correspond to setting the k corresponding positions in the workload to be 1, the remainder being 0, and normalized appropriately.

No algorithmic results are previously known for our problem.

1.4 Our Result

Our work here germinated in the context of citesurajit: can one design workload-optimal synopses? In [1], the authors considered histogram synopses. It is trivial to extend the basic dynamic programming algorithm in [6] to get workload-optimal B-term histogram in time $O(N^2B)$, in same time as it takes to get optimal B-term histogram for the uniform workload case. The advantage of wavelets is that the optimal B-term representation can be obtained very efficiently (in $O(N)$ time) for the uniform load case. Does this advantage extend to the nonuniform workload case, i.e., *can we find workload optimal B-term wavelet synopsis in $o(N^2B)$ time?* Indeed the larger question that is posed is can we even find the workload optimal B-term wavelet synopsis in polynomial time? The straightforward approach is to consider dynamic programming. While it works for finding (workload) optimal histograms, it does not directly work for finding workload optimal wavelet synopsis. This is because the problem does not subdivide into disjoint parts; any two partition of the problem may have upto $\log N$ wavelet vectors that straddle the partition and affect both the subproblems, and hence, one may incur $N^{O\log N}$ type bound trivially.

Our result here is the *first known* polynomial time algorithm that finds a workload-optimal B-term wavelet synopsis. More precisely, our algorithm takes time $O(N^2B/\log B)$ to find the provably workload-optimal B-term wavelet synopses (this is $o(N^2B)$, but barely, so that may not be noteworthy).

Our algorithm relies on exhaustive search over the space of all wavelet vectors. There will be two kinds of exhaustive searches.

1. *(Internal Search)* For any given dyadic interval $I \in [0, N-1]$, the first exhaustive search will search all combinations of wavelet vectors whose support is *contained in* I. Unfortunately such vectors are rather large in number: any interval I has $O(I)$ wavelet vectors whose support is contained in I, and therefore there are $2^{O(1)}$ possible combinations of wavelet vectors to consider. So, we will be able to do this exhaustive search only for short I's.

2. *(External Search)* For long I's, we will be able to do exhaustive search over all the wavelet vectors whose support *contains* I. Luckily, the wavelets have the property that only atmost $\log N$ of them have supports that contains I for any I. Hence, we will be able to search their combinations exhaustively.

Using only external search above will still give a workload-optimal wavelet synopses but will take time $O(N^2 B \log B)$, so using the internal search additionally does improve the running time. Using the internal search by itself will not give a polynomial time algorithm.

2 The Result

We will first describe the algorithms for the internal and external searches for any given interval I. Then we will combine these into the complete algorithm for our problem.

2.1 Internal Search

Let us define the problem more precisely. Consider an interval I. We will have a contribution for all $i \in I$ from wavelets S that are chosen in the external search problem from those whose support contains I; so, S is an input parameter to the internal search problem. Our goal is to populate a table $IE(I, k, S)$ which is the least workload-based error for representing $\mathbf{A}[I]$ with any subset Λ_I of wavelets that have supports contained in I, $|\Lambda_I| \leq k$, together with the contribution for the set S of wavelets whose individual support strictly contain I. More formally, the contribution of S, $|S| \leq k$, to a $j \in I$ is denoted

$$v[j] = (\sum_{\psi_i \in S} \langle \mathbf{A}, \psi_i \rangle \, \psi_i)[j]$$

. Since all ψ_i's in S have support that contains I, $v[j]$'s will be identical for all $j \in I$. So, we have a generic v to denote the value $v[j]$ for any $j \in I$. Now,

$$IE(I, k, S) = \min_{|\Lambda_I| \leq k; \text{supp}(\psi_i \in \Lambda_I) \subset I} \sum_{j \in I} \pi[j] \left((\sum_{\psi_i \in \Lambda_I} \langle \mathbf{A}, \psi_i \rangle \, \psi_i)[j] + v - \mathbf{A}[j] \right)^2 .$$

This can be calculated in $O(1)$ time for each new $\psi_i \in \Lambda_I$ by considering them in the bottom order of the wavelet supports. So, the total time taken for populating this table is $O(2^I)$ for each S in the worst case.

2.2 External Search

Here too, given an interval I, we will focus on computing $IE(I, k, S)$. However, here, our emphasis on searching over the different sets S that impinge on I, and hence, our approach will be different. More specifically, we will now use dynamic programming to search over all possible S's that impinge on I.

Each dyadic interval I has a left half denoted I_L and a right half denoted I_R which are themselves dyadic intervals. With each interval I, we associate the wavelet vector ψ_I that is positive on I_L and is negative on I_R. There are two cases to consider while computing $IE(I, k, S)$: ψ_I is included in the set of k wavelet terms or not. Either way, we will write $IE(I, k, S)$ in terms of similar functions for I_R and I_L recursively.

Say ψ_I is included in the set of k wavelet terms. Then,

$$IE(I, k, S) = \min_{0 \le k' \le k-1} IE(I_L, k', S \cup \psi_I) + IE(I_R, k - 1 - k', S \cup \psi_I).$$

In the other case,

$$IE(I, k, S) = \min_{0 \le k' \le k} IE(I_L, k', S) + IE(I_R, k - k', S).$$

Hence, $IE(I, k, S)$ is the minimum of the two possibilities above. The boundary cases are easy to define, and $IE(I, k, S)$ can be calculated using dynamic programming. Each such term takes time $O(\log k)$ to compute using the monotonicity of the terms on the right as k' increases. Clearly $IE([1, N], B, \phi)$ solves the entire problem (where ϕ is the empty set).

2.3 Putting Things Together

The final algorithm involves doing the external search approach for large intervals and finishing it off with internal search for short intervals.

Say we do external search for all dyadic intervals of size strictly greater than 2^j, and internal search for the intervals of size precisely 2^j (called *short*).

The number of short intervals is at most $N/2^j$. Each such interval I has $\log N - j$ wavelet vectors whose support strictly contains I. Hence the set of possible sets S that are involved in computing $IE(I, k, S)$ is at most $2^{\log N - j}$. Each term takes $O(2^{2^j})$ time to compute. Hence, the total cost is

$$O(\frac{N}{2^j} 2^{\log N - j} 2^{2^j}).$$

For longer intervals, the total cost is dominated by the intervals of smallest size, i.e., 2^{j+1}, but we will perform the computations as if the smallest size interval is size 2^j for convenience. The total cost here is

$$O(\frac{N}{2^j} 2^{\log N - j} B \log B).$$

Now we will balance the two costs to choose j. When the two costs are equal, we have $2^{2^j} = B \log B$, so we choose $j = \log \log B$. Then, the total cost is $O((\frac{N}{2^j})^2 B \log B) =$

$O(N^2B/\log B)$. From the optimal cost, one can easily extract the actual choice of wavelets that will give the optimal cost using standard techniques in dynamic programming applications.

Theorem 2 *There exists an $O(N^2B/\log B)$ time algorithm to calcuate workload-optimal B-term wavelet representation for a signal of length N.*

The proof of correctness follows from the following observation.

Lemma 3 *Let S be any set of wavelet vectors and $E[I,S]$ the error of representing the dyadic interval I of signal $\mathbf{A}$ using S, that is,*

$$E(I,S) = \sum_{j\in I} \pi[j] \left((\sum_{\psi_i \in S} \langle \mathbf{A}, \psi_i \rangle \, \psi_i))[j] - \mathbf{A}[j] \right)^2 .$$

Further let $I = I_1 \cup I_2 \cup \cdots I_\ell$ for some ℓ, where I_i's are dyadic intervals with any two intervals I_i and I_j being disjoint. Also, let S_i denote those wavelets in i whose support intersects I_i, i.e., $S_i = \{\psi_j \mid \mathrm{supp}(\psi_j) \in I_i \text{ or } I_i \in \mathrm{supp}(\psi_j)\}$. Then,

$$E[I,S] = E[I_1,S_1] + E[I_2,S_2] + \cdots + E[I_\ell,S_\ell].$$

3 Extensions

Recently, Garofalakis and Kumar [5] studied the problem of choosing B-term wavelet synopsis for a signal that minimizes the *maximum* relative error in estimating a signal point (rather the sum of squares of the error), *under uniform workload*. They presented a polynomial time algorithm for their problem and the approach is based on external exhaustive search. Using our approach of coupling the external and internal searches will improve the time bounds in [5] by a $\log^2 B$ term. Also, one can imagine minimizing the expected maximum relative error under the given probabilistic workload, and our algorithm here will work for that measure too.

A problem related to the one studied in [5] is its dual defined as follows.

Problem 2 *Given a bound δ on the maximum absolute error on estimating a signal point, minimize the number B of wavelet synopsis terms needed to represent a signal. The workload is assumed to be uniform.*

Given an interval I and set S of wavelets whose individual supports strictly contain I, consider $B(I,\delta,S)$, the least number of wavelets from Λ_I needed for representing $\mathbf{A}[I]$ such that for each $i \in I$, $|\mathbf{R}[i] - a[i]| \le \delta$.[2] Say ψ_I is included in the set of k wavelet terms. Then,

$$B(I,\delta,S) = B(I_L,\delta,S \cup \psi_I) + B(I_R,\delta,S \cup \psi_I) + 1.$$

[2]The results that ensue can be applied to the relative error definition as well.

In the other case,

$$B(I, \delta, S) = B(I_L, \delta, S) + B(I_R, \delta, S).$$

Hence, $B(I, \delta, S)$ is the minimum of the two possibilities above. This external search method gives an algorithm that takes time $O(N^2)$.

We can improve this result with an additional observation and using internal search. The observation is that we do not have to solve $B(I, \delta, S)$ for each possible S. Instead, *it suffices to keep the maximum and minimum absolute errors for* each *size ℓ of the subset of wavelets in Λ_I.* This is because

$$\min_{S\,, i \in 2^{\Lambda_I}} \max_{j \in I} |\delta_i^j - v_S|,$$

where δ_i^j is the error in $\mathbf{A}[j]$ due to set i of wavelets chosen from Λ_I and v_S is the contribution of S to each $i \in I$, is equal to

$$\min_{S\,, i \in 2^{\Lambda_I}} \max\{|(\max_{j \in I} \delta_i^j) - v_S|, |(\min_{j \in I} \delta_i^j) - v_S|\}.$$

As before, we run external search until we have intervals of size 2^j and following that, we run the internal search; the combined algorithm will take time $O(\frac{N}{2^j}\frac{N}{2^j}2^j + \frac{N}{2^j}2^{2^j})$, which, when balanced, reduces to $O(N^2/logN)$. This result for the dual problem is not only of independent interest, but also may be used to approximate the solution to the primal problem studied in [5] by binary searching with guesses for δ, which may be more efficient in certain cases.

Finally, we can solve our problem for other notions of error in representing a signal, say L_1 norm of the error, using the internal and external search algorithm.

4 Concluding Remarks

It is a challenge to solve our problem dynamically as the workload changes under updates and we need to design a streaming algorithm to process such a stream of updates to both the workload as well as the signal, and still maintain an *approximate* B-term wavelet represenation of the current signal under the current workload.

Finally, under realistic non-uniform workloads, it will be interesting to compare the performance of workload-optimal histograms vs wavelets using experiments. Likewise, it will be interesting to evaluate different heuristics for workload-optimal wavelets using experiments.

References

[1] A. Aboulnaga and S. Chaudhuri. Self-tuning histograms: building histograms without looking at data. *Proc. SIGMOD*, 181–192, 1999.

[2] A. Deligiannakis and N. Roussopoulos. Extended wavelets for multiple measures. *Proc. SIGMOD*, 2003.

[3] R. Devore and G. Lorentz. *Constructive approximation.* Springer Verlag, 1991.

[4] V. Ganti, M. Lee and R. Ramakrishnan. ICICLES: Self-tuning samples for approximate query answering. *VLDB Journal*, 176–187, 2000.

[5] M. Garofalakis and A. Kumar. Deterministic wavelet thresholding for maximum error metrics. *Proc. PODS*, 2004.

[6] H. Jagadish, N. Koudas, S. Muthukrishnan, V. Poosala, K. Sevcik, and T. Suel. Optimal Histograms with Quality Guarantees. *Proc. VLDB*, 1998, 275–286.

[7] Y. Matias, J. Vitter and M. Wang. Wavelet-based histograms for selectivity estimation. *Proc. ACM SIGMOD*, 448–459, 1998.

[8] R. Schmidt and C. Shahabi. How to evaluate multiple range-sum queries progessively. *Proc. PODS*, 2002.