APPENDICES

APPENDIX

A

WAVELET PACKETS AND WAVELET PACKET TRANSFORMS

A.1 Introduction

A wavelet decomposition or transform simply re-expresses a function in terms of the wavelet basis $\{\psi_{j,k}(t)\}$. This amounts to decomposing the function space L^2 into a direct sum of orthogonal subspaces $\{W_j\}$ and choosing the combination of the orthonormal bases for Ws as the orthonormal basis for L^2. In the case of finite data with information up to a resolution level J, a wavelet transform performs a decomposition of the space V_J into a direct sum of orthogonal subspaces

$$V_J = W_{J-1} \oplus V_{J-1} = W_{J-1} \oplus W_{J-2} \oplus V_{J-2} = \cdots = \bigoplus_{j=0}^{J-1} W_j \oplus V_0 \quad \text{(A.1)}$$

and the union of the bases of these subspaces forms a basis for the wavelet decomposition. This, of course, is by no means the only way to decompose the space L^2 or V_J. In this section we generalize the wavelet decomposition and introduce a whole family of orthonormal bases for function space.

From multiresolution analysis, we know that given the basis functions $\{\phi_{1,k}(t)\}$ of V_1, $\{\phi(t-k)\}$ and $\{\psi(t-k)\}$ constitute an orthonormal basis for V_0 and W_0, respectively, and $V_1 = V_0 \oplus W_0$, where

$$\phi(t) = \sqrt{2} \sum_k h_k \phi(2t-k) \text{ and } \psi(t) = \sqrt{2} \sum_k g_k \phi(2t-k). \quad \text{(A.2)}$$

So the V space can be decomposed into a direct sum of the two orthogonal subspaces defined by their basis functions given by the above two equations. This "splitting trick" or splitting algorithm can be used to decompose W spaces as well. For example, if we analogously define

$$w_2(t) = \sqrt{2} \sum_k h_k \psi(2t-k) \text{ and } w_3(t) = \sqrt{2} \sum_k g_k \psi(2t-k), \tag{A.3}$$

then $\{w_2(t-k)\}$ and $\{w_3(t-k)\}$ are orthonormal basis functions for the two subspaces whose direct sum is W_1. In general, for $n = 0, 1, \cdots$, we define a sequence of functions as follows:

$$w_{2n}(t) = \sqrt{2} \sum_k h_k w_n(2t-k) \tag{A.4}$$

and

$$w_{2n+1}(t) = \sqrt{2} \sum_k g_k w_n(2t-k). \tag{A.5}$$

Clearly, setting $n = 0$, we get $w_0(t) = \phi(t)$, the scaling function, and $n = 1$ yields $w_1(t) = \psi(t)$, the mother wavelet. So far we have been using the combination of $\{\phi(2^j t-k)\}$ and $\{\psi(2^j t-k)\}$ to form a basis for V_J, and now we have a whole sequence of functions $w_n(t)$ at our disposal. Various combinations of these and their dilations and translations can give rise to various bases for the function space. So we have a whole collection of orthonormal bases generated from $\{w_n(t)\}$. We call this collection a "library of wavelet packet bases", and the function of the form $w_{n,j,k} = 2^{j/2} w_n(2^j t-k)$ is called a wavelet packet. Let us call the space formed by the basis $\{w_{n,j,k}(t)\}_k$ $\mathbf{w}_{n,j}$; the following diagram illustrates the decomposition of the space $\mathbf{w}_{0,3}$ (*i.e.*, V_3) using wavelet packets.

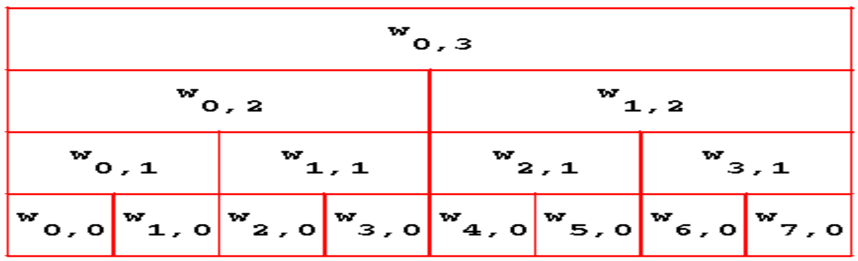

Fig. A.1 Wavelet packet decomposition

Accordingly, a function *x*(*t*) expressed in terms of these orthogonal family of wavelet packets is as follows:

$$x(t) = \sum_{i=-\infty}^{\infty} \sum_{j=1}^{J} a(i,\ j)\theta(t-i) + \sum_{i=-\infty}^{\infty} \sum_{j=1}^{J} d(i,\ j)\theta(t-\ j) \qquad \text{(A.6)}$$

Where *a*(*i*,*j*) are scaling coefficients at *j* scale and *i* delay and *d*(*i*,*j*) are details.

As an example, we look at the wavelet packets generated from the Haar filter. Since the Haar filter has $h_0 = h_1 = 1/\sqrt{2}$, and, using $g_k = (-1)^k h_{1-k}$, $g_0 = -g_1 = 1/\sqrt{2}$, we have

$$w_{2n}(t) = w_n(2t) + w_n(2t-1) \qquad \text{(A.7)}$$

and

$$w_{2n+1}(t) = w_n(2t) - w_n(2t-1) \qquad \text{(A.8)}$$

with $w_0(t)$ the characteristic function on the unit interval.

Using a pair of low-pass and high-pass filters to split a space corresponds to splitting the frequency content of a signal into roughly a low-frequency and a high-frequency component. In wavelet decomposition we leave the high-frequency part alone and keep splitting the low-frequency part. In wavelet packet decomposition, we can choose to split the high-frequency part also into a low-frequency part and a high-frequency part. So in general, wavelet packet decomposition divides the frequency space into various parts and allows better frequency localization of signals.

The transformation of data into wavelet packet basis presents no extra difficulties. We can simply do a convolution using filters h and g on the details $\{d_k^j\}$ as well as on the trend $\{f_k^j\}$. As in the wavelet transform, we can keep doing the decomposition until we cannot go any further. On the other hand, we can also choose not to decompose a particular subspace while decomposing others. So there are many choices for decomposing a signal. We can keep all the coefficients at all decomposition levels and generate a table of coefficients of wavelet packet decomposition.

A.2 Matlab® Wavelet Toolbox

The Wavelet Toolbox contains graphical tools and command line functions that let you examine and explore characteristics of individual wavelet packets Perform wavelet packet analysis of one- and two-dimensional data Use wavelet packets to compress and remove noise from signals and images This chapter takes you step-by-step through examples that teach you how to use the Wavelet Packet 1-D and Wavelet Packet 2-D graphical tools. The last section discusses how to transfer information from the graphical tools into your disk, and back again. Because of the inherent complexity of packing and unpacking complete wavelet packet decomposition tree structures, we recommend using the Wavelet Packet 1-D and Wavelet Packet 2-D graphical tools for performing exploratory analyses. The command line functions are also available and provide the same capabilities. However, it is most efficient to use the command line only for performing batch processing.

A.3 Wavelet Packet

The wavelet packet method is a generalization of wavelet decomposition that offers a richer signal analysis. Wavelet packet atoms are waveforms indexed by three naturally interpreted parameters: position, scale (as in wavelet decomposition), and frequency. For a given orthogonal wavelet function, we generate a library of bases called wavelet packet bases. Each of these bases offers a particular way of coding signals, preserving global energy, and reconstructing exact features. The wavelet packets can be used for numerous expansions of a given signal. We then select the most suitable decomposition of a given signal with respect to an entropy-based criterion.

There exist simple and efficient algorithms for both wavelet packet decomposition and optimal decomposition selection. We can then produce adaptive filtering algorithms with direct applications in optimal signal coding and data compression.

A.4 from Wavelets to Wavelet Packets: Decomposing the Details

In the orthogonal wavelet decomposition procedure, the generic step splits the approximation coefficients into two parts. After splitting we obtain a vector of approximation coefficients and a vector of detail coefficients, both at a coarser scale. The information lost between two successive approximations is captured in the detail coefficients. Then the next step consists of splitting the new approximations coefficient vector; successive details are never reanalyzed.

In the corresponding wavelet packet situation, each detail coefficient vector is also decomposed into two parts using the same approach as in approximation vector splitting. This offers the richest analysis: the complete binary tree is produced as shown in the following figure.

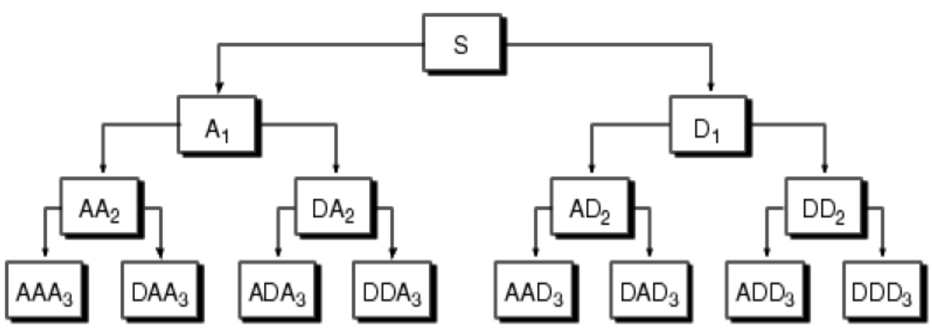

Fig.A.2: Wavelet Packet Decomposition Tree at Level 3

The idea of this decomposition is to start from a scale-oriented decomposition, and then to analyze the obtained signals on frequency sub-bands. The wavelet packet functions and their purpose are shown in the following Tables A.1 - A.4.

Table A.1: Analysis-Decomposition Functions.

Function Name	Purpose
wpcoef	Wavelet packet coefficients
wpdec and wpdec2	Full decomposition
wpsplt	Decompose packet

Table A.2: Synthesis-Reconstruction Functions.

Function Name	Purpose
wprcoef	Reconstruct coefficients
wprec and wprec2	Full reconstruction
wpjoin	Recompose packet

Table A.3: Decomposition Structure Utilities.

Function Name	Purpose
besttree	Find best tree
bestlevt	Find best level tree
entrupd	Update wavelet packets entropy
get	Get WPTREE object fields contents
read	Read values in WPTREE object fields
wenergy	Entropy
wp2wtree	Extract wavelet tree from wavelet packet tree
wpcutree	Cut wavelet packet tree

Table A.4: De-Noising and Compression.

Function Name	Purpose
ddencmp	Default values for de-noising and compression
wpbmpen	Penalized threshold for wavelet packet de-noising
wpdencmp	De-noising and compression using wavelet packets
wpthcoef	Wavelet packets coefficients thresholding
wthrmngr	Threshold settings manager

In the wavelet packet framework, compression and de-noising ideas are exactly the same as those developed in the wavelet framework. The only difference is that wavelet packets offer a more complex and flexible analysis, because in wavelet packet analysis, the details as well as the approximations are split as shown in Fig A.2.

Single wavelet packet decomposition gives a lot of bases from which you can look for the best representation with respect to a design objective. This can be done by finding the "best tree" based on an entropy criterion.

De-noising and compression are interesting applications of wavelet packet analysis. The Wavelet packet de-noising or compression procedure involves four steps:

Step1. Decomposition

For a given wavelet, compute the wavelet packet decomposition of signal x at level N.

Step2. Computation of the Best Tree

For given entropy, compute the optimal wavelet packet tree. Of course, this step is optional. The graphical tools provide a Best Tree button for making this computation quick and easy.

Step3. Thresholding of Wavelet Packet Coefficients

For each packet (except for the approximation), select a threshold and apply thresholding to coefficients. The graphical tools automatically provide an initial threshold based on balancing the amount of compression and retained energy. This threshold is a reasonable first approximation for most cases. However, in general you will have to refine your threshold by trial and error so as to optimize the results to fit your particular analysis and design criteria. The tools facilitate experimentation with different thresholds, and make it easy to alter the tradeoff between amount of compression and retained signal energy.

Step4. Reconstruction

Compute wavelet packet reconstruction based on the original approximation coefficients at level N and the modified coefficients.

References

[1]. Matlab® 7, “Wavelet Packet Toolbox” , 2007.

[2]. Olivier R. et,al. “Wavelet and signal processing” , IEEE signal processing magazine, 1991.

[3]. http:// www.dsprelated.com