Satisficing Search Algorithms for Selecting Near-Best Bases in Adaptive Tree-Structured Wavelet Transforms

Carl Taswell

Abstract—**Satisficing search algorithms are proposed for adaptively selecting near-best basis and near-best frame decompositions in redundant tree-structured wavelet transforms. Any of a variety of additive or non-additive information cost functions can be used as the decision criterion for comparing and selecting nodes when searching through the tree. The algorithms are applicable to tree-structured transforms generated by any kind of wavelet whether orthogonal, biorthogonal, or non-orthogonal. These satisficing search algorithms implement sub-optimizing rather than optimizing principles, and acquire the important advantage of reduced computational complexity with significant savings in memory, flops, and time. Despite the sub-optimal approach, top-down tree-search algorithms with additive or non-additive costs that yield near-best bases can be considered, in certain important and practical situations, better than bottom-up tree-search algorithms with additive costs that yield best bases. Here "better than" means that effectively the same level of performance can be attained for a relative fraction of the computational work. Experimental results comparing the various information cost functions and basis selection methods are demonstrated for both data compression of real speech and time-frequency analysis of artificial transients.**

Keywords— **Satisficing search, near-best basis, information cost, wavelet packet transform, cosine packet transform, time-frequency analysis, data compression, speech coding.**

EDICS— **SP 2.2.1, 2.2.7, 2.4.4, and 2.4.5.**

I. Introduction

Coifman and Wickerhauser [2] presented an algorithm for the selection of the best basis representation of a signal within a collection of orthonormal basis representations. They defined redundant transforms, including wavelet packet transforms and local trigonometric transforms, that generate these basis libraries, each of which can be structured and searched as a full balanced binary tree. To select a particular basis within a given library (a particular sub-tree of a tree), they defined the best basis to be that which minimized an information cost function $\mathcal{C}$ and chose the $-\ell^2 \ln \ell^2$ functional (related to the Shannon entropy) as their archetype for $\mathcal{C}$. The optimality of the best basis algorithm requires the key restriction of additivity for $\mathcal{C}$. The computational cost of the best basis algorithm is $O(LN)$ where $L = \lfloor \log_2 N \rfloor$ is the number of levels or depth of the transform or tree, and N is the length of the signal.

Mallat and Zhang [3] developed a greedy algorithm for the selection of the best matching pursuit decomposition of a signal into time-frequency packets from a large collection of such packet waveforms. The matching pursuit decomposition can be performed either with or without backprojection [3], [4] resulting in either orthogonal or non-orthogonal decompositions. The computational cost of the non-orthogonal matching pursuit algorithm is $O(MLN)$ where $M <= N$ is the number of packets selected. The matching pursuit algorithm with its local optimization properties guarantees a more compact signal decomposition (typically $M << N$) than that of the best basis algorithm with its global optimization properties. However, this more compact signal representation can be obtained only at the expense of more computation in which although $M << N$, nevertheless $1 << M$, such that the additional computational cost of the matching pursuit algorithm is significant relative to that of the best basis algorithm.

Hybrid algorithms combining the principles of both best basis and matching pursuit decompositions have since been developed. Coifman and Wickerhauser [5] proposed an algorithm which they called adaptive waveform analysis or adaptive waveform denoising. Here the computational complexity is $O(KLN)$ where K is the number of denoising iterations. Since typically $K < M$, adaptive waveform denoising has also been described as a "fast approximate version of the matching pursuit procedure" [6, page 418]. Another new "meta-algorithm" [7, page 8] was described by Saito [8]. His algorithm incorporates a best basis algorithm to select the best basis within each of multiple libraries and then selects the final decomposition as the best basis of the best library. Here the complexity is $O(JLN)$ where J is the number of search libraries. In all of these algorithms, the actual complexity depends on the particular implementations necessitated by the particular search libraries as well as search paths and search decision criteria. Thus, the complexity estimates summarized here are approximate lower bounds.

To varying degrees, all of these algorithms trade the mutually exclusive optimal design goals of efficiency of algorithmic computation versus efficiency of signal representation (ie, compression). Using the original Coifman-Wickerhauser and Mallat-Zhang algorithms as prototypes, the design trade-off raises several questions: Can the compression efficiency of the Mallat-Zhang matching pursuit decomposition be attained or approached by an algorithm with the computation efficiency of the Coifman-

C. Taswell was with Scientific Computing and Computational Mathematics, Stanford University Department of Computer Science, Stanford, CA 94305-9025. He is now with ToolSmiths, Stanford, CA 94309-9925. Web: http://www.toolsmiths.com; Internet: taswell@toolsmiths.com; Tel: 415-323-4336; Fax: 415-323-5779. Published in October 1996 IEEE Transactions on Signal Processing 44(10):2423–2438; revision dated 2/27/96; original manuscript dated 6/27/95 available as an SCCM Technical Report [1].

Wickerhauser best basis decomposition? Which information cost functions and basis selection methods should be used to choose a basis in an adaptive tree-structured transform? Can compromise algorithms be developed with intermediate or adjustable rates of efficiency of compression and computation as required by the application? Under what circumstances is it relevant and necessary to perform additional computation in order to obtain more compression? Where is the transition point of diminishing returns in the range of choices from the fixed standard wavelet transform to the adaptive wavelet packet transforms with decompositions selected by complete basis searches and matching pursuit searches?

In an initial attempt to explore these questions, Taswell proposed near-best bases with non-additive costs [9] as an alternative to best bases with additive costs [2]. In subsequent work, Taswell proposed several more basis selection algorithms and decision criteria [10]. In particular, top-down and bottom-up tree searches were distinguished. Moreover, the statistical performance of the various basis and pursuit decompositions was investigated with Monte Carlo experiments on test signals with additive white noise [10], [11]. Although some of the individual components of these algorithms had been considered previously by others (such as the various cost functions discussed by Wickerhauser [12]), actual experimental results from this previous work were limited to Coifman-Wickerhauser "entropy" selected best bases [2]. As a consequence, the results presented by Taswell [9], [10], [13], [11] provided the first systematic experimental investigation of both search paths and decision criteria. General conclusions from these experiments can be summarized with the following remarks: If the standard wavelet transform is considered to provide insufficient compression and the wavelet packet matching pursuit decomposition to require excessive computation, then the wavelet packet complete basis decomposition can be considered an effective compromise. Moreover, if the wavelet packet complete basis decomposition is accepted as the method of choice for the given problem, then a near-best basis selected by a top-down tree search (with either additive or non-additive costs) is just as good as a best basis selected by a bottom-up tree search (with additive costs), and can be obtained at a relative fraction of the computational requirements measured in memory, flops, and time.

In this report, I demonstrate the effectiveness of these near-best bases for a real-world class of signals. In particular, I apply these methods to the coding of speech signals from a large speech database containing sentences spoken in all major American dialects [14]. However, developing a state-of-the-art speech coder is *not* the objective of this work. Rather it is simply to demonstrate the performance advantages of near-best bases relative to best bases in the context of a real-world application. As in previous reports [10], [11], the work is pragmatic and rooted in the empiric satisficing philosophies of Simon [15], [16] and Berkson [17]. This empirical approach contrasts with much of the statistically oriented wavelet literature which focuses on theoretical models of stochastic processes and/or asymptotic properties of statistical estimators related to wavelet analysis and methods.

Nevertheless, according to Berkson [17], "Statistics, however you define it, is very much earthbound and deals with real observable data; what is statistically true must be literally verifiably true for such data." Referring to theorems of aymptotic analysis, he believed that "if these theorems were valid for large samples, they must refer to *infinitely* large samples, which is to say, samples so large that no statistician ever gets them, at least not on this unpleasant earth." As a consequence, he advocated the use of actual experiments to evaluate the performance of statistical methods on small samples. It is this pragmatic empirical approach of Berkson that is adopted as the foundation for the work on satisficing searches for near-best bases presented in this report. Its relation to the satisficing search of Simon is discussed in detail in Section VII. To emphasize with further clarity the pragmatic nature of the work, all methods and algorithms are presented and discussed using matrix data structures and generic pseudocode typical of high-level languages such as MATLAB, with explicit names of functions being those of the software library WavBox 4 [18]. Since the function input and output arguments and function dependencies are not arbitrary for the different algorithms investigated here, the explicit description of the algorithms with the particular example of WavBox 4 serves to clarify the distinctions between the various alternatives.

In the following Sections II–IV, the mathematical definitions, notations, and computational algorithms for adaptive tree-structured wavelet transforms and the selection of their best basis and near-best basis decompositions are described in detail for the data structures (Section II on Discrete Packet Transforms), decision criteria (Section III on Information Cost Functions), and search paths (Section IV on Basis Selection Methods). Case studies demonstrating the application of these methods to the time-frequency analysis of artificial transients and chirps are presented in Section V, while population studies demonstrating their application to the compression of speech data are reported with statistical performance results in Section VI. In the final Section VII, the various methods and results are reviewed and compared in the context of the satisficing perspective for which the development attributed to Simon [15], [16] is also reviewed in greater depth.

II. Discrete Packet Transforms

We consider vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^N$ and orthonormal transformation matrix $\mathbf{B} \in \mathbb{R}^{N \times N}$. Then $\mathbf{y} = \mathbf{B}\mathbf{x}$ and $\mathcal{C}(\mathbf{y})$ are respectively the coefficient vector and information cost scalar for the signal $\mathbf{x}$ in the transform coordinate system represented by the basis $\mathbf{B}$. We wish to find a basis $\mathbf{B}$ for which $\mathcal{C}(\mathbf{y})$ is minimal, subject possibly to some constraint on the search $\mathcal{S}$ for the basis $\mathbf{B}$. To do so, we require various data structures for representing information relevant to the transforms and selected decompositions. Thus, a discrete packet transform (DPT) is considered to be any multiresolution transform, such as a wavelet packet transform (WPT), cosine packet transform (CPT), or other local trigonometric transform, that yields a table of transform coefficients which can be organized as a balanced binary tree. The transform table is called a discrete packet table $\mathbf{P}^{\text{table}}$ with levels l and blocks b of the table corresponding to levels l and branches b of the tree. For the sake of mnemonics, the term branch is often used here instead of the more customary term node. However, the conventional term node is also used synonymously in this report. Thus, the root node at level 0 and the terminal nodes at level L are considered to be the top and bottom of of the full balanced tree corresponding, respectively, to the finest and coarsest time resolutions of the data for tree-structured wavelet transforms, and to the finest and coarsest frequency resolutions of the data for tree-structured trigonometric transforms.

There are 2^l blocks on each level and thus $K = 2^{(L+1)} - 1$ blocks in the entire table. Within each block b on level l, there are $2^{-l}N$ cells c where N is the length of the original signal $\mathbf{x} \in \mathbb{R}^N$. Each coefficient in the packet table $\mathbf{P}^{\text{table}}$ can be specified as the 4-vector $[a, l, b, c]$ where a is the packet's amplitude and l, b, and c are its level, block, and cell indices. For wavelet transforms, these level, block, and cell indices also correspond to scale, frequency, and time indices; and for trigonometric transforms, they correspond to scale, time, and frequency indices. Thus, as members of the generic class of DPTs, the tree-structured wavelet transforms such as the WPT and the tree-structured trigonometric transforms such as the CPT can be viewed as duals with regard to time and frequency.

Now, if the signal $\mathbf{x} \in \mathbb{R}^N$ has N samples to be transformed, then a DPT to a depth of L levels yields a packet table matrix $\mathbf{P}^{\text{table}} \in \mathbb{R}^{N\times(L+1)}$ with a total of $(L+1)N$ coefficients. A particular basis within this redundant representation can be specified with the basis selection tree $\mathbf{S} \in \chi^K$ where each of the K variables $\chi_k \in \{0, 1\}$ is an indicator variable for the selection of the k^{th} block/branch of the table/tree. The redundant table $\mathbf{P}^{\text{table}} \in \mathbb{R}^{N\times(L+1)}$ can then be converted to the non-redundant basis $\mathbf{P}^{\text{basis}} \in \mathbb{R}^N$. In WavBox 4 [18], the function *dpt2dpb* (discrete packet table to discrete packet basis) performs this restructuring of the data via the mapping $\mathbf{P}^{\text{basis}} = \text{dpt2dpb}(\mathbf{P}^{\text{table}}, \mathbf{S})$.

To compare various decompositions, it is also convenient to convert discrete packet tables $\mathbf{P}^{\text{table}}$ or bases $\mathbf{P}^{\text{basis}}$ to discrete packet lists $\mathbf{P}^{\text{list}}$ representing the selected decompositions. In WavBox 4, the functions *dpt2dpl* and *dpb2dpl* perform this restructuring of the data via the mappings $\mathbf{P}^{\text{list}} = \text{dpt2dpl}(\mathbf{P}^{\text{table}}, \mathbf{S})$ and $\mathbf{P}^{\text{list}} = \text{dpb2dpl}(\mathbf{P}^{\text{basis}}, \mathbf{S})$ where again the functions are named as the abbreviations for their input and output data structures analogous to the naming convention for *dpt2dpb*. Each list contains M packets specified as row 4-vectors $[a_i, l_i, b_i, c_i]$ with rows $i = 1, ..., M$ ordered so that $|a_1| \geq \cdots \geq |a_M|$. To study a complete basis decomposition, we must examine the entire list where $M = N$. However, we may also study subsets of the list where $M < N$.

Thus, there are four data structures presented here: $\mathbf{P}^{\text{table}} \in \mathbb{R}^{N\times(L+1)}$, $\mathbf{P}^{\text{basis}} \in \mathbb{R}^N$, $\mathbf{P}^{\text{list}} \in \mathbb{R}^{M\times 4}$, and $\mathbf{S} \in \chi^K$. Since packet tables $\mathbf{P}^{\text{table}}$ and selection trees $\mathbf{S}$ are implemented respectively as matrices and vectors, table blocks and corresponding tree branches indexed by (l, b) are respectively submatrices and scalars. They are denoted $\mathbf{P}^{\text{table}}_{lb} \equiv \mathbf{P}^{\text{table}}_{i_{lb},j_{lb}}$ and $S_{lb} \equiv S_{k_{lb}}$ where for $l \in \{0, 1, \ldots, L\}$ and $b \in \{0, 1, \ldots, 2^l - 1\}$, the row and column vector indices i_{lb}, j_{lb} are for level l block b in a table matrix, and the scalar index k_{lb} is for level l branch b in a tree vector. The same holds true analogously for packet bases $\mathbf{P}^{\text{basis}}_{lb} \equiv \mathbf{P}^{\text{basis}}_{i_{lb}}$ with the proviso that not all levels l and blocks b of $\mathbf{P}^{\text{table}}$ are stored in $\mathbf{P}^{\text{basis}}$ since $\mathbf{P}^{\text{basis}}$ is not redundant by its definition as a basis. In fact, $\mathbf{P}^{\text{basis}}$ contains only those blocks b on levels l for which $S_{lb} = 1$. Correct manipulation of coefficients stored in $\mathbf{P}^{\text{basis}}$ requires using the level-block indexing information encoded as logical values in $\mathbf{S}$. Finally, since the i^{th} packet in $\mathbf{P}^{\text{list}}$ is denoted $\mathbf{P}^{\text{list}}_i \equiv [a_i, l_i, b_i, c_i]$, it should be clear from context that $\mathbf{P}_i$ is from the list $\mathbf{P}^{\text{list}}$ while $\mathbf{P}_{lb}$ is from the table $\mathbf{P}^{\text{table}}$.

III. Information Cost Functions

We consider data vectors $\mathbf{y} \in \mathbb{R}^{2^{-l}N}$ for one parent block and $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^{2^{-(l+1)}N}$ for its two children blocks in a discrete packet table represented by a binary tree. We wish to compare their information costs by some measure used as a decision criterion when searching the table/tree to select a particular basis. In the following definitions, we will assume $l = 0$ which is the case for the root node.

A. Additive Costs and Comparisons

Additive costs were originally intended for use with the best bases of Coifman and Wickerhauser [2].

Definition: A cost functional $\mathcal{C}^{\text{add}}$ from vectors $\mathbf{y} \in \mathbb{R}^N$ to $\mathbb{R}$ is called an *additive information cost function* if $\mathcal{C}^{\text{add}}(0) = 0$ and $\mathcal{C}^{\text{add}}(\mathbf{y}) = \sum_i \mathcal{C}^{\text{add}}(y_i)$.

Definition: The inequality $\mathcal{C}^{\text{add}}(\mathbf{y}) \leq \mathcal{C}^{\text{add}}(\mathbf{z}_1, \mathbf{z}_2)$ between vectors $\mathbf{y} \in \mathbb{R}^N$ and $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^{N/2}$ is called an *additive information cost comparision* if

$$\mathcal{C}^{\text{add}}(\mathbf{z}_1, \mathbf{z}_2) \equiv \mathcal{C}^{\text{add}}(\mathbf{z}_1 \oplus \mathbf{z}_2) = \mathcal{C}^{\text{add}}(\mathbf{z}_1) + \mathcal{C}^{\text{add}}(\mathbf{z}_2).$$

We can define several additive cost functions as

$$\begin{aligned}
\mathcal{C}_1^{\text{add}}(\mathbf{y}) &= \mathcal{E}^p(\mathbf{y}) = \begin{cases} \sum_i |y_i|^p & \text{for } 0 < p < 2 \\ -\sum_i |y_i|^p & \text{for } 2 < p < \infty \end{cases} \\
\mathcal{C}_2^{\text{add}}(\mathbf{y}) &= \mathcal{F}(\mathbf{y}) = -\sum_{i:y_i\neq 0} y_i^2 \ln y_i^2 \\
\mathcal{C}_3^{\text{add}}(\mathbf{y}) &= \mathcal{G}(\mathbf{y}) = \sum_{i:y_i\neq 0} \ln y_i^2
\end{aligned}$$

which are respectively the ℓ^p functional related to energy and the ℓ^p norm, the $-\ell^2 \ln \ell^2$ functional related to Shannon entropy, and the $\ln \ell^2$ functional related to Gauss-Markov entropy[1] (*cf.* [12]).

B. Non-Additive Costs and Comparisons

Non-additive costs were proposed for use with the near-best bases of Taswell [9].

[1] More precisely, the Shannon entropy of a Gauss-Markov process.

Definition: A cost functional $\mathcal{C}^{\text{non}}$ from vectors $\mathbf{y} \in \mathbb{R}^N$ to $\mathbb{R}$ is called a *non-additive information cost function* if it serves as a decision criterion for a basis selection algorithm and it is not an additive information cost function $\mathcal{C}^{\text{add}}$.

Definition: The inequality $\mathcal{C}^{\text{non}}(\mathbf{y}) \leq \mathcal{C}^{\text{non}}(\mathbf{z}_1, \mathbf{z}_2)$ between vectors $\mathbf{y} \in \mathbb{R}^N$ and $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^{N/2}$ is called a *non-additive information cost comparision* if

$$\mathcal{C}^{\text{non}}(\mathbf{z}_1, \mathbf{z}_2) \equiv \mathcal{C}^{\text{non}}(\mathbf{z}_1 \oplus \mathbf{z}_2) \neq \mathcal{C}^{\text{non}}(\mathbf{z}_1) + \mathcal{C}^{\text{non}}(\mathbf{z}_2).$$

We can construct several examples of non-additive cost functions from the probability density function for the data coefficients. In the discrete vector context, a probability mass function (pmf) can be estimated with simple histogram binning methods in conjunction with various rules for the number of bins. Thus let

$$\begin{aligned} J_{\text{S}} &= 1 + \log_2 N \\ J_{\text{D}} &= 1 + \log_2 N + \log_2(1 + \hat{\gamma}\sqrt{N/6}) \\ J_{\text{TS}} &= \sqrt[3]{2N} \end{aligned}$$

be the number of bins J according, respectively, to the Sturges', Doane's, and Terrell-Scott's rules [19, pages 48 and 73], where $\hat{\gamma}$ is an estimate of the standardized skewness coefficient. Given the number of bins J and the sample data interval $[a, b]$ where $a = \min_i y_i$ and $b = \max_i y_i$, then the bin width is $w = (b - a)/J$. Using the bin width w, the frequency f_j for the j^{th} bin is defined as

$$f_j = \#\{y_i \mid y_i \leq a + jw\} - \sum_{k=1}^{j-1} f_k$$

and the probabilities p_j are calculated from the frequencies f_j simply as $p_j = f_j/N$. Let $\mathbf{p}_{\text{S}}$, $\mathbf{p}_{\text{D}}$, and $\mathbf{p}_{\text{TS}}$ denote the pmf vectors $\mathbf{p}$ when estimated with J_{S}, J_{D}, and J_{TS}, respectively.

Now the Shannon entropy $\mathcal{H}_{\text{S}}$ [20] for a finite scheme $\{(A_j, p_j) \mid 1 \leq j \leq J\}$ of events A_j with probabilities p_j is defined as

$$\mathcal{H}_{\text{S}}(\mathbf{p}) = -\sum_{j=1}^{J} p_j \log_2 p_j$$

where the probabilistic events (A_j, p_j) are identified with the fractions of coefficients located within the histogram bin intervals. Therefore, three non-additive cost functions can be defined as

$$\begin{aligned} \mathcal{C}_1^{\text{non}}(\mathbf{y}) &= \mathcal{H}_{\text{S}}(\mathbf{p}_{\text{S}}(\mathbf{y})) \\ \mathcal{C}_2^{\text{non}}(\mathbf{y}) &= \mathcal{H}_{\text{S}}(\mathbf{p}_{\text{D}}(\mathbf{y})) \\ \mathcal{C}_3^{\text{non}}(\mathbf{y}) &= \mathcal{H}_{\text{S}}(\mathbf{p}_{\text{TS}}(\mathbf{y})). \end{aligned}$$

Another non-additive cost function is the Coifman-Wickerhauser entropy $\mathcal{H}_{\text{CW}}$ [2]. This functional is also the Shannon entropy of a finite scheme but one where the probabilistic events (A_j, p_j) are identified with the normalized energies rather than probabilities of the data coefficients:

$$\mathcal{C}_4^{\text{non}}(\mathbf{y}) = \mathcal{H}_{\text{CW}}(\mathbf{y}) = -\sum_{i=1}^{N} \frac{|y_i|^2}{\|\mathbf{y}\|_2^2} \ln \frac{|y_i|^2}{\|\mathbf{y}\|_2^2}.$$

We can construct additional examples of $\mathcal{C}^{\text{non}}$ with the sorted vector $[y_{(k)}]$ where

$$y_{(1)} = |y_{i_1}| \geq \cdots \geq y_{(N)} = |y_{i_N}|$$

so that $y_{(k)} = |y_{i_k}|$ is the k^{th} largest absolute value element of the vector $[y_i]$. The decreasing-absolute-value sorted vector $[y_{(k)}]$ suffices to define the weak-ℓ^p norm (*cf.* [21]). However, constructing the decreasingly sorted, powered, and cumulatively summed vector $[v_k(\mathbf{y}, p)]$, and renormalized vector $[u_k(\mathbf{y}, p)]$ where

$$u_k(\mathbf{y}, p) = \frac{v_k(\mathbf{y}, p)}{v_N(\mathbf{y}, p)} \quad \text{with} \quad v_k(\mathbf{y}, p) = \sum_{i=1}^{k} y_{(i)}^p$$

makes it convenient to define several other $\mathcal{C}^{\text{non}}$. (Note that $0 \leq u_k(\mathbf{y}, p) \leq 1$ because of the normalization.) Thus, with $[y_{(k)}]$ and $[u_k(\mathbf{y}, p)]$ obtained from $[y_i]$, define the non-additive information cost functions

$$\begin{aligned} \mathcal{C}_5^{\text{non}}(\mathbf{y}) &= \mathcal{W}\ell^p(\mathbf{y}) = \max_k \; k^{(1/p)} y_{(k)} \\ \mathcal{C}_6^{\text{non}}(\mathbf{y}) &= \mathcal{N}_f^p(\mathbf{y}) = \arg\min_k |u_k(\mathbf{y}, p) - f| \\ \mathcal{C}_7^{\text{non}}(\mathbf{y}) &= \mathcal{A}^p(\mathbf{y}) = N - \sum_k u_k(\mathbf{y}, p) \end{aligned}$$

which are respectively the weak-ℓ^p norm, data compression number, and data compression area [9].

Here the power p and fraction f are parameters[2] chosen from the intervals $0 < p \leq 2$ and $0 < f < 1$. The functions $\mathcal{N}_f^p$ and $\mathcal{A}^p$ were designed to yield scalar values that could be meaningfully minimized in a basis search algorithm and were named according to their natural or geometric interpretation. For example, choosing $p = 2$ and $f = .99$ and then using $\mathcal{N}_{.99}^2$ yields the minimum number of vector coefficients containing 99% of the energy of the entire vector. The data compression number $\mathcal{N}_f^p$ and area $\mathcal{A}^p$ can be contrasted by observing that the number $\mathcal{N}_f^p$ is a local measure with varying "sensitivity" to different intervals of the u_k versus k curve whereas the area $\mathcal{A}^p$ is a global measure of the entire curve. The minimum values attainable represent maximum compression. They are readily computed for a Kronecker delta vector δ with unit energy: $\mathcal{N}_f^p(\delta) = 1$ and $\mathcal{A}^p(\delta) = 0$.

IV. Basis Selection Methods

Given a particular DPT computed to depth level L for a signal $\mathbf{x}$ of length $N = 2^L$, there are $O(2^N)$ basis decompositions $\mathbf{y}_i = \mathbf{B}_i \mathbf{x}$ resulting from the library of orthogonal transforms $\mathbf{B}_i \in \mathcal{B}$ generated by the redundant DPT [2]. We wish to select one of these pairs $(\mathbf{B}_i, \mathbf{y}_i)$ by searching the tree-structured packet table with a given information cost functional $\mathcal{C}$, and by minimizing the cost $\mathcal{C}(\mathbf{y}_i)$ over $\mathbf{y}_i$ subject possibly to some constraint on the search $\mathcal{S}$ restricting the number of $\mathbf{y}_i$ examined.

[2] The use of the parameter p for power and f for fraction should not be confused with the use of the vectors $\mathbf{p}$ and $[p_j]$ for probabilities and $\mathbf{f}$ and $[f_j]$ for frequencies.

A. Best and Near-Best Bases

Coifman and Wickerhauser defined the best basis for additive costs [2]. Subsequently, Taswell defined the near-best basis for non-additive costs [9], and then more generally for both non-additive and additive costs [10]. Here we review definitions for both best and near-best bases.

Definition: The *best basis* relative to $\mathcal{C}^{\text{add}}$ for a vector $\mathbf{x}$ in a library $\mathcal{B}$ of bases is that $\mathbf{B}$ for which $\mathcal{C}^{\text{add}}(\mathbf{Bx})$ is minimal.

Definition: The *near-best basis* relative to $\mathcal{C}$ (either $\mathcal{C}^{\text{non}}$ or $\mathcal{C}^{\text{add}}$) for a vector $\mathbf{x}$ in a library $\mathcal{B}$ of bases is that $\mathbf{B} \in \mathcal{B}^* \subset \mathcal{B}$ for which $\mathcal{C}(\mathbf{Bx})$ is minimal subject to the constraints of the search within the subset $\mathcal{B}^*$ defined by the search type.

Here $\mathcal{B}^*$ is the proper subset of library bases that are searched by the selection algorithm. Searching the subset $\mathcal{B}^*$ defined by the Coifman-Wickerhauser bottom-up tree search (*cf.* Section IV-B) yields the optimal or best basis within the entire library $\mathcal{B}$ for an *additive* information cost function $\mathcal{C}^{\text{add}}$ (*cf.* proof [2, page 717]). However, since $\mathcal{B}^* \neq \mathcal{B}$, this search is not exhaustive and cannot guarantee the selection of a *best* basis for a *non-additive* information cost function $\mathcal{C}^{\text{non}}$. Moreover, there are many other search types, including top-down tree searches (*cf.* Section IV-B), for which neither additive nor non-additive costs can guarantee the selection of a best basis. For this reason as well as empirical evidence suggesting nearly equivalent performance, a basis selected by either a non-additive or additive cost subject to the constraint of a search within a proper subset $\mathcal{B}^*$ of the library $\mathcal{B}$ is called a *near-best* basis.[3]

B. Bottom-Up and Top-Down Tree Searches

Both the best basis of Coifman and Wickerhauser [2] and the near-best basis of Taswell [9] were originally defined for bottom-up tree searches. However, the near-best basis with additive or non-additive costs permits either a bottom-up or top-down search [10], [13] through the discrete packet table to find the basis selection tree. Searches subject to other patterns of constraint are possible as well. The various search methods are denoted $\mathcal{S}$ generically, with $\mathcal{S} = \mathcal{U}$, $\mathcal{S} = \mathcal{D}$, and $\mathcal{S} = \mathcal{L}$ indicating the particular examples of bottom-Up, top-Down, and fixed-Level search types. The latter case provides the simplest search type since there are only $L+1$ bases examined corresponding to those with all nodes constrained to the same level l of the table and tree. Such a fixed-level basis is also known as a Fourier-like basis because it yields a decomposition corresponding to a tiling of the time-frequency plane with fixed resolution time-frequency cells in a manner analogous to that of a short-time windowed Fourier transform.

We begin with a description of search algorithms appropriate for additive costs $\mathcal{C}^{\text{add}}$. To exploit the algorithmic modularity enabled by the independence of such additive costs, it is necessary to build two trees for each packet table $\mathbf{P}$: the additive information cost tree $\mathbf{C}^{\text{add}}$ and the basis selection tree $\mathbf{S}$. In WavBox 4, the functions *dpt2ict* and *ict2bst* perform these mappings from Discrete Packet Table to Information Cost Tree and from Information Cost Tree to Basis Selection Tree, respectively, as

$$\begin{aligned} \mathbf{C}^{\text{add}} &= \text{dpt2ict}(\mathbf{P}, \mathcal{C}^{\text{add}}) \\ \mathbf{S} &= \text{ict2bst}(\mathbf{C}^{\text{add}}, \mathcal{S}) \end{aligned}$$

with the notational convention that cost functions $\mathcal{C}$ and selection methods $\mathcal{S}$ are denoted in script font while cost trees $\mathbf{C}$ and selection trees $\mathbf{S}$ are denoted in bold font. This modularity permits 1) the output of various cost trees $\mathbf{C}^{\text{add}}$ for the same packet table $\mathbf{P}$ input to *dpt2ict* with various choices of cost functions $\mathcal{C}^{\text{add}}$ as second argument, and 2) the output of various selection trees $\mathbf{S}$ for the same cost tree $\mathbf{C}^{\text{add}}$ input to *ict2bst* with various choices of selection methods $\mathcal{S}$ as second argument.

Now we focus on the bottom-up search $\mathcal{U}$ with additive costs $\mathcal{C}^{\text{add}}$ yielding best bases. With $C_{lb}^{\text{add}} = \mathcal{C}^{\text{add}}(\mathbf{P}_{lb})$ already computed for all l and b, and S_{lb} initialized to 1 for all b on level L and to 0 elsewhere, then the comparison and selection step of the best basis search can be expressed as

```
if $C_{lb}^{\text{add}} \leq C_{l+1,2b}^{\text{add}} + C_{l+1,2b+1}^{\text{add}}$
then $S_{lb} = 1$
else $C_{lb}^{\text{add}} = C_{l+1,2b}^{\text{add}} + C_{l+1,2b+1}^{\text{add}}$
```

and the search is performed breadth-first and bottom-up through the tree. Retaining only the top-most selected branches of $\mathbf{S}$ by resetting any lower selected branches to 0 (*ie.,*pruning descendant lines) yields the best basis selection tree $\mathbf{S}$ with $S_{lb} = 1$ indicating a selected branch.

To obtain the near-best basis search, we perform the same bottom-up search $\mathcal{U}$ with the same sequence of comparisons of basis blocks' information costs as above but we replace the additive cost $\mathcal{C}^{\text{add}}$ with the non-additive cost $\mathcal{C}^{\text{non}}$. This substitution of $\mathcal{C}^{\text{non}}$ for $\mathcal{C}^{\text{add}}$ invalidates the modular independence separating the computation of costs from the selection of bases described above. It is therefore necessary to combine the basis selection with the cost computation. So with $C_{lb}^{\text{non}} = \mathcal{C}^{\text{non}}(\mathbf{P}_{lb})$ already computed for all b on level L, and S_{lb} initialized to 1 for all b on level L and to 0 elsewhere, then the comparison and selection step of the near-best basis search can be expressed as

```
if $\mathcal{C}^{\text{non}}(\mathbf{P}_{lb}) \leq \mathcal{C}^{\text{non}}(\mathbf{P}_{l+1,2b} \oplus \mathbf{P}_{l+1,2b+1})$
then $S_{lb} = 1$
else $\mathbf{P}_{lb} = \mathbf{P}_{l+1,2b} \oplus \mathbf{P}_{l+1,2b+1}$
```

and the search is performed breadth-first and bottom-up through the tree with pruning of descendant lines as described above for the best basis search.

In WavBox 4, the function *dpt2bst* performs this mapping from Discrete Packet Table to Basis Selection Tree as

$$[\mathbf{S}, \mathbf{C}] = \text{dpt2bst}(\mathbf{P}, \mathcal{S}, \mathcal{C})$$

which allows for the greater generality of accepting as input the various search methods $\mathcal{S}$ and cost functions $\mathcal{C}$. The

[3] If $\mathcal{B}^*$ is chosen to be that represented by the Coifman-Wickerhauser search and if $\mathcal{C}$ is chosen to be an additive cost, then the near-best basis is also a best basis.

additional computational cost of *dpt2bst* with $\mathcal{S} = \mathcal{U}$ and $\mathcal{C} = \mathcal{C}^{\text{non}}$ relative to *dpt2ict* and *ict2bst* with $\mathcal{S} = \mathcal{U}$ and $\mathcal{C} = \mathcal{C}^{\text{add}}$ is essentially just the cost of the sorting for those examples ($\mathcal{W}\ell^p$, $\mathcal{N}_f^p$, and $\mathcal{A}^p$) of $\mathcal{C}^{\text{non}}$ which require it as described in Section III-B. Although not detailed here, it is possible to implement this algorithm *without* repeating for the same coefficients the required sorts and powers.

The best and near-best bases as described above are selected by breadth-first bottom-up searches through the table and tree. These searches can be implemented as the additive or non-additive cost comparison and basis selection step inside 1) an inner for-loop for the table blocks and tree branches and 2) an outer for-loop for the levels. Therefore, they have been named bottom-up additive best and non-additive near-best bases with selection method $\mathcal{S} = \mathcal{U}$ to distinguish them from top-down additive near-best and non-additive near-best bases with $\mathcal{S} = \mathcal{D}$. These top-down bases are selected in the opposite direction by depth-first top-down searches with the search terminated as soon as the cost of the children blocks or branches is greater than the cost of the parent block or branch. They can be implemented as the cost comparison and basis selection step within a recursion controlled by a last-in first-out stack. Table I provides a summary of these alternative selection algorithms.

TABLE I

Tree search algorithms for adaptively selecting bases in redundant discrete packet transforms.

Discrete Packet Decomposition	Notation
(Bottom-Up Additive) Best	$\text{DPD}(\mathcal{U}, \mathcal{C}^{\text{add}})$
Bottom-Up Non-Additive Near-Best	$\text{DPD}(\mathcal{U}, \mathcal{C}^{\text{non}})$
Top-Down Additive Near-Best	$\text{DPD}(\mathcal{D}, \mathcal{C}^{\text{add}})$
Top-Down Non-Additive Near-Best	$\text{DPD}(\mathcal{D}, \mathcal{C}^{\text{non}})$

To demonstrate an example with pseudocode, we consider a typical application of the most general purpose function *dpt2bst* which performs the mapping $\mathbf{S} = \text{dpt2bst}(\mathbf{P}, \mathcal{S}, \mathcal{C})$. This function is considered the most general because its implementation makes it valid for all search methods and cost functions. The pseudocode example will also use the function *wpt* for the Wavelet Packet Transform, and *dcnum* for the Data Compression NUMber, which is the mathematical function $\mathcal{N}_f^p(\mathbf{x})$ described above in Section III-B with default values of $p = 2$ and $f = 0.99$ for the parameters. Then, using these functions, the sequence of program statements

```
$\mathbf{P}^{\text{table}} = \text{wpt}(\mathbf{x})$
$\mathbf{S} = \text{dpt2bst}(\mathbf{P}^{\text{table}}, \mathcal{S}, \mathcal{C})$
$\mathbf{P}^{\text{list}} = \text{dpt2dpl}(\mathbf{P}^{\text{table}}, \mathbf{S})$
$M = \text{dcnum}(\mathbf{P}^{\text{list}}(1:N, 1))$
$\mathbf{P}^{\text{list}} = \mathbf{P}^{\text{list}}(1:M, 1:4)$
```

yields a wavelet packet decomposition returned as a packet list truncated to the M largest absolute value packet coefficients constituting 99% of the energy of the transform (and of the original data if the transform mapping is orthonormal). The truncated packet list can now be used for display in tiling plots of the time-frequency plane or in subsequent data processing. While valid for near-best basis decompositions in general, this approach fails to exploit the advantages which can be potentially gained in particular for top-down tree searches with $\mathcal{S} = \mathcal{D}$.

Therefore, we wish to design an appropriate algorithm specialized for top-down tree searches with $\mathcal{S} = \mathcal{D}$ operating via $\mathbf{P}^{\text{basis}}$ instead of the more general algorithm operating via $\mathbf{P}^{\text{table}}$ as described above. Naming this function *wpdd* for Wavelet Packet Decomposition by top-Down search, then the pseudocode segment

```
$[\mathbf{P}^{\text{basis}}, \mathbf{S}] = \text{wpdd}(\mathbf{x})$
$\mathbf{P}^{\text{list}} = \text{dpb2dpl}(\mathbf{P}^{\text{basis}}, \mathbf{S})$
$M = \text{dcnum}(\mathbf{P}^{\text{list}}(1:N, 1))$
$\mathbf{P}^{\text{list}} = \mathbf{P}^{\text{list}}(1:M, 1:4)$
```

incorporating *wpdd* replaces the more general one incorporating *wpt* and *dpt2bst* separately as described above. It is also possible to combine the two functions *dpb2dpl* and *dcnum* so that the M-packet truncated list is returned directly from the combined function. The demonstrated method of first returning the complete N-packet list from the function *dpb2dpl*, returning M from the function *dcnum*, and then truncating the N-packet list to an M-packet list requires significantly more memory. This memory requirement is not necessary and can be eliminated with use of the combined function if packets $\mathbf{P}^{\text{list}}_{M+1}, \ldots, \mathbf{P}^{\text{list}}_{N}$ are never used in subsequent processing. Alternatively, in many practical data processing applications, $\mathbf{P}^{\text{list}}$ would never be generated. Instead, $\mathbf{P}^{\text{basis}}$ would be processed directly by thresholding or other functions.

Since top-down searches do not necessarily examine the entire table and tree, they cannot guarantee finding an optimal basis. However, they enable the possibility of performing the cost computation and basis selection simultaneously with generation of the packet table transform coefficients as described above in the example. Because the algorithm runs unidirectionally downward through the levels of the table and tree, it can be performed essentially "in place", thus significantly reducing memory storage requirements from $O((L+1)N)$ for $\mathbf{P}^{\text{table}}$ in *wpt* and *dpt2bst* to $O(2N)$ for $\mathbf{P}^{\text{basis}}$ and a temporary copy in *wpdd*. Furthermore, because the algorithm does not necessarily require that the entire table and tree be generated and searched, it can be performed with significant savings in machine operations and computing time. This reduction in computational cost corresponds to a number $\hat{L}$ representing the number of levels of the transform that need to be computed.

The number $\hat{L}$ is estimated by summing over all levels the fraction of computed blocks to total blocks on each level. Computed blocks include all ancestral blocks from the root block to the parental blocks above the selected blocks, the selected blocks themselves, and the two children blocks below each selected block (unless the selected block is already at the maximum level L). This estimate yields $\hat{L}$ as a rational (not necessarily integer) number that ranges between 1 and L. Thus, the "in place" algorithm reduces computational costs from approximately $O(LN)$ for $\mathbf{P}^{\text{table}}$ in *wpt* and *dpt2bst* to $O(\hat{L}N)$ for $\mathbf{P}^{\text{basis}}$ in *wpdd*. The amount by which $\hat{L} \leq L$ is dependent on the basis family, selection criterion, and signal data class. However, the reduction in memory storage requirements from $O((L+1)N)$ to $O(2N)$ is independent of these factors. Nevertheless, for both issues of memory storage and computational cost, the savings for higher dimensional signals can be significant even for small values of N and small differences between $\hat{L}$ and L.

The pseudocode example and discussion above have focused on the WPT but of course similar remarks apply to the CPT and other DPTs. In the remainder of this paper, the integrated algorithms *wpdd* and *cpdd* refer to Wavelet and Cosine Packet Decompositions by top-Down search. Analogously, *wpdu* and *cpdu* refer to Wavelet and Cosine Packet Decompositions by bottom-Up search.

V. Time-Frequency Analysis

For the comparison of time-frequency decompositions resulting from various choices of $\mathcal{C}$ with the same adaptive wavelet packet decomposition algorithm *wpdu*, all $\mathbf{P}^{\text{list}}$ were truncated to M packets with $M = \mathcal{N}^2_{0.99}$ different for each $\mathbf{P}^{\text{list}}$. Histograms of these variable-M equal-energy packet lists were then computed for total energy, cells, and blocks per individual level l (results not shown, however, *cf.* [9] for similar results). These histograms demonstrated wide variations in the resulting distributions of energies, cell-numbers, and block-numbers across levels for the different decompositions. Because the distribution of selected blocks across levels of a packet decomposition corresponds to the distribution of sizes and shapes of tiles in a time-frequency tiling plot of cell energies, the observed numerical differences in histograms were also visually apparent in these time-frequency plots.

These differences do impact time-frequency analysis. To demonstrate this effect with an explicit example, a simple test signal of length $N = 512$ was constructed as the sum of four equal energy impulses: two time impulses at 0.2 and 0.8 and two frequency impulses at 0.2 and 0.8 for time and frequency axes both normalized from 0 to 1. This test signal was designed to be symmetric in time and frequency. It was wavelet packet transformed to level $L = 5$ using Daubechies' orthogonal least asymmetric wavelets with filter length 8 [22] and a convolution version with boundary-adjusted wavelets at the ends of the interval [23]. Figures 1 and 2 display the wavelet packet transform and two different adaptive wavelet packet decompositions of the test signal. Both decompositions were selected by bottom-up tree searches with $\mathcal{S} = \mathcal{U}$. However, they differed in the cost function $\mathcal{C}$ used as selection criterion. The additive cost $\mathcal{C}^{\text{add}} = \mathcal{F}$, the $-\ell^2 \ln \ell^2$ functional equivalent to the Coifman-Wickerhauser entropy, selected the root node with 335 coefficients containing 99% of the energy. This decomposition revealed only the time impulses, and not the frequency impulses, in the time-frequency tiling plot. The non-additive cost $\mathcal{C}^{\text{non}} = \mathcal{A}^1$, the data compression area, selected a sub-tree with 21 nodes and 173 coefficients containing 99% of the energy. This decomposition revealed both time and frequency impulses in the time-frequency tiling plot. This example demonstrates that the original "entropy" cost function proposed by Coifman and Wickerhauser may not be appropriate for the time-frequency analysis of all classes of signals.

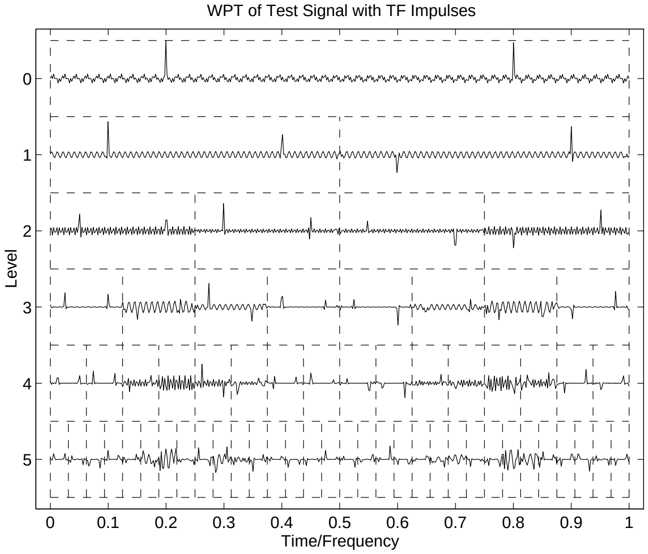

Fig. 1. Wavelet packet transform of test signal with symmetric time-frequency impulses. Horizontal and vertical dashed lines demarcate levels and blocks, respectively.

For a systematic comparison of the time-frequency analyses resulting from decompositions computed with each of the different $\mathcal{S}$ and $\mathcal{C}$, a subjective opinion score was assigned to the visual appearance of the time-frequency tiling plots. Table II lists scores for decompositions obtained from the WPT and CPT computed with the same transform parameters described in Section VI-B and used for the data compression experiments. The subjective opinion scores were determined by qualitative characteristics: 0 for failure to reveal the known test pattern as in the example of CWE in Figure 2, 1 for revealing the known test pattern but with additional interference patterns as in the example of DCA in Figure 2, and 2–4 for revealing the known test pattern without additional interference patterns and with increasingly better joint time-frequency resolution and appearance matched to the known test pattern. For the simple test patterns examined here, this subjective measure was found to be more informative than the objective cross-correlation measure used in [9].

The scores listed in Table II demonstrate that even with decompositions selected with the bottom-up search $\mathcal{S} = \mathcal{U}$, the choice of information cost function $\mathcal{C}$ can dramatically impact the visual appearance of time-frequency tiling plots for a signal with sudden transients but less so for a signal with more even distributions of energies in the time-frequency plane. Furthermore, decompositions selected with the top-down search $\mathcal{S} = \mathcal{D}$ can fail consistently regardless of choice of cost function $\mathcal{C}$ unless the basis library is preselected to match the signal class (*cf.* scores for cosine packets and the CPDD versus wavelet packets and the WPDD in Table II). However, these individual case studies are presented here in the spirit of "counter-examples" in this section for time-frequency analysis. This approach stands in contrast to the population studies and statistical performance results presented in the next section for data compression.

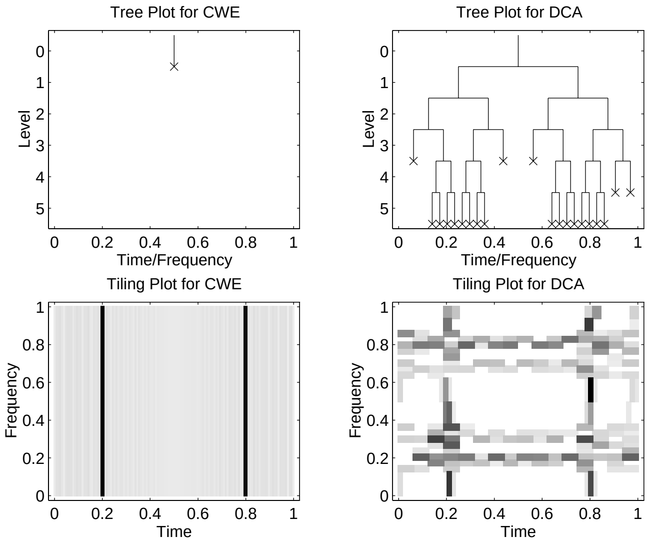

Fig. 2. Adaptive wavelet packet decompositions of symmetric time-frequency impulse test signal: on the left – tree and tiling plots for $\mathcal{C}^{\mathrm{add}} = \mathcal{F}$, the $-\ell^2 \ln \ell^2$ functional equivalent to the Coifman-Wickerhauser entropy (CWE); and on the right – tree and tiling plots for $\mathcal{C}^{\mathrm{non}} = \mathcal{A}^1$, the data compression area (DCA).

TABLE II

Subjective Opinion Scores for Time-Frequency Tiling Plots. A: Symmetric Time-Frequency Impulses; B: Linear Chirp.

$\mathcal{C}$	WPDU		CPDU		WPDD		CPDD	
	A	B	A	B	A	B	A	B
$\mathcal{G}$	1	2	4	3	0	2	0	3
$\mathcal{F}$	0	1	4	4	0	1	0	4
$\mathcal{E}^{0.5}$	1	2	4	3	0	1	0	3
$\mathcal{E}^{1.0}$	2	2	4	4	0	1	0	3
$\mathcal{E}^{1.5}$	3	1	4	4	0	1	0	4
$\mathcal{E}^{3.0}$	0	1	0	4	0	1	0	4
$\mathcal{E}^{4.0}$	0	1	0	4	0	1	0	4
$\mathcal{E}^{5.0}$	0	1	0	4	0	1	0	4
$\mathcal{W}\ell^{0.5}$	1	2	4	3	0	1	4	2
$\mathcal{W}\ell^{1.0}$	3	1	4	4	0	1	0	4
$\mathcal{W}\ell^{2.0}$	1	1	0	2	0	1	1	2
$\mathcal{A}^{0.5}$	1	3	4	2	0	2	4	2
$\mathcal{A}^{1.0}$	1	2	4	3	0	1	4	3
$\mathcal{A}^{2.0}$	3	1	4	4	0	1	0	4
$\mathcal{N}^1_{.900}$	1	3	4	2	0	2	4	2
$\mathcal{N}^1_{.990}$	1	3	2	2	0	2	0	2
$\mathcal{N}^1_{.999}$	1	2	2	2	0	1	0	0
$\mathcal{N}^2_{.900}$	2	2	4	3	0	1	4	3
$\mathcal{N}^2_{.990}$	1	3	4	3	0	2	4	3
$\mathcal{N}^2_{.999}$	2	3	4	2	0	2	0	2
$\mathcal{H}_{\mathrm{S}}(\mathbf{p}_{\mathrm{S}})$	0	2	0	3	0	1	0	3
$\mathcal{H}_{\mathrm{S}}(\mathbf{p}_{\mathrm{D}})$	0	3	4	3	0	2	0	3
$\mathcal{H}_{\mathrm{S}}(\mathbf{p}_{\mathrm{TS}})$	0	2	0	2	0	1	0	3
$\mathcal{H}_{\mathrm{CW}}$	0	1	4	4	0	1	0	4

VI. Data Compression

For experiments investigating lossy compression of data, we wish to minimize the distortion D resulting between the reconstructed estimate $\hat{\mathbf{x}}$ and the original signal $\mathbf{x}$ following compression and coding of the discrete packet decomposition $\mathbf{P}^{\mathrm{list}}$. Compression can be achieved by truncating the N packets in the list to the $M < N$ largest absolute-value packets and then quantizing and coding the remaining M packets. Standard methods of coding data include scalar and vector quantization [24], [25]. The quantization and coding of the M packets remaining after truncation of the list applies only to the amplitudes a and not to the level-l, block-b, and cell-c indices which must be coded without loss of information. Since the (l, b, c)-index information is retained for each packet retained in this compression scheme, it is possible to consider other coding schemes for the packet amplitudes a such as the parameterized-model coding proposed by Taswell [10], [13]. Alternatively, a quantization coder can be applied directly to $\mathbf{P}^{\mathrm{basis}}$ instead of $\mathbf{P}^{\mathrm{list}}$. However, in this case, the order of the coefficients must be maintained since there is no (l, b, c)-index information retained as side information. It is this approach that is used in the following data compression experiments on speech signals from the TIMIT speech corpus. This database of continuous speech was originally developed by Texas Instruments (TI), Massachusetts Institute of Technology (MIT), and SRI International [14].

A. Quantization Coder

A uniform mid-tread quantizer with adaptive feed-forward gain control [26] was modified to include some of the features of the wavelet scalar quantizer (WSQ) characteristic of Bradley and Brislawn [27]. Using their notation, let Z_k be the bin width of the zero bin for the k^{th} subband, and Q_k be the uniform bin width of all other bins for the k^{th} subband. Then quantization encoding of the k^{th} subband amplitudes $a_k(n)$ returns the integer codes

$$p_k(n) = \begin{cases} \left\lceil \frac{a_k(n)+Z_k/2}{Q_k} \right\rceil - 1 & a_k(n) < -Z_k/2 \\ \left\lfloor \frac{a_k(n)-Z_k/2}{Q_k} \right\rfloor + 1 & a_k(n) > Z_k/2 \\ 0 & -Z_k/2 \le a_k(n) \le Z_k/2 \end{cases}$$

and decoding returns the amplitude estimates

$$\hat{a}_k(n) = \begin{cases} (p_k(n) + C)Q_k - Z_k/2 & p_k(n) < 0 \\ (p_k(n) - C)Q_k + Z_k/2 & p_k(n) > 0 \\ 0 & p_k(n) = 0 \end{cases}$$

where $0 < C < 1$ is a parameter that determines the location of the reconstruction estimates within the bins. For $C = 0.5$, these values correspond to the bins' midpoints.

The Bradley-Brislawn WSQ characteristic was simplified to the case of Q_k and Z_k constant for all subbands k, thus enabling use of the same values of Q and Z for all N transform coefficients considered together as one collection instead of as separate subbands. These bin widths Q and Z were adaptively computed as a function of the maximum absolute value amplitude A, the bit rate parameter β in bits per quantized coefficient, and a thresholding parameter α as a fractional multiplier. Thus,

$$\begin{aligned} A &= \max_n |a(n)| \\ Z &= 2\alpha A \\ Q &= (1-\alpha)A/(2^{\beta-1}-1+C) \end{aligned}$$

provided a quantizer characteristic that thresheld values smaller than the fraction α of the maximum A and appropriately "centered" the maximum A in its bin so as to minimize its error. In this specification of the quantizer, the important parameters are α and β, of which α determines the resulting number M of surviving non-zero transform coefficients in each segment of length N, and β determines the precision of the M surviving coefficients. Alternatively, M can be used as the parameter which determines the zero bin width Z either directly as

$$Z = 2|a(n_M)| - \epsilon$$

or indirectly via the fractional multiplier

$$\alpha = |a(n_M)/a(n_1)| - \epsilon = |a(n_M)|/A - \epsilon$$

where the index n_i identifies the i^{th} largest of the coefficients $\{|a(n)| : 1 \le n \le N\}$ sorted in decreasing absolute value order, and ϵ is a tolerance taken as a small multiple of machine precision.

B. Rate-Distortion Curves

Speech signals (each an entire spoken sentence of several seconds duration sampled at 16 kHz as 12 bit integers) were scaled to zero-mean unit-variance signals in floating point format and then segmented with a frame length of $N = 512$ samples. There were approximately $O(10^2)$ segments $\mathbf{x}$ per spoken sentence, with each segment containing sampled data from a time interval of 32 milliseconds of speech. Defining the peak signal data value X in each segment as

$$X = \max_n |x(n)|$$

(analogous to the maximum transform coefficient value $A = \max_n |a(n)|$), the signal-to-noise ratio (SNR) and peak signal-to-noise ratio (PSNR) distortion measures were computed in each segment as

$$\begin{aligned} \text{SNR} &= 10\log_{10}\frac{\|\mathbf{x}\|^2/N}{\|\mathbf{x}-\hat{\mathbf{x}}\|^2/N} = 10\log_{10}\frac{\|\mathbf{x}\|^2}{\|\mathbf{x}-\hat{\mathbf{x}}\|^2} \\ \text{PSNR} &= 10\log_{10}\frac{X^2}{\|\mathbf{x}-\hat{\mathbf{x}}\|^2/N} = 10\log_{10}\frac{NX^2}{\|\mathbf{x}-\hat{\mathbf{x}}\|^2} \end{aligned}$$

in decibels. These distortion measures were computed for each segment of all signals in an experiment, averaged over all segments from all signals, and reported as the mean segmental values with standard errors of the means (SEM). Rate-distortion curves were then plotted as the mean segmental SNR or PSNR versus the mean segmental number M of transform coefficients that were quantized in the non-zero bins of width Q. (In each segment, there were $N - M$ coefficients discarded in the zero bin of width Z.)

The number M of surviving non-zero coefficients was determined by the quantization coder as a function of the thresholding parameter α. For the comparison of different transforms, several kinds of experiments were performed. In the first kind called a "fixed α" experiment, the value of α was held fixed at the same constant value for all transforms and all segments with resulting variable M different for each transform and each segment. In the second kind called a "fixed M" experiment, the value of M was held fixed at the same constant value for all transforms and all segments with resulting variable α different for each transform and each segment. In the third kind called a "fixed f" experiment, the value of M was used to determine α as in the fixed M experiment, but instead of holding M fixed, M itself was determined by the data compression number function $\mathcal{N}_f^p$ with $p = 2$ and the fraction f held fixed for all transforms and all segments. Thus, in the fixed f experiment, the energy of the surviving non-zero transform coefficients was constant for all transforms and segments.

Experiments were performed comparing operational rate-distortion curves for adaptive pulse code modulation (APCM), the discrete wavelet transform (DWT) also known as the fast wavelet transform, the wavelet packet transform (WPT), and the cosine packet transform (CPT). All transforms were computed to depth level $L = 6$. Wavelet and wavelet packet transforms were computed with circular-periodized wavelets [28] of order 7 (with length 14) derived from Daubechies' orthogonal least asymmetric family [22]. These transform parameters of depth level, filter type, filter order, and convolution version were found to be optimal for the DWT as determined by preliminary experiments. Cosine packet transforms were computed with Wickerhauser's symmetric sine bell of order 1 and the type-iv discrete cosine transform [12]. Again as determined by preliminary experiments and as expected by theory, higher-order bells (with increasing approximation to a rectangular window function) performed worse than the smooth bell of order 1. All experiments were done in MATLAB 4.2c.1 running on an ALR Evolution V computer with a 60 MHz Pentium processor.

C. Experimental Results

Experiment 1 was performed as a fixed α experiment on a total of 1385 segments from 20 sentences of type `sx` from 4 different TIMIT speakers consisting of one male and one female speaker from each of two dialect regions. Figures 3 and 4 display results from this experiment in which both α and β were varied over the Cartesian grid (α, β) with α = .128, .064, .032, .016, .008, .004, .002, .001, 0 and β = 4, 6, 8, 10, 12, 14, 16. This approach produced the family of operational rate-distortion curves shown for each

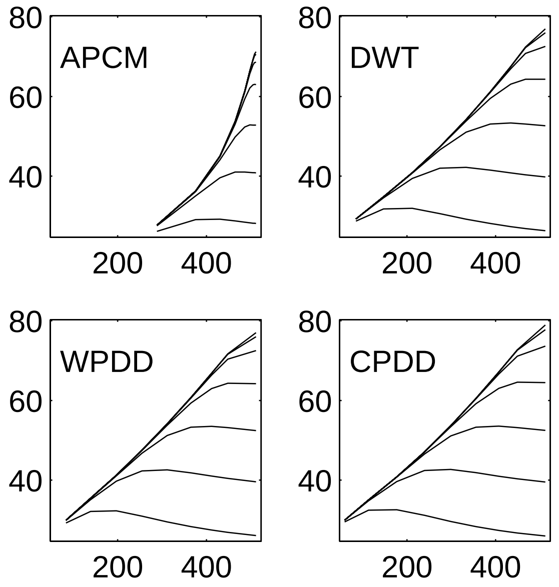

Fig. 3. Mean segmental PSNR versus M for a fixed α experiment. Each panel displays a family of 7 curves; each curve corresponds to a fixed value of β ordered from bottom to top as $\beta = 4, 6, 8, \ldots, 16$; there are 9 points along each curve corresponding to the values of α ordered from left to right as $\alpha = .128, .064, \ldots, .002, .001, 0$.

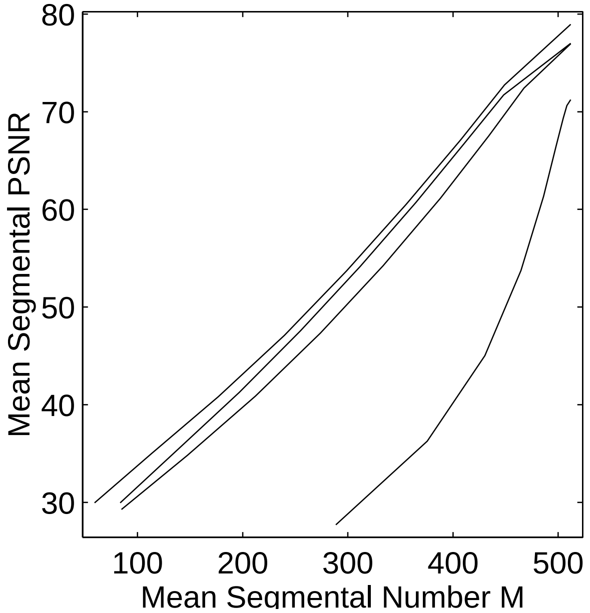

Fig. 4. Mean segmental PSNR versus M for a fixed α experiment. Curves, all with $\beta = 16$, are ordered from upper left to lower right as CPDD, WPDD, DWT, and APCM. Points on curves do not have the same abscissa: M was determined from fixed α with values $\alpha = .128, .064, \ldots, .002, .001, 0$.

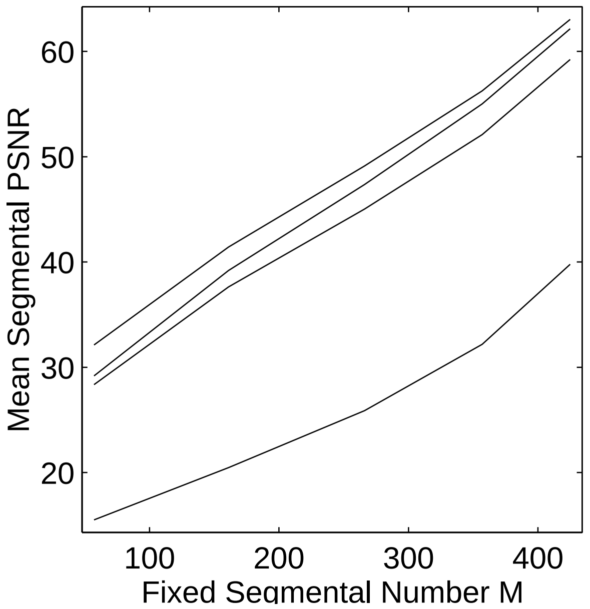

Fig. 5. Mean segmental PSNR versus M for a fixed M experiment. Curves, all with $\beta = 12$, are ordered from upper left to lower right as CPDD, WPDD, DWT, and APCM. Points on curves have the same abscissa: α was determined from fixed M with values $M = 57, 161, 266, 357, 425$.

transform in the panels of Figure 3 and shown for all the transforms together in the same panel in Figure 4. Both CPDD and WPDD were computed as the DPD$(\mathcal{D}, \mathcal{G})$ of the CPT and WPT, respectively. They both outperformed the DWT by several dB higher PSNR with less distortion for given rates of compression measured by the number M of surviving coefficients.

Experiment 2 was performed as a fixed M experiment on a total of 6968 segments from 80 sentences of type `sx` from 16 different TIMIT speakers consisting of one male and one female speaker from each of eight dialect regions. Figure 5 displays results from this experiment with $\beta = 12$ and $M = 57$, 161, 266, 357, 425. As in Experiment 1, both CPDD and WPDD were computed with the DPD$(\mathcal{D}, \mathcal{G})$ of the CPT and WPT, respectively. Again, both of the top-down near-best basis decompositions, CPDD and WPDD, outperformed the DWT, with the CPDD surpassing the WPDD. Table III presents interpolated values of PSNR in dB obtained by cubic spline interpolation at specified values of M. The CPDD and WPDD provided, respectively, as much as a 4 dB and 3 dB gain in PSNR values relative to those for the DWT. Table IV presents the inverse picture with values of M interpolated for given values of PSNR. From this perspective, the CPDD and WPDD provided as much as a 42% and 12% improvement in compression rate over that for the DWT as measured by the number M of coefficients quantized in each segment of length $N = 512$.

Experiment 3 was performed as a fixed f experiment on the same speech data examined in Experiment 2. Table V lists the values of the mean segmental data compression number $\mathcal{N}_f^2$ for the different transforms for fixed f with values of $f = 0.9$, 0.99, 0.999, 0.9999, 0.9999. By

TABLE III
Interpolated PSNR for Given M in a Fixed M Experiment.

M	APCM	DWT	WPDD	CPDD
64	15.85	29.10	29.97	32.84
96	17.39	32.24	33.34	35.95
128	18.90	35.04	36.36	38.76
160	20.43	37.56	39.11	41.35
192	21.99	39.90	41.68	43.78
224	23.61	42.13	44.14	46.11
256	25.33	44.33	46.58	48.41
288	27.18	46.59	49.08	50.74
320	29.26	48.99	51.71	53.18
352	31.75	51.65	54.55	55.82
384	34.80	54.67	57.66	58.75
416	38.57	58.16	61.11	62.04

TABLE IV
Interpolated M for Given PSNR in a Fixed M Experiment.

PSNR	APCM	DWT	WPDD	CPDD
30	329.4	70.8	63.4	40.8
35	386.9	126.1	111.5	84.2
40	426.4	194.1	171.1	142.5
45	451.7	265.6	235.7	209.2
50	466.5	331.9	299.1	277.7
55	474.4	388.2	356.9	342.1
60	479.2	430.4	406.6	397.5

this measure, the CPDD and WPDD provided better signal energy compaction than did the DWT by as much as 20% and 9%, respectively. Figure 6 displays the operational rate-distortion curves for this fixed f experiment, while Tables VI and VII list interpolated values of PSNR and M analogous to those for Experiment 2. Again, the CPDD and WPDD provided, respectively, as much as a 4.2 and 3.5 dB gain in PSNR and as much as a 38% and 9% improvement in compression number M relative to the values for the DWT.

In all of these experiments, standard errors of means were insignificant compared to the means themselves. In fact, the ratio of standard errors to means was at most 0.015 for both PSNR and M. Furthermore, differences between

TABLE V
Mean Segmental Data Compression Numbers $\mathcal{N}_f^2$.

f	APCM	DWT	WPDD	CPDD
0.9	193.6	71.6	66.2	57.5
0.99	346.6	181.5	165.6	160.7
0.999	430.1	291.5	266.2	269.2
0.9999	473.2	384.5	356.7	367.2
0.99999	494.2	447.1	425.1	436.6

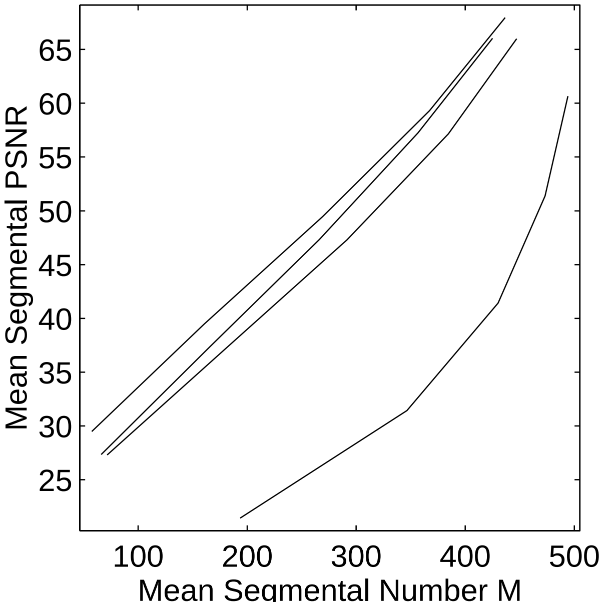

Fig. 6. Mean segmental PSNR versus M for a fixed f experiment. Curves, all with $\beta = 12$, are ordered from upper left to lower right as CPDD, WPDD, DWT, and APCM. Points on curves do not have the same abscissa: α was determined from variable $M = \mathcal{N}_f^2$ with fixed $f = 0.9, 0.99, 0.999, 0.9999, 0.9999$.

TABLE VI
Interpolated PSNR for Given M in a Fixed f Experiment.

M	APCM	DWT	WPDD	CPDD
64	7.32	26.58	27.10	30.15
96	12.10	29.60	30.42	33.34
128	15.89	32.53	33.63	36.42
160	18.89	35.39	36.79	39.42
192	21.32	38.24	39.92	42.36
224	23.36	41.09	43.08	45.29
256	25.24	43.98	46.29	48.24
288	27.15	46.96	49.59	51.23
320	29.31	50.06	53.05	54.34
352	31.91	53.39	56.70	57.64
384	35.17	57.09	60.59	61.23
416	39.28	61.29	64.79	65.18

TABLE VII
Interpolated M for Given PSNR in a Fixed f Experiment.

PSNR	APCM	DWT	WPDD	CPDD
30	329.7	100.2	91.8	62.5
35	382.8	155.5	141.8	112.8
40	421.2	211.9	192.8	166.3
45	449.0	267.2	243.3	221.0
50	468.8	319.1	291.8	274.9
55	482.8	366.1	337.3	326.2
60	493.0	407.1	379.4	373.4

TABLE VIII

Mean Segmental PSNR in a Fixed M Experiment for WPDD.

$\mathcal{C}$	PSNR in dB at $M =$					$\hat{L}$
	57	161	266	357	425	
$\mathcal{G}$	29.4	39.4	47.3	54.8	61.8	3.80
$\mathcal{F}$	30.2	40.0	47.6	54.7	61.5	3.79
$\mathcal{E}^{0.5}$	29.9	39.8	47.8	55.1	61.9	3.90
$\mathcal{E}^{1.0}$	30.2	40.0	47.8	55.1	61.8	3.87
$\mathcal{E}^{1.5}$	30.3	40.1	47.8	54.9	61.7	3.83
$\mathcal{E}^{3.0}$	30.1	39.8	47.3	54.4	61.2	3.71
$\mathcal{E}^{4.0}$	29.9	39.6	47.1	54.2	61.0	3.64
$\mathcal{E}^{5.0}$	29.7	39.4	47.0	54.0	60.8	3.59
$\mathcal{W}\ell^{0.5}$	29.7	39.6	47.6	54.9	61.5	3.96
$\mathcal{W}\ell^{1.0}$	29.8	39.6	47.4	54.6	61.2	3.90
$\mathcal{W}\ell^{2.0}$	18.0	24.2	31.0	37.9	45.4	2.01
$\mathcal{A}^{0.5}$	29.1	39.0	47.0	54.5	61.6	3.87
$\mathcal{A}^{1.0}$	29.7	39.6	47.6	55.0	61.8	4.01
$\mathcal{A}^{2.0}$	30.2	40.1	47.9	55.1	61.7	3.92
$\mathcal{N}^{1}_{.900}$	28.7	38.6	46.6	54.1	61.1	3.67
$\mathcal{N}^{1}_{.990}$	27.4	37.4	45.4	52.9	60.1	3.27
$\mathcal{N}^{1}_{.999}$	26.1	36.2	44.2	51.7	58.8	2.97
$\mathcal{N}^{2}_{.900}$	29.6	39.5	47.3	54.5	61.2	3.74
$\mathcal{N}^{2}_{.990}$	29.0	39.0	47.0	54.4	61.3	3.72
$\mathcal{N}^{2}_{.999}$	28.2	38.3	46.3	53.7	60.9	3.51
$\mathcal{H}_{\mathrm{S}}(\mathbf{p}_{\mathrm{S}})$	28.9	38.5	46.1	53.1	60.0	3.45
$\mathcal{H}_{\mathrm{S}}(\mathbf{p}_{\mathrm{D}})$	28.6	38.1	45.7	52.7	59.7	3.38
$\mathcal{H}_{\mathrm{S}}(\mathbf{p}_{\mathrm{TS}})$	28.8	38.5	46.1	53.1	60.0	3.46
$\mathcal{H}_{\mathrm{CW}}$	30.2	40.0	47.6	54.7	61.5	3.79

TABLE IX

Mean Segmental PSNR in a Fixed M Experiment for CPDD.

$\mathcal{C}$	PSNR in dB at $M =$					$\hat{L}$
	57	161	266	357	425	
$\mathcal{G}$	31.8	41.1	48.8	55.8	62.5	2.36
$\mathcal{F}$	33.0	41.5	48.2	54.6	61.1	1.89
$\mathcal{E}^{0.5}$	32.7	41.6	48.6	55.2	61.8	2.08
$\mathcal{E}^{1.0}$	32.9	41.5	48.3	54.7	61.2	1.94
$\mathcal{E}^{1.5}$	32.9	41.5	48.3	54.6	61.1	1.92
$\mathcal{E}^{3.0}$	33.0	41.5	48.2	54.5	61.0	1.89
$\mathcal{E}^{4.0}$	33.0	41.4	48.1	54.5	61.1	1.91
$\mathcal{E}^{5.0}$	33.0	41.4	48.1	54.5	61.1	1.92
$\mathcal{W}\ell^{0.5}$	31.9	41.1	48.5	55.1	61.7	2.24
$\mathcal{W}\ell^{1.0}$	32.6	41.2	47.9	54.3	60.8	2.00
$\mathcal{W}\ell^{2.0}$	28.9	37.6	44.5	51.3	58.4	2.25
$\mathcal{A}^{0.5}$	30.2	39.8	47.8	55.0	61.8	2.57
$\mathcal{A}^{1.0}$	31.3	40.7	48.3	55.2	61.9	2.39
$\mathcal{A}^{2.0}$	32.9	41.5	48.2	54.6	61.1	1.95
$\mathcal{N}^{1}_{.900}$	30.8	40.3	47.8	54.7	61.4	2.26
$\mathcal{N}^{1}_{.990}$	30.2	39.7	47.7	54.9	61.8	2.34
$\mathcal{N}^{1}_{.999}$	30.2	39.5	47.4	54.6	61.5	2.27
$\mathcal{N}^{2}_{.900}$	32.6	41.2	47.9	54.3	60.8	1.88
$\mathcal{N}^{2}_{.990}$	32.4	41.1	48.0	54.5	61.1	1.94
$\mathcal{N}^{2}_{.999}$	31.8	41.0	48.2	54.9	61.6	2.06
$\mathcal{H}_{\mathrm{S}}(\mathbf{p}_{\mathrm{S}})$	32.3	40.8	47.5	54.0	60.7	1.96
$\mathcal{H}_{\mathrm{S}}(\mathbf{p}_{\mathrm{D}})$	32.1	40.6	47.4	53.8	60.6	1.96
$\mathcal{H}_{\mathrm{S}}(\mathbf{p}_{\mathrm{TS}})$	32.3	40.8	47.5	54.0	60.7	1.95
$\mathcal{H}_{\mathrm{CW}}$	33.0	41.5	48.2	54.6	61.1	1.89

the means were statistically significant since they were typically 20–40 fold larger in magnitude than the standard errors. With regard to the mean segmental values of $\hat{L}$ for the speech data used in Experiments 2 and 3, CPDD provided significant computational savings relative to WPDD as evidenced by the respective values of $\hat{L} = 2.30 \pm 0.01$ for CPDD and $\hat{L} = 3.85 \pm 0.01$ for WPDD.

Experiment 4 was performed as a fixed M experiment on the same speech data used in Experiment 1. Tables VIII and IX list PSNR values for the same values of M used in Experiment 2. SEM values for all PSNR values listed in the tables were approximately 0.3 dB. Both CPDD and WPDD were computed with the DPD$(\mathcal{D}, \mathcal{C})$ of the CPT and WPT, respectively, with the information cost function $\mathcal{C}$ taking the values of basis selection criteria listed in the tables. For the WPDD, the additive cost $\mathcal{E}^{1.0}$ and non-additive cost $\mathcal{A}^{2.0}$ performed better than the others over the entire range of data compression rates considered. Analogously, for the CPDD, the additive cost $\mathcal{E}^{0.5}$ and non-additive cost $\mathcal{A}^{2.0}$ performed better than the others when considering the entire range of rates M. However, $\mathcal{G}$ clearly outperformed the others at lower rates of compression (higher values of M). For the WPDD, the corresponding values of $\hat{L}$ for $\mathcal{E}^{1.0}$ and $\mathcal{A}^{2.0}$ selected decompositions were 3.87 and 3.92, whereas for the CPDD, the values of $\hat{L}$ for $\mathcal{E}^{0.5}$ and $\mathcal{A}^{2.0}$ selected decompositions were 2.08 and 1.95, all with SEM values of approximately 0.03.

Finally, Experiment 5 was performed as a fixed M experiment on the same speech data used in Experiments 2 and 3. Tables X and XI list interpolated values of PSNR and M analogous to those in Tables III and IV for Experiment 2. Using the selection criterion $\mathcal{C} = \mathcal{G}$, WPDU outperformed WPDD at all compression rates, however, CPDD outperformed CPDU at high compression rates. This superior performance for a top-down near-best basis over a bottom-up best basis occurred at high compression rates (low values of M) with the cost $\mathcal{G}$ known to perform better at low compression rates (high values of M). In contrast, CPDU did outperform CPDD at all compression rates with the cost $\mathcal{E}^1$ known to perform well at all compression rates (*cf.* Table IX). Since SEM values were typically 0.1 dB for the PSNR values, the differences in results between search methods $\mathcal{S}$ and cost functions $\mathcal{C}$ were not as dramatic as the differences between transforms. For example, from Tables III and X at $M = 64$, PSNR was 33.9 and 34.2 for CPD$(\mathcal{D}, \mathcal{E}^1)$ and CPD$(\mathcal{U}, \mathcal{E}^1)$ compared to 29.1 for DWT; while from Tables IV and XI at PSNR = 30, M was 35.5 and 33.9 for CPD$(\mathcal{D}, \mathcal{E}^1)$ and CPD$(\mathcal{U}, \mathcal{E}^1)$ compared to 70.8 for DWT. Thus the additional improvement provided by the CPD with bottom-up best basis relative to the CPD with top-down near-best basis was negligible in comparison to the improvement already attained by the

TABLE X
Interpolated PSNR for Given M in a Fixed M Experiment with Search $\mathcal{S}=\mathcal{D}$ Top-Down or $\mathcal{S}=\mathcal{U}$ Bottom-Up.

	WPD($\mathcal{S},\mathcal{G}$)		CPD($\mathcal{S},\mathcal{G}$)		CPD($\mathcal{S},\mathcal{E}^1$)	
M	$\mathcal{D}$	$\mathcal{U}$	$\mathcal{D}$	$\mathcal{U}$	$\mathcal{D}$	$\mathcal{U}$
64	30.0	31.3	32.8	32.4	33.9	34.2
96	33.3	34.7	36.0	35.6	36.8	37.2
128	36.4	37.7	38.8	38.6	39.4	39.8
160	39.1	40.4	41.4	41.3	41.7	42.2
192	41.7	43.0	43.8	43.8	43.9	44.5
224	44.1	45.6	46.1	46.3	46.0	46.6
256	46.6	48.1	48.4	48.7	48.1	48.7
288	49.1	50.6	50.7	51.2	50.2	50.9
320	51.7	53.3	53.2	53.8	52.4	53.1
352	54.6	56.2	55.8	56.6	54.8	55.6
384	57.7	59.4	58.8	59.6	57.5	58.4
416	61.1	62.8	62.0	63.0	60.7	61.6

TABLE XI
Interpolated M for Given PSNR in a Fixed M Experiment with Search $\mathcal{S}=\mathcal{D}$ Top-Down or $\mathcal{S}=\mathcal{U}$ Bottom-Up.

	WPD($\mathcal{S},\mathcal{G}$)		CPD($\mathcal{S},\mathcal{G}$)		CPD($\mathcal{S},\mathcal{E}^1$)	
PSNR	$\mathcal{D}$	$\mathcal{U}$	$\mathcal{D}$	$\mathcal{U}$	$\mathcal{D}$	$\mathcal{U}$
30	63.4	53.0	40.8	43.8	35.5	33.9
35	111.5	97.8	84.2	88.0	74.1	70.8
40	171.1	154.6	142.5	144.3	135.4	129.5
45	235.7	217.4	209.2	207.5	209.1	200.7
50	299.1	280.3	277.7	272.3	285.2	275.2
55	356.9	338.7	342.1	333.8	354.8	344.4
60	406.6	390.5	397.5	388.5	410.3	401.9

CPD over the DWT. Moreover, the mean segmental value of $\hat{L}=1.94$ for CPD($\mathcal{D},\mathcal{E}^1$) was significantly smaller than the fixed value of $L=6$ required for CPD($\mathcal{U},\mathcal{E}^1$). The significance of the differences between values of the statistic $\hat{L}$ used to estimate computational complexity was confirmed with actual measurements of both flops and time. Each of the percentile curves displayed in Figures 7 and 8 contains 100 points representing the quantiles estimated from 6968 sample points for flop counts and processing times of the adaptively selected decompositions for the 6968 segments of speech in Experiment 5. The differences between the curves are clearly and dramatically visible. Thus, the small increase in compression rates provided by the bottom-up best basis relative to the top-down near-best basis did not justify the large increase in computational complexity required to obtain that improvement in performance.

VII. Discussion

To select a basis adaptively within a redundant tree-structured wavelet transform, it is necessary to specify a search path through the tree and a decision criterion by which to compare and select branches of the tree. It is

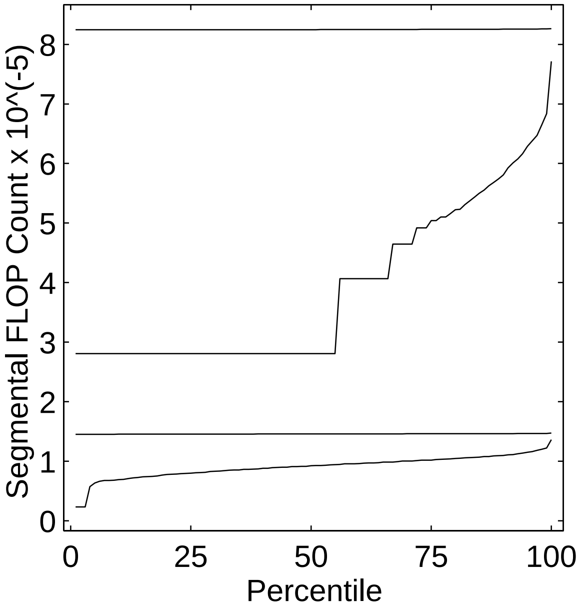

Fig. 7. Percentile curves for segmental flop counts from a fixed M experiment. Number of flops ($\times 10^{-5}$) are counted from start of transform decomposition to finish of basis tree selection. Curves (with median values) are ordered from top to bottom as CPDU (8.25), CPDD (2.81), WPDU (1.46), and WPDD (0.92).

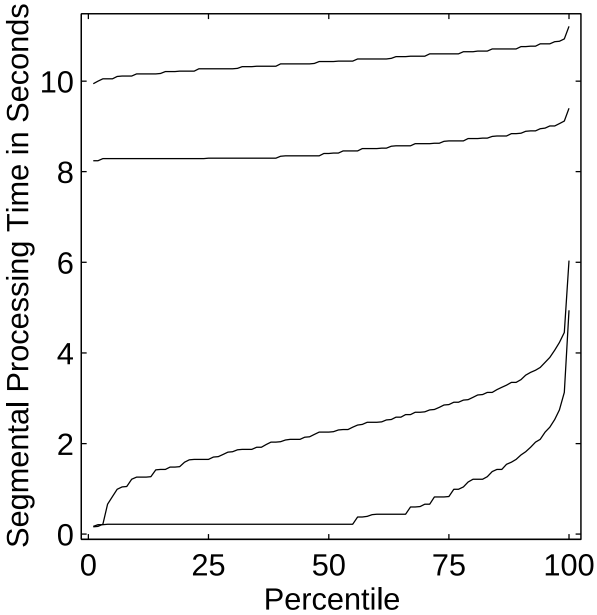

Fig. 8. Percentile curves for segmental processing time from a fixed M experiment. Processing time in seconds is measured from start of transform decomposition to finish of basis tree selection. Curves (with median values) are ordered from top to bottom as WPDU (10.43), CPDU (8.40), WPDD (2.25), and CPDD (0.22).

then possible to implement an appropriate algorithm incorporating the search path and selection criterion. Both Daubechies [29, pages 326–331] and Meyer [7, pages 97–98] discussed the mathematical principles of the "splitting trick" and "splitting algorithm" to generate arbitrary tree-structured wavelet transforms. However, neither advocated a particular search path or decision criterion for selecting a basis within the tree. Moreover, they did not present any results from experiments. Such work was first performed by Coifman and Wickerhauser [2]. They implemented their best basis algorithm by incorporating a bottom-up tree search path with an additive information cost function as selection criterion. Their algorithm can be seen as an application to wavelet packet bases of the theoretical framework developed by Chou et al [30] for optimal pruning of tree-structured systems. The origins of optimal pruning can be traced back to the classification and regression trees of Breiman et al [31]. More recent developments of optimal pruning can be followed forward to the optimal bit allocation of Riskin [32] and the best wavelet packet bases of Ramchandran and Vetterli [33]. The fundamental approach common to these developments, including both the best (minimal entropy) basis of Coifman and Wickerhauser [2] and the best (optimal rate-distortion) basis of Ramchandran and Vetterli [33], is the growth of a larger tree which is then optimally pruned to a smaller subtree.

In this report, I advocate a different approach, one that is sub-optimal rather than optimal. In this approach, the larger tree is never grown and then pruned. Instead the smaller subtree is grown directly. This approach results in what I have called a near-best basis obtained with a top-down search instead of a best basis obtained with a bottom-up search. Advantages gained by this approach are significant. Computational complexity can be reduced with tremendous savings in memory, flops, and time. These savings become especially significant when the methods are extended from 1-D signals to 2-D images [13] and other higher-dimensional signals where the so-called "curse of dimensions" prevails. Moreover, because the constraints imposed by the requirement of additivity for best bases have been eliminated, sub-optimal searches for near-best bases with either additive or non-additive costs can be used more flexibly and extensibly. In particular, they can be used for transforms with bi-orthogonal or non-orthogonal wavelets in addition to orthogonal wavelets (although the stability of iterating wavelets in packet trees must always be considered [34]). In this regard, several new non-additive cost functions have been proposed as decision criteria for use in the basis search algorithms. However, in this more general context of arbitrary cost function as decision criterion for selecting decompositions computed with arbitrary wavelets, the near-best bases should be more appropriately termed near-best frames in the case of non-orthogonal wavelets.

Experiments described in this report demonstrated that the choice of cost function used as selection criterion for finding a basis decomposition may have a significant impact on both time-frequency analysis (*cf.* Section V) and data compression (*cf.* Section VI). Comparing the various cost functions investigated, the ℓ^p and $\ln \ell^2$ functionals (both additive costs) and the data compression area (a non-additive cost) provided the best performance in general. Of note, the $-\ell^2 \ln \ell^2$ functional (an additive cost) used by Coifman and Wickerhauser [2] was unable to resolve the time-frequency components of a signal with artificial transients (*cf.* Section V). Moreover, the complexity and speed of computation of a cost function is dependent on the particular functional and the hardware implementation rather than the classification of the functional as additive or non-additive. For example, computing the additive functional $-\ell^2 \ln \ell^2$ requires squaring all of the coefficients, taking their logarithms, and then adding them, while computing the simple variant of the non-additive functionals $\mathcal{N}_f^p$ and $\mathcal{A}^p$ as $\mathcal{V}_k \equiv v_k(\mathbf{y}, 1)$, where $v_k(\mathbf{y}, p)$ is defined in Section III-B, requires sorting only k of the coefficients and adding them.

However, the choice of search path had an even greater impact on performance than did the choice of cost function. Thus, the sub-optimal top-down tree search, instead of optimal bottom-up tree search, significantly increased the efficiency of computation without decreasing the efficiency of compression of signals from a speech database (*cf.* Section VI). This dramatic increase in computational efficiency applies to memory, flops, and time. If the bottom-up search is implemented for maximal speed, memory storage requires $O((L+1)N)$ locations. If the bottom-up search is implemented for minimal space, memory storage requires $O(2N)$ locations but only with the trade-off of computing some or all coefficients twice [6, page 313]. In contrast, the top-down search is implemented for both maximal speed and minimal space by intentional design requiring at most $O(2N)$ memory locations *without* doubling the computations as in the space-saving bottom-up search. In fact, the top-down search was experimentally observed to require only 1/3 to 2/3 the computations of the non-space-saving bottom-up search. Compare $\hat{L} \approx 2$ and $\hat{L} \approx 4$ respectfully for the cosine and wavelet packet top-down decompositions to the fixed $L = 6$ for the bottom-up decompositions in Tables IV and XI. These results were confirmed by actual flop counts as shown in Figure 7. Moreover, due to various sources of overhead in the MATLAB implementation of the algorithms used in the experiments, they translated into even more dramatic savings in processing time, requiring only 1/40 to 1/4 of the computing time when comparing median values for cosine and wavelet packet top-down and bottom-up decompositions, as shown in Figure 8.

This general conclusion regarding speech data compression was based on comparisons of the mean segmental PSNR value as distortion measure and either the fixed or mean segmental number M of quantized non-zero coefficients as rate measure. Since DeVore et al [35] demonstrated an empirical relationship between the number of coefficients and the number of bytes of compressed data in the context of image compression, use of this number has become a common rate measure for compression studies in the wavelet literature. However, for the development of an actual speech coder, entropy coding in addition to quantization coding, actual bit rates rather than coefficient counts, and psychoacoustic distortion measures rather than PSNR should all be investigated [36], [37], [38].

Nevertheless, the use of the rate-distortion measures studied in this report does not invalidate the following key results and conclusions obtained with regard to data compression. First, for speech compression in particular, the cosine packet transform (which incorporates a discrete cosine transform) performed significantly better than the wavelet packet transform and the fast wavelet transform. This result reconfirms from another perspective the long established superiority of the discrete cosine transform (relative to other transforms) due to its close fit to the optimal Karhunen-Loeve transform [39]. Second, for data compression in general (as inferred from the results on speech compression reported herein), the choice of a decision criterion (information cost function) and basis selection method (tree search path) may very well impact performance and should be considered when designing a data compression method intended for application to a particular class of data and type of compression. For example, a decision criterion and tree search path appropriate for best performance at high rates of compression and distortion may not be appropriate at low rates of compression and distortion, and vice versa. Third, despite the caveat of the second conclusion, sub-optimizing near-best bases can be considered "as good as" or "better than" optimizing best bases in practical situations in which computational complexity must also be considered, especially in real-time signal processing applications. Here, the judgement "as good as" or "better than" depends on whether the perspective emphasizes performance efficiency of compression or computation.

These sub-optimal algorithms selecting near-best bases can be viewed as examples of a general class of algorithms known as satisficing searches. Simon developed his concept of satisficing search [40], [41], [42], [43] within the framework of his theory of bounded rationality [44], [15], [16]. In his earlier annotated collection of papers, Simon wrote two concise statements specifying the meaning of satisficing, one in the context of economic behavior:

> The key to the simplification of the choice process . . . is the replacement of the goal of *maximizing* with the goal of *satisficing*, of finding a course of action that is "good enough." [44, page 204]

and the other in the context of chess-playing programs:

> Again, the key to an effective solution appeared to lie in substituting the goal of satisficing, of finding a good enough move, for the goal of minimaxing, of finding the best move. [44, page 205]

However, it was not until a later paper that he clarified the etymology:

> But if all alternatives are not to be examined, some criterion must be used to determine that an adequate, or satisfactory, one has been found. In the psychological literature, criteria that perform this function in decision processes are called aspiration levels. The Scottish word "satisficing" (= satisfying) has been revived to denote problem solving and decision making that sets an aspiration level, searches until an alternative is found that is satisfactory by the aspiration level criterion, and selects that alternative. [42, page 168]

His most complete mathematical treatment of satisficing searches for solutions to problems represented as trees (directed graphs) can be found in the most recent [43] of his papers treating this topic.[4]

In computational mathematics (and numerical analysis), satisficing methods are apparently not well known or taught as such. However, examples do exist even if they have not been identified as "satisficing" methods. One such example is Gaussian elimination where complete and partial pivoting can be recognized as optimizing and satisficing methods, respectively. Trefethen and Schreiber [45] have shown that partial pivoting is sufficient for the average case, consistent with the problem-solving approach of bounded rationality espoused by Simon. The work of Trefethen and Schreiber provides evidence supporting the conjecture of Golub and Van Loan:

> There appears to be no practical justification for choosing complete pivoting over partial pivoting except in cases where rank determination is an issue. [46, page 119]

In analogy with the conjecture of Golub and Van Loan regarding pivoting for Gaussian elimination, I conclude this report by offering my conjecture regarding adaptive wavelet packet decompositions:

> There appears to be no practical justification for using bottom-up optimizing searches instead of top-down satisficing searches for selecting bases in tree-structured wavelet transforms except for investigation and characterization of unknown signal classes.

In essence, this engineering design advocates using a top-down satisficing search for the average case in a known signal class, and a bottom-up optimizing search for the worst case in an unknown signal class.

Acknowledgments

I thank Professors I. Daubechies, J. B. Keller, S. Mallat, G. Papanicolaou, and M. V. Wickerhauser for helpful discussions.

References

[1] C. Taswell, "Satisficing search algorithms for selecting near-best bases in adaptive tree-structured wavelet transforms," tech. rep., Scientific Computing and Computational Mathematics, Stanford University, Stanford, CA, June 1995. Technical Report SC-CM 95-08.

[2] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Transactions on Information Theory*, vol. 38, pp. 713–718, Mar. 1992.

[3] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3397–3415, Dec. 1993.

[4] G. Davis, S. Mallat, and Z. Zhang, "Adaptive time-frequency approximations with matching pursuits," in *Wavelets Theory, Algorithms, and Applications* (C. K. Chui, L. Montefusco, and L. Puccio, eds.), vol. 5 of *Wavelet Analysis and Its Applications*, pp. 271–293, San Diego, CA: Academic Press, 1994.

[4] The original papers [40], [41], [42], [43] have been reproduced in the collections [44], [15], [16].

[5] R. R. Coifman and M. V. Wickerhauser, "Wavelets and adapted waveform analysis," in *Wavelets: Mathematics and Applications* (J. J. Benedetto and M. W. Frazier, eds.), Studies in Advanced Mathematics, pp. 399–423, Boca Raton: CRC Press, 1994.

[6] M. V. Wickerhauser, *Adapted Wavelet Analysis from Theory to Software.* Wellesley, MA: A K Peters, Ltd., 1994.

[7] Y. Meyer, *Wavelets: Algorithms and Applications.* Philadelphia, PA: Society for Industrial and Applied Mathematics, 1993. Translated by Robert D. Ryan.

[8] N. Saito, "Simultaneous noise suppression and signal compression using a library of orthonormal bases and the minimum description length criterion," in *Wavelets in Geophysics* (E. Foufoula-Georgiou and P. Kumar, eds.), vol. 4 of *Wavelet Analysis and Its Applications*, pp. 299–324, San Diego, CA: Academic Press, 1994.

[9] C. Taswell, "Near-best basis selection algorithms with non-additive information cost functions," in *Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis* (M. G. Amin, ed.), (Philadelphia, PA), pp. 13–16, IEEE Press 94TH8007, Oct. 1994.

[10] C. Taswell, "Top-down and bottom-up tree search algorithms for selecting bases in wavelet packet transforms," in *Wavelets and Statistics* (A. Antoniadis and G. Oppenheim, eds.), vol. 103 of *Lecture Notes in Statistics*, pp. 345–359, Springer Verlag, 1995. Proceedings of the Villard de Lans Conference November 1994.

[11] C. Taswell, *Algorithms for Wavelet Transforms and Adaptive Wavelet Packet Decompositions.* PhD thesis, Scientific Computing and Computational Mathematics, Stanford University, Stanford, CA, Mar. 1995.

[12] M. V. Wickerhauser, "INRIA lectures on wavelet packet algorithms," tech. rep., INRIA, Roquencourt, France, 1991. minicourse lecture notes.

[13] C. Taswell, "Image compression by parameterized-model coding of wavelet packet near-best bases," in *SPIE Conference on Wavelet Applications* (H. Szu, ed.), vol. 2491, pp. 153–161, SPIE Press, Apr. 1995.

[14] DARPA, *TIMIT Acoustic-Phonetic Continuous Speech Corpus.* Gaithersburg, MD: National Institute of Standards and Technology, Oct. 1990. NIST Speech Disc 1-1.1.

[15] H. A. Simon, *Models of Bounded Rationality: Economic Analysis and Public Policy*, vol. 1. New York: John Wiley & Sons, Inc., 1982.

[16] H. A. Simon, *Models of Bounded Rationality: Behavioral Economics and Business Organization*, vol. 2. New York: John Wiley & Sons, Inc., 1982.

[17] J. Berkson, "Relative precision of minimum chi-square and maximum likelihood estimates of regression coefficients," in *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability* (J. Neyman, ed.), pp. 471–479, University of California Press, 1951.

[18] C. Taswell, "WavBox 4: A software toolbox for wavelet transforms and adaptive wavelet packet decompositions," in *Wavelets and Statistics* (A. Antoniadis and G. Oppenheim, eds.), vol. 103 of *Lecture Notes in Statistics*, pp. 361–375, Springer Verlag, 1995. Proceedings of the Villard de Lans Conference November 1994.

[19] D. W. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization.* Wiley Series in Applied Probability and Statistics, New York: John Wiley & Sons, Inc., 1992.

[20] A. I. Khinchin, *Mathematical Foundations of Information Theory.* New York: Dover Publications, Inc., 1957.

[21] D. L. Donoho, "Unconditional bases are optimal bases for data compression and for statistical estimation," *Applied and Computational Harmonic Analysis*, vol. 1, pp. 100–115, Dec. 1993.

[22] I. Daubechies, "Orthonormal bases of compactly supported wavelets: II. variations on a theme," *SIAM Journal on Mathematical Analysis*, vol. 24, pp. 499–519, Mar. 1993.

[23] A. Cohen, I. Daubechies, and P. Vial, "Wavelets on the interval and fast wavelet transforms," *Applied and Computational Harmonic Analysis*, vol. 1, pp. 54–81, Dec. 1993.

[24] R. M. Gray, *Source Coding Theory.* The Kluwer International Series in Engineering and Computer Science: Communications and Information Theory, Boston, MA: Kluwer Academic Publishers, 1990.

[25] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression.* The Kluwer International Series in Engineering and Computer Science: Communications and Information Theory, Boston, MA: Kluwer Academic Publishers, 1992.

[26] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals.* Prentice Hall Signal Processing Series, Englewood Cliffs, NJ: P T R Prentice-Hall, Inc., 1978.

[27] J. N. Bradley and C. M. Brislawn, "The FBI wavelet/scalar quantization standard for gray-scale fingerprint image compression," Technical Report LA-UR-93-1659, Los Alamos National Laboratory, Los Alamos, NM, 1993.

[28] C. Taswell and K. C. McGill, "Algorithm 735: Wavelet transform algorithms for finite-duration discrete-time signals," *ACM Transactions on Mathematical Software*, vol. 20, pp. 398–412, Sept. 1994.

[29] I. Daubechies, *Ten Lectures on Wavelets.* No. 61 in CBMS-NSF Series in Applied Mathematics, Philadelphia, PA: Society for Industrial and Applied Mathematics, 1992.

[30] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Optimal pruning with applications to tree-structured source coding and modeling," *IEEE Transactions on Information Theory*, vol. 35, pp. 299–315, Mar. 1989.

[31] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees.* New York, NY: Chapman and Hall, 1984.

[32] E. A. Riskin, "Optimal bit allocation via the generalized BFOS algorithm," *IEEE Transactions on Information Theory*, vol. 37, pp. 400–402, Mar. 1991.

[33] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Transactions on Image Processing*, vol. 2, pp. 160–175, Apr. 1993.

[34] A. Cohen and I. Daubechies, "On the instability of arbitrary biorthogonal wavelet packets," *SIAM Journal on Mathematical Analysis*, vol. 24, pp. 1340–1354, Sept. 1993.

[35] R. A. DeVore, B. Jawerth, and B. J. Lucier, "Image compression through wavelet transform coding," *IEEE Transactions on Information Theory*, vol. 38, pp. 719–746, Mar. 1992.

[36] P. Noll, "Wideband speech and audio coding," *IEEE Communications Magazine*, pp. 34–44, Nov. 1993.

[37] L. R. Rabiner, "Applications of voice processing to telecommunications," *Proceedings of the IEEE*, vol. 82, pp. 199–228, Feb. 1994.

[38] A. Gersho, "Advances in speech and audio compression," *Proceedings of the IEEE*, vol. 82, pp. 900–918, June 1994.

[39] J. L. Flanagan, M. R. Schroeder, B. S. Atal, R. E. Crochiere, N. S. Jayant, and J. M. Tribolet, "Speech coding," *IEEE Transactions on Communications*, vol. 27, pp. 710–737, Apr. 1979.

[40] H. A. Simon, "A behavioral model of rational choice," *Quarterly Journal of Economics*, vol. 69, pp. 99–118, Feb. 1955.

[41] H. A. Simon, "Rational choice and the structure of the environment," *Psychological Review*, vol. 63, pp. 129–138, Mar. 1956.

[42] H. A. Simon, "Theories of bounded rationality," in *Decision and Organization* (C. B. Radner and R. Radner, eds.), pp. 161–176, Amsterdam: North-Holland Publishing Co., 1972.

[43] H. A. Simon and J. B. Kadane, "Optimal problem-solving search: All-or-none solutions," *Artificial Intelligence*, vol. 6, pp. 235–247, 1975.

[44] H. A. Simon, *Models of Man: Mathematical Essays on Rational Behavior in a Social Setting.* New York: John Wiley & Sons, Inc., 1957.

[45] L. N. Trefethen and R. S. Schreiber, "Average-case stability of gaussian elimination," *SIAM Journal on Matrix Analysis and Applications*, vol. 11, pp. 335–360, July 1990.

[46] G. H. Golub and C. F. V. Loan, *Matrix Computations.* Baltimore: The Johns Hopkins University Press, second ed., 1989.

Carl Taswell was born in Jersey City, New Jersey in 1956. He received the BA degree in biochemistry in 1978 from Harvard University, the MS degree in mathematics in 1984 and MD degree in medicine in 1985 both from New York University, and the MS degree in 1991 and PhD degree in 1995 both in scientific computing and computational mathematics from Stanford University. Since receiving a top ten scholarship award in the 1974 Westinghouse Science Talent Search, he has published several dozen research papers spanning the years 1975–96 and the fields of biochemistry, immunology, medicine, biostatistics, and computational mathematics. His current research interests focus on wavelet transforms with data compression and pattern recognition applications in biomedical signal and image processing.