

Approximation Algorithms for Wavelet Transform Coding of Data Streams

Sudipto Guha*

Boulos Harb*

Abstract

Given a set of orthonormal basis functions $\{\psi_i\}$ and a target function/vector f , the wavelet representation problem is to construct \hat{f} as a combination of at most B basis vectors to minimize some normed distance between f and \hat{f} . The problem is well understood if the error is the mean squared error: The largest (ignoring signs) B coefficients of the wavelet expansion should be retained. This strategy follows from the proof of optimality and is not a built-in constraint.

The mean squared error, however, is not the optimization criterion in several scenarios. The above easy solution to the wavelet representation problem does not carry over to ℓ_p for $p \neq 2$, and it turns out that restricting the solution to any subset of coefficients of size B or less is suboptimal compared to the best solution which can choose arbitrary real numbers. Further, all the previous literature on non- ℓ_2 errors only considered the Haar system.

In this paper we provide the first approximation schemes for the unrestricted optimization problem. We provide a lower bounding technique based on a system of inequalities. We show that a modified greedy algorithm that retains the coefficients of expansion gives a $O(\log n)$ true (factor) approximation algorithm for a wide variety of compact wavelet systems, including Haar, Daubechies, Symmlets, Coiflets among others. This vindicates several scaling type algorithms which are used in practice. We subsequently augment the lower bound and give a FPTAS for the Haar system. The same ideas extend to a QPTAS for the more general class of compact wavelets mentioned above. We also consider adaptive quantization problems, which are generalizations of the B -term representations.

1 Introduction

Given a set of orthonormal basis functions $\{\psi_i\}$ and a target function/vector f , the wavelet representation problem is to construct \hat{f} as a combination of at most B basis vectors to minimize some normed distance between f and \hat{f} . This is one of the most studied problems in the area of Non-Linear Approximation theory which has a rich history (e.g. see the survey by DeVore [4]). More recently, wavelets and multi-fractals have been found extremely useful in image and data compression [18, 3], hence the name transform coding.

Observe that the problem does not restrict how the B vectors are to be combined—we are supposed to come up with a solution $\{z_i\}$ with at most B non-zero entries and $\hat{f} = \sum_i z_i \psi_i$.

The problem is well understood if the error is the mean squared error or ℓ_2 distance. Since the ℓ_2 distance is preserved in an orthonormal transformation, by Parseval's theorem, we can perform a change of basis and $\|f - \hat{f}\|_2^2 = \sum_i (z_i - \langle f, \psi_i \rangle)^2$. It is then clear that the solution in this case is to retain the largest B inner products $\langle f, \psi_i \rangle$, which are also the coefficients of the wavelet expansion of f . *The fact that we have to store the inner products or the coefficients is a natural consequence of the proof of optimality.*

However many optimization problems that arise in data compression [21, 6] and image analysis [16] need to minimize non- ℓ_2 distances. In fact Mallat [18, p. 528] observes that in transform coding although the ℓ_2 distance does not adequately quantify perceptual errors, it is nonetheless used since other norms are difficult to optimize. Further, given the increasing use of wavelets in the representation of time series, synopses, etc., [2], minimizing general (weighted) ℓ_p norms is a natural optimization problem that remains open.

The B -term representation considers storing $2B$ numbers, the index and the value. A natural generalization arises from the observation that the actual cost (in bits) of storing the real numbers z_i is non-uniform. Depending on the scenario, it may be beneficial to represent a function with a large number of low support vectors with low precision z_i 's or a few vectors with

*Department of Computer Information Science, University of Pennsylvania, 3330 Walnut St, Philadelphia, PA 19104. **Email:** {sudipto,boulos}@cis.upenn.edu. This research was supported in part by an Alfred P. Sloan Research Fellowship and by an NSF Award CCF-0430376.

more detailed precision z_i 's. This is the bit-budget version of the problem known as the *Adaptive Quantization* problem which we also consider in this paper. There are no known approximation algorithms for this problem. Observe that the bit-budget version is naturally not constrained to storing inner products.

The common strategy for the original B -term representation problem in the literature has been “to retain the $[B]$ terms in the wavelet expansion of the target function which are largest relative to the norm in which error of approximation is to be measured” [4, p. 4]. This strategy is reasonable in an extremal setting, i.e., if we are measuring the rate of the error as a function of B , there exist functions which match the worst case rate. But from an optimization point of view, given a particular f , the common strategy implies no guarantee of approximation. In fact, in a recent work [11], we showed that retaining B or less coefficients is sub-optimal by at least a factor ~ 1.2 compared to the optimal solution that can choose any numbers for the z_i 's. We showed a counterexample in the Haar setting that holds for general wavelets as well. We also showed that rounding the range $[-M, M]$ to multiples of $\epsilon M / \log n$, where M is the maximum value in the input, gives an additive error of ϵM . The following, however, remains unresolved:

Question *What is the true (factor) approximability of the B -term wavelet representation problem when the optimal solution can store any numbers and is not restricted to choosing only from the wavelet coefficients?*

Since most algorithms used in practice retain the wavelet coefficients, what is the approximation guarantee of retaining those coefficients? Further, is there a natural proof technique which suggests that retaining the coefficients of the expansion is a reasonable strategy?

It is noteworthy that several researchers [6, 5, 22] observe that the common strategy of retaining the largest coefficients relative to the error norm is suboptimal. *However, all works addressing this issue prior to this paper only consider the Haar system.*

Gibbons and Garofalakis [6] do store numbers other than coefficients. They propose a probabilistic framework and compare their solution to solutions which retain coefficients of the expansion. However, compared to the unrestricted optimum B -term synopsis, the quality of their solution remains unclear.

Under the restriction that the algorithm stores the wavelet coefficients only, Garofalakis and Kumar [5] gave an optimal algorithm for weighted ℓ_∞ norm. Under the same restriction, improvements were observed by Muthukrishnan [22], Matias and Urieli [19], and Guha [10].

In [20], Matias and Urieli show that for the Haar system and the weighted ℓ_2 norm, if the weights are known in advance, then we can “re-weight” the Haar basis to design a new basis that mimics the ℓ_2 behavior. They mention ℓ_∞ as an open problem. In essence, using ideas in this paper, their proofs can be viewed as showing that any weighted ℓ_p problem can be made to mimic ℓ_p behavior. We will assume the more common optimization model where the basis is fixed.

The result of Gilbert, Muthukrishnan and Strauss [9] who study general redundant basis sets with small coherence but use ℓ_2 distances and store the expansion coefficients is worth noting in this context. This was one of the early papers investigating non-extremal results in wavelet theory. Their proof techniques are based on projections and appear unsuitable to (easily) extend to non- ℓ_2 distances. Their problem is somewhat orthogonal to the representation question we are asking here.

From all the previous work, [6, 5, 22, 19, 10, 20, 11] several shortcomings become apparent:

- There are no analysis techniques for ℓ_p norms. In fact this is the bottleneck in analyzing any generalization of the B -term representation problem, e.g., the adaptive quantization.
- All of the (limited) analyses in the optimization setting have been done on the Haar system, which although important, is not the wavelet of choice in many applications.
- There is no analysis of the greedy algorithms used in practice. They are indeed suboptimal strategies, but the bounds on their performance are unclear. This relates to the lack of understanding regarding the question: is retaining the coefficients natural from an approximation point of view?

Our Results: We address the shortcomings above, more precisely,

- For the B -term representation problem we show that,
 - The unrestricted optimization problem has a PTAS for all ℓ_p distances in the Haar system.

The algorithm is one pass sublinear space (therefore streaming) for ℓ_p distances with $p \geq 3$. For ℓ_∞ the algorithm is polylog space and $\tilde{O}(n)$ time. The proof is based on a new bound that involves the wavelet scaling vectors $\{\phi_i\}$ (instead of only the wavelet basis vectors $\{\psi_i\}$) which in fact allows us to improve upon the running time of the additive approximation algorithm of [11] as well.

The results extend to fixed dimensions with appropriate increase in running time and space. For general compact systems we can show a QPTAS.

- The restricted solution that retains at most B wavelet coefficients is a $O(n^{1/p} \log n)$ approximation to the unrestricted solution for all ℓ_p distances for general compact systems (e.g. Haar, Daubechies, Symmlets, Coiflets, etc.)*. We provide a modified greedy strategy, which is not normalization, but is similar to some scaling strategies used in practice. This vindicates why several scaling based algorithms used in practice work well.
- In terms of techniques, we introduce a new lower bounding technique using the basis vectors $\{\psi_i\}$, which gives us the result involving the gap between the restricted and unrestricted versions. We also show that bounds using the scaling vectors $\{\phi_i\}$ are useful for these optimization problems and, along with the lower bounds using $\{\psi_i\}$, give us the approximation schemes. To the best of our knowledge, this is the first use of the scaling functions as well as the basis vectors to achieve such guarantees.
- We show that the lower bound for general compact systems can be extended to an approximation algorithm for adaptive quantization. This is the first approximation algorithm for this problem.

Our algorithms extend to weighted cases/workloads under the same assumptions as in [20]; namely, $\sum_{i=1}^n w_i = 1$ where $0 < w_i \leq 1$. However, the running times become dependent on the ratio $\max_i w_i / \min_i w_i$.

Organization: We begin by reviewing some preliminaries of compact wavelets. We then focus on the result regarding the restricted vs. the unrestricted versions of the problem. We subsequently investigate the approximation schemes in Section 4. Finally, we consider adaptive quantization in Section 5.

2 Preliminaries

For the purpose of this paper, a data stream computation is a space bounded algorithm, where the space is sublinear in the input. Input items are accessed sequentially and any item not explicitly stored cannot be accessed again in the same pass. In this paper we focus on *one pass* data streams. We will assume that we are given numbers $f = f(1), \dots, f(i), \dots, f(n)$ which correspond

*This statement is not similar to the statement in the extremal setting which says that discarding all coefficients below τ introduces $O(\tau \log n)$ error, since the latter does not account for the number of terms.

to the signal f to be summarized in the increasing order of i . This model is often referred to as the *aggregated model* and has been used widely [14, 8, 12]. It is specially suited to model streams of time series data [17, 1] and is natural for transcoding a single channel. We will focus on dyadic wavelets (that are down-sampled by 2) and assuming n to be a power of 2 will be convenient (but not necessary). As is standard in streaming [13, 7, 15], we assume that the numbers are polynomially bounded.

2.1 Compactly Supported Wavelets We include some of the basic concepts of wavelets, since although many readers are familiar with the Haar system, the non-Haar systems are slightly more involved. Readers familiar with wavelets can easily skip this section. We start with the definition of a compactly supported function.

Notation: The δ_{ik} is the standard Kronecker δ , i.e., $\delta_{ik} = 1$ if $i = k$ and 0 otherwise. All unconstrained sums and integrals are from $-\infty$ to $+\infty$.

DEFINITION 2.1. (COMPACT SUPPORT) *A function $f : \mathbb{R} \rightarrow \mathbb{R}$ has compact support if there is a closed interval $I = [a, b]$ such that $f(x) = 0$ for all $x \notin I$.*

Wavelets provide a special kind of basis where all basis functions $\{\psi_i(x)\}_{i \in [n]}$ are derived from a single function $\psi(t)$ called the *mother* wavelet. The mother wavelet is related to a scaling function defined below.

DEFINITION 2.2. (WAVELET SCALING FUNCTION) *The wavelet scaling function ϕ is defined by $\frac{1}{\sqrt{2}}\phi\left(\frac{x}{2}\right) = \sum_k h[k]\phi(x - k)$ where $h[k] = \left\langle \frac{1}{\sqrt{2}}\phi\left(\frac{x}{2}\right), \phi(x - k) \right\rangle$.*

The sequence $h[k]$ is interpreted as a discrete filter. Several admissibility conditions apply including $\sum_k h[k] = \sqrt{2}$ and $\sum_k (-1)^k h[k] = 0$, see [18, 3]. *In this paper we will only focus on wavelets whose scaling function is continuous.*

DEFINITION 2.3. (WAVELET FUNCTION) *The wavelet function ψ , also called the mother wavelet, is defined by $\frac{1}{\sqrt{2}}\psi\left(\frac{x}{2}\right) = \sum_k g[k]\phi(x - k)$ where $g[k] = \left\langle \frac{1}{\sqrt{2}}\psi\left(\frac{x}{2}\right), \phi(x - k) \right\rangle$.*

It can be shown that the components of g and h are related by the *mirror* relationship $g[k] = (-1)^k h[1 - k]$. Thus $\sum_k g[k] = 0$. The admissibility conditions on h allow ϕ, ψ to converge. It is shown in [18, 3] that the function ϕ has a compact support if and only if h has a compact support and their supports are equal. The support of ψ has the same length, but it is shifted.

DEFINITION 2.4. Let $\phi_{0,s}$ be defined as $\phi_{0,s}(t) = \delta_{st}$, i.e., the characteristic vector which is 1 at s and 0 everywhere else. Define $\phi_{j+1,s} = \sum_t h[t-2s]\phi_{j,t}$ and $\psi_{j+1,s} = \sum_t g[t-2s]\phi_{j,t}$.

PROPOSITION 2.1. For a compactly supported wavelet whose filter has $2q$ non-zero coefficients there are $O(q \log n)$ basis vectors with a non-zero value at any point t .

PROPOSITION 2.2. Given t , $\psi_{j,s}(t)$ and $\phi_{j,s}(t)$ can be computed in $O(q \log n)$ time.

PROPOSITION 2.3. ([18, THM. 7.7]) $\phi_{j,s} = \sum_t h[s-2t]\phi_{j+1,t} + \sum_t g[s-2t]\psi_{j+1,t}$. Further $\phi_{j,s}(x)$ converges to $2^{-j/2}\phi\left(\frac{x-2^j s}{2^j}\right)$.

The set of wavelet vectors $\{\psi_{j,s}\}_{(j,s) \in \mathbb{Z}^2}$ define an orthonormal basis of the space of functions F with finite energy $\int |F(t)|^2 dt < +\infty$ [18, Thm. 7.3]. The function $\psi_{j,s}$ is said to be centered at $2^j s$ and of scale j and is defined on at most $(2q-1)2^j$ points. For ease of notation, we will use both ψ_i and $\psi_{j,s}$ depending on the context and assume there is a consistent map between them.

The Cascade algorithm to compute $\langle f, \psi_{j,s} \rangle, \langle f, \phi_{j,s} \rangle$. Assume that we have the filter h with support $\{0, \dots, 2q-1\}$. Given a function f , set $a_0[i] = f(i)$, and repeatedly compute $a_{j+1}[t] = \sum_s h[s-2t]a_j[s]$ and $d_{j+1}[t] = \sum_s g[s-2t]a_j[s]$. Observe that for compact systems this forces $0 \leq s-2t \leq 2q-1$. It is easy to see that $a_j[t] = \langle f, \phi_{j,t} \rangle$ and $d_j[t] = \langle f, \psi_{j,t} \rangle$.

To compute the inverse transform, $a_j[t] = \sum_s h[t-2s]a_{j+1}[s] + \sum_s g[t-2s]d_{j+1}[s]$. Observe that setting a single $a_j[s]$ or $d_j[s]$ to 1 and the rest to 0, the inverse transform gives us $\phi_{j,s}$ or $\psi_{j,s}$; this is the algorithm to compute $\phi_{j,s}(t), \psi_{j,s}(t)$.

Example I. Haar Wavelets: In this case $q = 1$ and $h[] = \{1/\sqrt{2}, 1/\sqrt{2}\}$. Thus $g[] = \{1/\sqrt{2}, -1/\sqrt{2}\}$. The algorithm to compute the transform computes the ‘‘difference’’ coefficients $d_1[i] = (f(2i) - f(2i+1))/\sqrt{2}$. The ‘‘averages’’ $(f(2i) + f(2i+1))/\sqrt{2}$, corresponds to $a_1[i]$, and the entire process is repeated on these $a_1[i]$ but with $n := n/2$ since we have halved the number of values. In the inverse transform we get for example $a_0[0] = (a_1[0] + d_1[0])/\sqrt{2} = ((f(0) + f(1))/\sqrt{2} + (f(0) - f(1))/\sqrt{2})/\sqrt{2} = f(0)$ as expected. The coefficients naturally define a coefficient tree where the root is $a_{\log n+1}[0]$ (the overall average scaled by \sqrt{n}) with a single child $d_{\log n}[0]$ (the scaled differences of the averages of the left and right halves). Underneath $d_{\log n}[0]$ lies a complete binary tree. The total number

of nodes is $1 + 2^{\log n} - 1 = n$. The basis is:

$$\begin{aligned} \psi_1 &= \{1/\sqrt{n}, 1/\sqrt{n}, \dots, 1/\sqrt{n}\}, \dots, \\ \psi_{n/4+1} &= \{1/2, 1/2, -1/2, -1/2, 0, \dots\}, \\ \psi_{n/4+2} &= \{0, 0, 0, 0, 1/2, 1/2, -1/2, -1/2, 0, \dots\}, \dots, \\ \psi_{n/2+1} &= \left\{\frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}, 0, \dots\right\}, \\ \psi_{n/2+2} &= \left\{0, 0, \frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}, 0, \dots\right\}, \\ \psi_{n/2+3} &= \left\{0, 0, 0, 0, \frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}, 0, \dots\right\}, \\ \psi_{n/2+4} &= \left\{0, 0, 0, 0, 0, 0, \frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}, 0, \dots\right\}, \dots \end{aligned}$$

The scaling function is $\phi(x) = 1$ for $0 \leq x \leq 1$ and 0 otherwise, and the mother wavelet is $\psi(t) = 1$ if $0 \leq t < 1/2$, $\psi(t) = -1$ for $1/2 \leq t < 1$ and 0 otherwise. Note that ψ is discontinuous. Thus the synopses using Haar wavelets are better suited to handle ‘‘jumps’’ or discontinuities in data. This simple wavelet proposed in 1910 is still useful in a wide number of database applications [21, 23] since it is excellent in concentrating the *energy* of the transformed signal (sum of squares of coefficients). However, the energy of a signal is not the sole aim of transforms. A natural question to raise is whether there are ‘smooth’ wavelets, and the seminal work of Daubechies gives us several examples [3].

Example II. Daubechies Wavelets D_2 : In this case $q = 2$ and $h[] = \left\{\frac{1+\sqrt{3}}{4\sqrt{2}}, \frac{3+\sqrt{3}}{4\sqrt{2}}, \frac{3-\sqrt{3}}{4\sqrt{2}}, \frac{1-\sqrt{3}}{4\sqrt{2}}\right\}$. Thus $g[] = \{h[3], -h[2], h[1], -h[0]\}$. The ϕ and the ψ functions are shown below (normalized to the domain $[0, 1]$) and they converge quite rapidly. The coefficients now form a graph rather than a tree, which is given in Figure 1. The D_q wavelets have compact support (q is a fixed integer) but are unfortunately asymmetric. It turns out that Haar wavelets are the unique real symmetric compactly supported wavelets [3]. To ameliorate the symmetry (and other) issues in the real domain several proposals have been made (e.g. Symmlets). Our results are relevant as long as the wavelets are compactly supported.

3 Analysis of a Greedy Algorithm

Recall our optimization problem is: Given a set of basis functions $\{\psi_i\}$ and a target function f specified as a vector, we wish to find $\{z_i\}$ with at most B non-zero numbers to minimize $\|f - \sum_i z_i \psi_i\|_p$. We begin by analyzing the sufficient conditions that guarantee the error. A (modified) greedy coefficient retention algorithm will naturally fall out of the analysis. The main lemma is:

LEMMA 3.1. Let \mathcal{E} be the minimum error under the ℓ_p

We conclude with the following:

THEOREM 3.1. *The above algorithm of choosing the B coefficients with largest $|\langle f, \psi_i \rangle| / \|\psi_i\|_1$ is a $O(q^{3/2}n^{1/p} \log n)$ approximation for the unrestricted optimization problem under the ℓ_p norm.*

Observe that the above naturally extends to arbitrary dimensions.

4 (1 + ϵ) Approximations

In this section we will provide a PTAS for the Haar system and a QPTAS for the general compact wavelet case. A natural question in our mind will be: We have a new lower bound and we already had an additive approximation algorithm for the Haar system, do we get a PTAS from just putting together those ideas? The answer is no. But we come close, namely we get a solution which blows up the space bound. But in order to clarify the difficulty in getting a PTAS, we will further explore this idea, which will also establish why the new techniques in this paper were necessary.

4.1 Limits of using the Additive Approximation

Let us revisit the additive approximation in the Haar case [11]. Suppose we were promised that in the optimum solution the first half of the data uses b coefficients. Imagine we are at a point where we have seen the first half of the data. We cannot make any choices because the second half may force the top-level wavelet coefficient to be chosen or not chosen. Thus we have to build into the dynamic program the *combination* of all the choices of the ancestor nodes in the Haar coefficient tree. We can therefore set up a table $\text{ERR}[i, b, v]$ at each node i of the tree where v is the result of all the choices of the ancestors.

The dynamic program is simple, if the left and right children are i_L, i_R and we do not choose to store a coefficient we need to compute $\min_{b'} \text{ERR}[i_L, b', v] + \text{ERR}[i_R, b - b', v]$ (assuming we are evaluating $\|\cdot\|_p^p$, possibly weighted as well). If we do decide to store a coefficient, say z_i , then assuming appropriate scaling the left child will be seeing $v + z_i$ and the right child will be seeing $v - z_i$. Therefore, we need to compute $\min_{z_i} \min_{b'} \text{ERR}[i_L, b', v + z_i] + \text{ERR}[i_R, b - b' - 1, v - z_i]$. For the ℓ_∞ case both sums are replaced by \max . The next proposition is a generalization of wavelet thresholding.

PROPOSITION 4.1. *Rounding the optimum $\{z_i^*\}$ to the nearest multiple of ρ , gives us a solution of cost $\mathcal{E} + O(n^{1/p} \rho \log n)$.*

Now the contribution of [11] was to show that $|v|$ is a range which is at most $O(Mn^{1/p})$ for any optimum

solution, if the largest absolute number seen is M . Thus the table at each node was of size $O(B|R|)$, with $|R| = Mn^{1/p}/\rho$ and the running time was $O(nB|R|^2)$; the factor B is replaced by $\log^2 B$ for ℓ_∞ .

Observe that even an estimate of \mathcal{E} does not help us since the bottleneck is that the dynamic program has to be done in the original space (as opposed to the transformed space). Thus ρ must relate to M or otherwise the time and space bounds blow up.

Interestingly, we can combine the bound in the previous section and the additive approximation algorithm. Consider an f' which is same as f except with the largest (in our $\|\cdot\|_1$ scaled order) B coefficients set to 0. Then there is a solution of $2B$ vectors which approximates f' up to error \mathcal{E} . We can show using lemmas 3.1 and 3.2 that the largest number in f' is bounded by $q\mathcal{E}\sqrt{n} \log n$. If we set $\rho = \epsilon\mathcal{E}/(qn^{1/p} \log n)$, we get an approximation for f' with at most $2B$ vectors with error at most $(1 + \epsilon)\mathcal{E}$. We can now “put back” the coefficients of f and get an approximation with at most $3B$ vectors and error $(1 + \epsilon)\mathcal{E}$.

The discussion above requires proof but since we will improve the result, we omit the details from this version. However, the above makes it clear that the result cannot be improved by the techniques of the additive approximation alone. We need stronger guarantees; in particular, on the range of v . In what follows we show how to achieve such a result while ensuring that we use at most B coefficients. For the Haar case we also improve the running time by a $O(n^{2/p})$ factor compared to the additive approximation algorithm.

4.2 There and Back Again The next idea comes from dual wavelet bases; i.e., where we use one basis to construct the coefficients and another to reconstruct the function. Our bases will differ by scaling factors.

We will solve the problem in the scaled bases and translate the solution to the original basis.

DEFINITION 4.1. *Define $\psi_{j,s}^a = 2^{-j/2}\psi_{j,s}$ and $\psi_{j,s}^b = 2^{j/2}\psi_{j,s}$. Likewise define ϕ_i^a, ϕ_i^b .*

PROPOSITION 4.2. *The Cascade algorithm used with $\frac{1}{\sqrt{2}}h[]$ computes $\langle f, \psi_i^a \rangle$ and $\langle f, \phi_i^a \rangle$.*

We now use the change of basis. The next proposition is clear from the definition of $\{\psi_i^b\}$.

PROPOSITION 4.3. *The problem of finding a representation \hat{f} with $\{z_i\}$ and basis $\{\psi_i\}$ is equivalent to finding the same representation \hat{f} using the coefficients $\{y_i\}$ and basis $\{\psi_i^b\}$. The correspondence is $y_i = 2^{-j/2}z_i$ where $i = (j, s)$.*

LEMMA 4.1. Let $\{y_i^*\}$ be the optimal solution using the basis set $\{\psi_i^b\}$ for the reconstruction, i.e., $\hat{f} = \sum_i y_i^* \psi_i^b$ and $\|f - \hat{f}\|_p = \mathcal{E}$. Let $\{y_i^\rho\}$ be the set where each y_i^ρ is rounded to the nearest multiple of ρ . If $f^\rho = \sum_i y_i^\rho \psi_i^b$ then $\|f - f^\rho\|_p \leq \mathcal{E} + O(qn^{1/p} \rho \log n)$.

Proof. From a generalization of the wavelet thresholding result it is straightforward that $\|f - f^\rho\|_p \leq \mathcal{E} + O(qn^{1/p} \rho \log n \max_i \|\psi_i^b\|_\infty)$. Now ψ_i^b is $2^{j/2} \psi_i$. From the proof of Lemma 3.3 we know that for large j $\|\psi_i\|_\infty$ is at most $2^{-j/2}$ times a constant. For smaller j the $\|\psi_i^b\|_\infty$ is a constant.

We will provide a dynamic programming formulation using the new basis. But we still need to show two results; the first concerning y_i^* and the second concerning the $a_j[\cdot]$'s. The next lemma is very similar to Lemma 3.1 and follows from the fact that $\|\psi_{j,s}^a\|_1 = 2^{-j/2} \|\psi_{j,s}\|_1 \leq \sqrt{2q}$.

LEMMA 4.2. $-C_0 \sqrt{q} \mathcal{E} \leq \langle f, \psi_i^a \rangle - y_i^* \leq C_0 \sqrt{q} \mathcal{E}$ for some constant C_0 .

Now suppose we know the optimal solution \hat{f} , and suppose we are computing the coefficients $a_j[\cdot]$ and $d_j[\cdot]$ for both f and \hat{f} at each step j of the Cascade algorithm. We wish to know by how much their coefficients differ since bounding this gap would shed more light on the solution \hat{f} .

PROPOSITION 4.4. Let $a_j[s](F)$ be $a_j[s]$ computed from $a_0[s] = F(s)$ then $a_j[s](f) - a_j[s](\hat{f}) = a_j[s](f - \hat{f})$.

LEMMA 4.3. If $\|f - \hat{f}\|_p \leq \mathcal{E}$ then $|a_j[s](f - \hat{f})| \leq C_1 \sqrt{q} \mathcal{E}$ for some constant C_1 . (We are using $\frac{1}{\sqrt{2}} h[\cdot]$.)

Proof. The proof is near identical to Lemma 3.1. Let $F = f - \hat{f}$. We know $-\mathcal{E} \leq F(t) \leq \mathcal{E}$. This gives us $-\mathcal{E} \|\phi_{j,s}^a\|_1 \leq \langle F, \phi_{j,s}^a \rangle = a_j[s](F) \leq \mathcal{E} \|\phi_{j,s}^a\|_1$. We now invoke the fact that $\phi_{j,s}^a = 2^{-j/2} \phi_{j,s}$. Further $\|\phi_{j,s}\|_2 = 1$ and has at most $(2q)2^j$ non-zero values. Taken together, we get $\|\phi_{j,s}^a\|_1 \leq \sqrt{2q}$.

At this point we have all the pieces. Summarizing:

LEMMA 4.4. Let $\{z_i\}$ be a solution with B non-zero coefficients and let $\hat{f} = \sum_i z_i \psi_i$. If $\|f - \hat{f}\|_p \leq \mathcal{E}$, then there is a solution $\{y_i\}$ with B non-zero coefficients s.t. if (i) y_i is a multiple of ρ (ii) $|y_i - \langle f, \psi_i^a \rangle| \leq C_0 \sqrt{q} \mathcal{E}$ and (iii) $|\langle f, \phi_i^a \rangle - \langle f', \phi_i^a \rangle| \leq C_1 \sqrt{q} \mathcal{E}$ where $f' = \sum_i y_i \psi_i^b$, then $\|f - f'\|_p \leq \mathcal{E} + O(qn^{1/p} \rho \log n)$.

4.3 Haar Systems and a Streaming FPTAS We will assume that we know \mathcal{E} ; which can be circumvented

by $O(\log n)$ guesses, and running as many instances of the algorithm presented below ‘in parallel’. This will increase the time and space requirements by a $O(\log n)$ factor, which is accounted for in Theorem 4.1 below.

The Algorithm: Given \mathcal{E} the algorithm is:

- Let $\rho = \epsilon \mathcal{E} / (cqn^{1/p} \log n)$ for some suitably large constant c . Note that $q = 1$ in the Haar case.
- Consider the $\log n$ level frontier from root of the coefficient tree to leaf t as t changes, i.e., as new data shows up.
- At each level of the tree keep a binary counter.
- At node i , compute $u_i = \langle f, \phi_i^a \rangle$ and the wavelet coefficient $o_i = \langle f, \psi_i^a \rangle$. This involves using the $a[\cdot]$ values of the two children and taking their average to compute u_i and their difference divided by 2 to compute o_i (recall that we are using $\frac{1}{\sqrt{2}} h[\cdot]$).
- If at any point of time the number of coefficients larger than \mathcal{E} exceeds B then we know our guess of \mathcal{E} is wrong and we abort that thread.
- For all values v s.t. $|v - u_i| \leq c_2 \mathcal{E}$ where c_2 is a large enough constant and v is a multiple of ρ , compute the table $\text{ERR}[i, v, b]$ for all $0 \leq b \leq B$. This uses the tables of the two children i_L, i_R . The size of the table is $O(\epsilon^{-1} B n^{1/p} \log n)$. Note that the value y_i of a chosen coefficient at node i is at most a constant multiple of \mathcal{E} away from o_i . Keeping track of the chosen coefficients (the answer) costs $O(B)$ factor space more.
- If the binary counter is 0, stop; Else discard the tables for the two children and proceed one level up.

THEOREM 4.1. The above algorithm is a $O(\epsilon^{-1} B n^{1/p} \log^3 n)$ space algorithm that computes a $(1 + \epsilon)$ approximation to the best B -term unrestricted representation of a signal in the Haar system. Under the ℓ_p norm, the algorithm runs in time $O(n^{1+2/p} \epsilon^{-2} B \log^3 n)$. Under ℓ_∞ the running time becomes $O(n \epsilon^{-2} \log^2 B \log^3 n)$.

4.4 PTAS for multi-dimensional Haar Systems

Our algorithm and analysis extend to multi-dimensional Haar wavelets when the dimension D is a given constant. For $D \geq 2$ define $2^D - 1$ mother wavelets (see also [6, 5]). For all integers $0 \leq d < 2^D$ let

$$\psi^d(x) = \theta^{d_1}(x_1) \theta^{d_2}(x_2) \cdots \theta^{d_D}(x_D),$$

where $d_1 d_2 \dots d_D$ is the binary representation of d and $\theta^0 = \phi, \theta^1 = \psi$. For $d = 0$ we obtain the D -dimensional

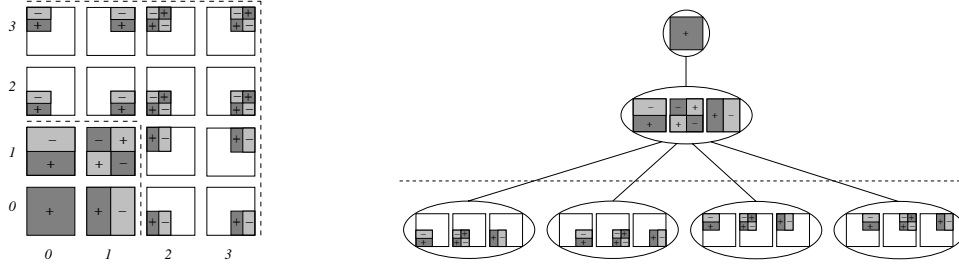


Figure 2: Support regions and coefficient tree structure of the 2-dimensional Haar basis vectors for a 4×4 data array. Each node other than the root contains the 3 wavelet coefficients having the same support region.

scaling function $\psi^0(x) = \phi(x_1)\phi(x_2) \cdots \phi(x_D)$. At scale 2^j and for $s = (s_1, s_2, \dots, s_D)$ define

$$\psi_{j,s}^d(x) = 2^{-Dj/2} \psi^d \left(\frac{x_1 - 2^j s_1}{2^j}, \dots, \frac{x_D - 2^j s_D}{2^j} \right).$$

The family $\{\psi_{j,s}^d\}_{1 \leq d < 2^D, (j,s) \in \mathbb{Z}^{D+1}}$ is an orthonormal basis of $L^2(\mathbb{R}^D)$ [18, Thm. 7.25]. Note that in multi-dimensions we define $\psi_{j,s}^{a,d} = 2^{-Dj/2} \psi_{j,s}^d$, $\psi_{j,s}^{b,d} = 2^{Dj/2} \psi_{j,s}^d$ and $\phi_{j,s}^{a,d} = 2^{-Dj/2} \psi_{j,s}^0$ which is analogous to Definition 4.1. Thus $\|\psi_{j,s}^{a,d}\|_1 = \|\phi_{j,s}^{a,d}\|_1 = 1$ since $\|\psi_{j,s}^d\|_1 = 2^{Dj} 2^{-Dj/2} = 2^{Dj/2}$. Also $\|\psi_{j,s}^{b,d}\|_\infty = 1$.

As an illustration of a multi-dimensional basis, Figure 2 [5, Fig. 1,2] shows the support regions and signs of the 16 two-dimensional Haar basis vectors corresponding to a 4×4 input array. The figure shows that the support region of say $\psi_{1,(0,0)}^3$, which corresponds to entry (2,2) in the array, is the 2^{2D} ($j = 1, D = 2$) elements comprising the lower left quadrant of the input array. Figure 2 also shows the coefficient tree for this case. In general, each node in the tree has 2^D children and corresponds to $2^D - 1$ coefficients (assuming the input is a hypercube). The structure of the coefficient tree will result in a $O(R^{2^D-1})$ increase in running time over the one-dimensional case where $R = O(\epsilon^{-1} n^{1/p} \log n)$.

As in Section 4.1, we associate an error array $\text{ERR}[i, b, v]$ with each node i in the tree where v is the result of the choices of i 's ancestors and $b \leq B$ is the number of coefficients used by the subtree rooted at i . The size of each table is thus $O(\min\{2^{Dj}, B\}R)$ where j is the level of the tree to which i belongs. When computing an entry $\text{ERR}[i, b, v]$ in the table, we need to choose the best non-zero subset S of the $2^D - 1$ coefficients that belong to the node and the best assignment of values to these $|S|$ coefficients. These choices contribute a factor $O((2R)^{2^D-1})$ to the time complexity. We also have to choose the best partition of the remaining $b - |S|$ coefficients into 2^D parts adding

another $O(B^{2^D})$ factor to the running time. We can avoid the latter factor by ordering the search among the node's children as in [5, 6]. Each node is broken into $2^D - 1$ subnodes: Suppose node i has children c_1, \dots, c_{2^D} ordered in some manner. Then subnode i_t , will have c_t as its left child and subnode i_{t-1} as its right child. Subnode i_{2^D-1} will have c_{2^D-1} and c_{2^D} as its children. Now all subnode i_t needs to do is search for the best partition of b into 2 parts as usual. Specifically, fix S and the values given to the coefficients in S . For each v, b' with $0 \leq b' \leq \min\{2^{Dj}, b - |S|\}$, each subnode starting from i_{2^D-1} computes the best allotment of b' coefficients to its children. This process takes $O(R(\min\{2^{Dj}, B\})^2)$ time per subnode. For ℓ_∞ the bounds are better. All the error arrays for the subnodes are discarded before considering the next choice of S and values assigned to its elements. Hence, assuming the input is of size N , and since there are $N/2^{Dj}$ nodes per level of the coefficient tree, the total running time is

$$O \left(\sum_{j=1}^{\frac{\log N}{D}} \frac{N}{2^{Dj}} (2R)^{2^D-1} 2^D R (\min\{2^{Dj}, B\})^2 \right) = O(NBR^{2^D})$$

where we dropped the constant factors involving D in the final expression. Finally recall from Section 4.3 that we need to make $O(\log N)$ guesses for the error \mathcal{E} .

4.5 General Compact Systems We show a simple dynamic programming algorithm that finds a $(1 + \epsilon)$ -approximation to the wavelet synopsis construction problem under the ℓ_∞ norm. The algorithm uses $g(q, n) = n^{O(q(\log q + \log \log n))}$ time and space. Under the ℓ_p norm, the algorithm uses $n^{O(q(\log q + \frac{\log n}{p}))}$ time and space. We will describe the algorithm for the Daubechies wavelet under the ℓ_∞ norm. Recall that the Daubechies filters have $2q$ non-zero coefficients.

For a given subproblem, call an edge an *interface edge* if exactly one of its endpoints is in the subproblem.

Each interface edge has a value associated with it which is eventually determined at a later stage. We will maintain that each subproblem has at most $4q \log n$ interface edges. A subproblem has a table ERR associated with it where for each $b \leq B$ and each configuration I of values on interface edges, $\text{ERR}[b, I]$ stores the minimum contribution to the overall error when the subproblem uses b coefficients and the interface configuration is I . From Lemma 4.4, setting $\rho = \epsilon \mathcal{E} / (c_1 q \log n)$ for some suitably large constant c_1 , each interface edge can have one of $V = O(\frac{q^{3/2} \log n}{\epsilon})$ values under the ℓ_∞ norm. Hence, the size of ERR is bounded by $BV^{4q \log n} = g(q, n)$.

The algorithm starts with an initialization phase that creates the first subproblem. This phase essentially flattens the cone-shape of the coefficient graph, and the only difference between it and later steps is that it results in one subproblem as opposed to two. We select any $2q$ consecutive leaves in the coefficient graph and their ancestors. This is at most $2q \log n$ nodes. We will guess the coefficients of the optimal solution associated with this set of nodes. Again, from Lemma 4.4, each coefficient can take one of $W = O(\frac{q^{3/2} \log n}{\epsilon})$ values under the ℓ_∞ norm. For each of the $(2W)^{2q \log n} = g(q, n)$ guesses, we will run the second phase of the algorithm.

In the second phase, given a subproblem A , we first select the $2q$ ‘middle’ leaves and their ancestors. Call this strip of nodes S . Note that $|S| \leq 2q \log n$. The nodes in S break A into two smaller subproblems L and R (see Figure 3). Suppose we have ERR_L and ERR_R , the two error arrays associated with L and R respectively. We compute each entry $\text{ERR}_A[b, I]$ as follows. First, we guess the b' non-zero coefficients of the optimal solution associated with the nodes in S and their values. Combined with the configuration I , these values define a configuration I_L (resp. I_R) for the interface edges of L (resp. R) in the obvious way. Furthermore, they result in an error e associated with the leaf nodes in S . Hence,

$$\text{ERR}[b, I] = e + \min_{b''} \max\{\text{ERR}_L[b'', I_L], \text{ERR}_R[b - b' - b'', I_R]\}.$$

Therefore, computing each entry in ERR takes at most $B(2W)^{2q \log n} = g(q, n)$ time. The running time of the algorithm follows.

THEOREM 4.2. *We can compute a $(1 + \epsilon)$ approximation to the best B -term unrestricted representation of a compact system under the ℓ_∞ norm in time $n^{O(q(\log q + \log \log n))}$.*

5 Adaptive Quantization

Wavelets are extensively used in compression of images and audio signals. In such a setting a small percent

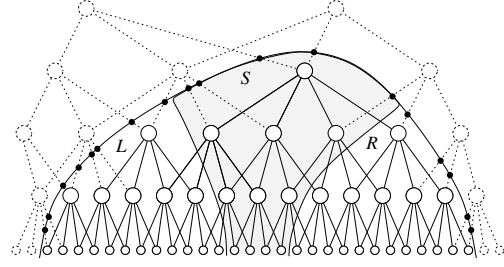


Figure 3: An example subproblem. The shaded nodes belong to the strip S . The edges crossing the ‘frontier’ are interface edges.

saving of space is considered important and attention is paid to the bits being stored. These techniques are heavily engineered and typically designed by some domain expert. The complexity is usually two-fold.

First, the numbers z_i do not all cost the same to represent. In some strategies, e.g. strategies used for audio signals, the number of bits of precision to represent z_i for a $\psi_i = \psi_{j,s}$ is fixed, and typically depends only on j . Further the a_j ’s are computed with a higher precision than the d_j ’s. This affects the space needed by the top-most coefficients. In yet another strategy, which is standard to a broad compression literature, it is assumed that we require $\log_2 z$ bits to represent a number z . All of these bit counting techniques need to assume that the signal is bounded and there is some reference unit of precision.

Second, recall that we assumed there is a mapping from ψ_i to $\psi_{j,s}$. In several systems, a bitmap is used to indicate the non-zero entries. However the bitmap requires $O(n)$ space and it is often preferred that we store only the status of the non-zero values instead of the status of all values in the transform. In a $o(n)$ space setting, as in the streaming setting, we need to store the location i , and the map between coefficients and locations becomes important. For example, we can represent $\psi_{j,s}$ using $\log \log n + \log(n/2^j) + O(1)$ bits instead of $\log n$ bits to specify i . Supposing only the vectors with support of \sqrt{n} or larger are important for a particular signal, we will only use half the number of bits. Notice that this way of encoding *increases the number of bits required for a coefficient of small j to larger than $\log n$* and this is mitigated (hopefully) by savings at larger j .

Most of the strategies in the literature can be covered by simply assuming that deciding to store any number for ψ_i has a fixed cost c_i . A few remaining strategies assume that to store a number z the cost $b(z_i)$ depends on z_i . In those cases it is a fair assumption that

storing a number z_i which satisfies $|a - z_i| < t$ for a fixed constant a depends only on t .

In the case where the cost of storing a number for i is a constant c_i we arrive at a quadratic program similar to (3.1), i.e. minimize τ with the obvious constraints $x_i \in \{0, 1\}$ and $\sum_i x_i c_i \leq B$, and

$$(5.2) \quad -\tau \|\psi_i\|_1 \leq \langle f, \psi_i \rangle - x_i z_i^* \leq \tau \|\psi_i\|_1 \quad \text{for all } i$$

The above can be clearly solved optimally since the c_i 's are polynomially bounded. We can also exceed space bounds by at most one item and solve the above using rules similar to Smith's ratio rule. In the case where the cost is dependent on z_i we cannot write such a system of equations, but we can guess τ (up to constant factors) and verify if the guess is correct. To verify the guess, we need to be able to solve equations of the form $\min_z b(z)$ s.t. $|a - z| \leq t$. This is solvable for most reasonable cost models, e.g. if $b(z)$ is simply $\log_2 |z|$. After we get a lower bound τ we can proceed as in the B -term case and get a $O(n^{1/p} \log n)$ approximation while respecting the space constraint B by following arguments similar to those in Section 3.

THEOREM 5.1. *In the model where storing a number z_i corresponding to the basis ψ_i has cost $c(i) + b(z_i)$ and we can solve $\min_z b(z) : |a - z| \leq t$, we can achieve a $O(n^{1/p} \log n)$ approximation in a one pass data stream model.*

Observe that in adaptive quantization, the precision is fixed. Once the precision of the representation is fixed, it is unclear if a PTAS exists since it is much harder to find a solution that respects the fixed precision. In fact this introduces combinatorial properties which are interesting in their own right. Resolving the approximation guarantee of quantizations remains an interesting problem both in theory and in practice.

References

- [1] K. Chakrabarti, E. J. Keogh, S. Mehrotra, and M. J. Pazzani. Locally adaptive dimensionality reduction for indexing large time series databases. *ACM TODS*, 27(2):188–228, 2002.
- [2] Kaushik Chakrabarti, Minos N. Garofalakis, Rajeev Rastogi, and Kyuseok Shim. Approximate query processing using wavelets. In *VLDB Conference*, 2000.
- [3] I. Daubechies. *Ten Lectures on Wavelets*. SIAM, 1992.
- [4] R. DeVore. Nonlinear approximation. *Acta Numerica*, pages 1–99, 1998.
- [5] M. Garofalakis and A. Kumar. Deterministic wavelet thresholding for maximum error metric. *Proc. of PODS*, 2004.
- [6] M. N. Garofalakis and P. B. Gibbons. Probabilistic wavelet synopses. *ACM TODS*, 29:43–90, 2004.
- [7] A. C. Gilbert, S. Guha, P. Indyk, Y. Kotidis, S. Muthukrishnan, and Martin Strauss. Fast, small-space algorithms for approximate histogram maintenance. In *Proc. of ACM STOC*, 2002.
- [8] A. C. Gilbert, Y. Kotidis, S. Muthukrishnan, and M. Strauss. Optimal and approximate computation of summary statistics for range aggregates. In *Proc. of ACM PODS*, 2001.
- [9] Anna C. Gilbert, S. Muthukrishnan, and Martin Strauss. Approximation of functions over redundant dictionaries using coherence. *Proc. of SODA*, pages 243–252, 2003.
- [10] S. Guha. Space efficiency in synopsis construction problems. *Proc. of VLDB Conference*, 2005.
- [11] S. Guha and B. Harb. Wavelet synopsis for data streams: Minimizing non-euclidean error. *Proc. of KDD*, 2005.
- [12] S. Guha, P. Indyk, S. Muthukrishnan, and M. Strauss. Histogramming data streams with fast per-item processing. In *Proc. of ICALP*, 2002.
- [13] S. Guha, N. Koudas, and K. Shim. Data Streams and Histograms. In *Proc. of STOC*, 2001.
- [14] S. Guha, N. Mishra, R. Motwani, and L. O'Callaghan. Clustering data streams. *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*, pages 359–366, 2000.
- [15] Y. E. Ioannidis. The history of histograms (abridged). *Proc. of VLDB Conference*, pages 19–30, 2003.
- [16] C. E. Jacobs, A. Finkelstein, and D. H. Salesin. Fast multiresolution image querying. *Computer Graphics*, 29(Annual Conference Series):277–286, 1995.
- [17] E. Keogh, K. Chakrabati, S. Mehrotra, and M. Pazzani. Locally Adaptive Dimensionality Reduction for Indexing Large Time Series Databases. *Proc. of ACM SIGMOD*, Santa Barbara, March 2001.
- [18] S. Mallat. *A wavelet tour of signal processing*. Academic Press, 1999.
- [19] Y. Matias and D. Urieli. Personal communication, 2004.
- [20] Y. Matias and D. Urieli. Optimal workload-based weighted wavelet synopses. *Proc. of ICDT*, pages 368–382, 2005.
- [21] Y. Matias, J. Scott Vitter, and M. Wang. Wavelet-Based Histograms for Selectivity Estimation. *Proc. of ACM SIGMOD*, 1998.
- [22] S. Muthukrishnan. Nonuniform sparse approximation using haar wavelet basis. *DIMACS TR 2004-42*, 2004.
- [23] Jeffrey Scott Vitter and Min Wang. Approximate computation of multidimensional aggregates of sparse data using wavelets. In *SIGMOD Conference*, 1999.