FAST COMPUTATION OF THE CONTINUOUS WAVELET TRANSFORM THROUGH OBLIQUE PROJECTIONS

Michael Vrhel, Chulhee Lee, Michael Unser

Biomedical Engineering and Instrumentation Program, NCRR, Bldg. 13, Room 3W13
National Institutes of Health, Bethesda, MD 20892 USA

ABSTRACT

We introduce a fast simple method for computing the real continuous wavelet transform (CWT). The approach achieves $O(N)$ complexity per scale and the filter coefficients can be analytically obtained by a simple integration. Our method is to use P wavelets per octave and to approximate them with their oblique projection onto a space defined by a compact scaling function. The wavelet templates are expanded to larger sizes (octaves) using the two-scale relation and zero padded filtering. Error bounds are presented to justify the use of an oblique projection over an orthogonal one.

1. INTRODUCTION

We define the CWT of the signal $s(t)$ as the inner product

$$W_\psi s(\alpha,\tau)=\frac{1}{\sqrt{\alpha}}\left\langle s(t),\psi\left(\frac{\tau-t}{\alpha}\right)\right\rangle=\frac{1}{\sqrt{\alpha}}\int_{-\infty}^{+\infty}s(t)\psi\left(\frac{\tau-t}{\alpha}\right)dt$$

where α and τ are respectively the continuously varying scaling and shifting parameters, and the real function $\psi(t)$ is the mother wavelet.

Fast algorithms exist for computing the wavelet transform at the dyadic scales when the wavelet is associated with a multi-resolution [1, 2, 5, 7]. Here we are interested in a finer sampling of the scale axis.

Previous methods for computing at non-dyadic scales have either been restricted to a particular scale spacing, wavelet shape, or implemented with $O(N\ \log(N))$ computations per scale [3, 4, 6, 9, 11]. A method which achieved $O(N)$ operations per scale, with no restrictions to the shape of the wavelet, and with arbitrarily fine exponential sampling along the scale axis was described in [12]. That method approximated the wavelet by its orthogonal projection into a space defined by a compact scaling function.

Here, we introduce an extension of this previous approach by using an oblique projection to approximate the wavelet. The main advantages of the new approach are that it is simpler to implement and that the filters used in the implementation are shorter or of a lower order. The present framework is also more general since it includes the algorithm in [12] as a special case. A surprising result is that the increase in speed, flexibility, and simplicity of the non-orthogonal case is achieved with a negligible loss in accuracy.

The algorithm consists of an FIR filter bank and a fast recursive IIR filter. The system is shown in Fig. 1 where the additional FIR filter $[h(k)]_{\uparrow 2^i}$ performs the two scale stretching of the approximating scaling function that enables us to use the same FIR filters in computing the CWT for P scales within each octave.

2. THE APPROACH

To perform the computation, we will construct a set of P auxiliary wavelets $\left\{\psi_i(t)\cong\alpha_i^{-1/2}\,\psi(t/\alpha_i)\right\}_{i=0,\dots P-1}$ which are the oblique projection of the wavelets $\left\{\alpha_i^{-1/2}\,\psi(t/\alpha_i)\right\}_{i=0,\dots P-1}$ into a space defined by the compact scaling function φ_2 orthogonal to the space defined by the analysis function φ_1.

An Lth order scaling function is a function $\varphi(t)\in L_2$ that satisfies the following three conditions:

(i) $0<A\le\sum_{k\in Z}\left|\hat{\varphi}(\omega+2\pi k)\right|^2\le B<+\infty$

(ii) $\hat{\varphi}(0)=1,\ \ \hat{\varphi}^{(m)}(2\pi k)=0,\ k\in Z,\ k\ne 0$ for $m=0,\dots,L-1$

(iii) $\varphi\left(\frac{t}{2}\right)=\sum_{k\in Z}h(k)\varphi(t-k)$

where $\hat{\varphi}(\omega)$ is the Fourier transform of $\varphi(t)$ and $\hat{\varphi}^{(m)}(2\pi k)$ is the mth derivative of $\hat{\varphi}(\omega)$ evaluated at $2\pi k$. Property (i) insures that the subspace $V(\varphi)$ is a well defined subspace of L_2. Property (ii) implies that $\varphi(t)$ reproduces all polynomials of degree $L-1$. Property (iii) is referred to as the two-scale relation and it allows us to dilate all the wavelet filters by a power of two. An analysis function is defined as a function which satisfies only properties (i) and (ii). Criteria for the selection of a scaling or analysis function should be based on its approximating power, smoothness, and support to name a few.

For a pair of analysis functions φ_1 and φ_2, the oblique projection of the wavelet $\psi_{\alpha_i}(t)=\psi(t/\alpha_i)$ into $V(\varphi_2)$ orthogonal to $V(\varphi_1)$ can be expressed as

$$\psi_i(t)=\sum_{k\in Z}p_{\alpha_i}(k)\varphi_2(t-k) \qquad (1)$$

where $p_{\alpha_i} = q_{\alpha_i} * q_{12}$, $q_{\alpha_i}(k) = \langle \psi_{\alpha_i}(t), \varphi_1(t-k) \rangle$ and the digital filter q_{12} is the convolution inverse of the cross correlation sequence $a_{12}(k) = \langle \varphi_1(t-k), \varphi_2(t) \rangle$. Details of the oblique projection operator are contained in [10]. Note that we have an orthogonal projection if $\varphi_1(t) \in V(\varphi_2)$.

In order to achieve $O(N)$ complexity per scale, we will replace the computation of the convolution in the definition of the CWT by the approximation

$$\tilde{W}_\psi s(\alpha_i, \tau) = (s * \psi_i)(\tau). \qquad (2)$$

Substituting (1) into the above, we have the following equations:

$$s_0(t) = (s * \varphi_2)(t) \qquad (3)$$

$$\tilde{W}_\psi s(\alpha_i, \tau) = \sum_{k \in Z} p_{\alpha_i}(k) s_0(\tau - k). \qquad (4)$$

We then make use of the two-scale relation for φ_2 given as property (iii). This allows us to use the same set of filters for each octave.

In practice, we have the sample values $s[k] = s(t)|_{t=k}$. To compute the continuous convolution in (3) we use the approximation $s_0[k] \cong (s * b_2)(k)$ where $b_2(k) = \varphi_2(t)|_{t=k}$.

Incorporating property (iii) into (4), sampling, and performing simple algebraic manipulations, we obtain the following algorithm:

(i) $s_{i+1}(k) = \left(s_i * [h]_{\uparrow 2^i}\right)(k)$

(ii) $\tilde{s}_i(k) = \left(s_i * [q_{12}]_{\uparrow 2^i}\right)(k)$

(iii) $\tilde{W}_\psi s(2^i \alpha_j, k) = \left(\tilde{s}_i * \left[q_{\alpha_j}\right]_{\uparrow 2^i}\right)(k) \qquad j = 0, \ldots, P-1$

where $[h]_{\uparrow 2^i}$ is $h(k)$ with $2^i - 1$ zeros between each sample (i.e. expanded by a factor of 2^i). Step (i) performs the two-scale filtering defined by property (iii) for φ_2; step (ii) is the correction filtering which insures that we have an oblique projection of our wavelet; and step (iii) is the FIR filtering which constructs the approximated wavelets in terms of the scaling function and computes the wavelet coefficients. These equations describe the algorithm as it is shown in Fig. 1.

3. THE APPROXIMATION ERROR

In the previous section, we replaced the convolution of the CWT by its approximation (2). Here we consider the behavior of the approximation error and show how we can design our algorithm to maintain the error below an acceptable threshold. Reduction of the error is achieved by selecting the appropriate functions φ_1 and φ_2, and adjusting the fine scale resolution α_0.

The approximation power of a scaling function depends upon its ability to reproduce polynomials up to a specific degree. This degree plus one is defined as the order of accuracy of the approximating function. The properties of the approximation error for the orthogonal projection case are given by the Strang-Fix conditions [8]. We will first describe the relationship between the oblique and the orthogonal approximation errors, and then translate this relationship to the Strang-Fix conditions.

The oblique projection into $V(\varphi_2)$, orthogonal to $V(\varphi_1)$, is related to the orthogonal projection into $V(\varphi_2)$ by the following: (cf. [10])

$$\|\psi_\alpha - P_2\psi_\alpha\| \le \|\psi_\alpha - P_{2\perp1}\psi_\alpha\| \le \frac{1}{\cos(\theta)}\|\psi_\alpha - P_2\psi_\alpha\|$$

where P_2 is the orthogonal projection operator into $V(\varphi_2)$, $P_{2\perp1}$ is the oblique projection operator into $V(\varphi_2)$ orthogonal to $V(\varphi_1)$, and θ is the largest angle between $V(\varphi_1)$ and $V(\varphi_2)$.

For the Lth order scaling function φ_2 (cf. condition ii), the Strang-Fix theory provides us with the relationship

$$\|\psi_\alpha - P_2\psi_\alpha\| \le C_{\varphi_2}\frac{\|\psi^{(L)}\|}{\alpha^L}$$

where the constant C_{φ_2} is a function of φ_2 and $\|\psi^{(L)}\|$ is the norm of the Lth derivative of ψ. This bound translates to a bound on the oblique projection in the following fashion

$$\|\psi_\alpha - P_{2\perp1}\psi_\alpha\| \le \frac{C_{\varphi_2}}{\cos(\theta)}\frac{\|\psi^{(L)}\|}{\alpha^L}. \qquad (5)$$

In practice, the oblique error is much closer to the orthogonal error than what is indicated by the above worst case bound. In fact, it can be shown that the oblique and orthogonal errors are asymptotically equivalent as $\alpha \to \infty$. Specifically the approximation error behaves as

$$\|\psi_\alpha - P_{2\perp1}\psi_\alpha\| = C_2\frac{\|\psi^{(L)}\|}{\alpha^L} + O\left(\frac{1}{\alpha^{L+1}}\right)$$

where L is again the order of approximation and

$$C_2 = \frac{1}{L!}\left(\sum_{k \ne 0}\left|\hat{\varphi}_2^{(L)}(2\pi k)\right|^2\right)^{1/2}.$$

Figure 2 displays the approximation error curve for the cubic LS approximation of a 2nd derivative Gaussian wavelet and an oblique approximation using the zero and third degree B-splines for φ_1 and φ_2 respectively. The difference between the LS and oblique errors is very small and they approach the same asymptote. In fact, the loss of performance is negligible and the oblique error is much less than the upper bound in (5) .

4. IMPLEMENTATION AND RESULTS

In this section, we will summarize the implementation of the algorithm as a series of steps, each of which are illustrated with an example.

Step 1*:* Select the compactly supported wavelet function for the particular application. For illustrative purposes, we consider the wavelet

$$\psi_{_{mex}}(t) = \begin{cases} K_o\left(1-\left(\dfrac{t}{\alpha_o}\right)^2\right) e^{\frac{-(t/\alpha_o)^2}{2}} - K_1 \; ; & \left|\dfrac{t}{\alpha_o}\right| \le 5 \\ 0 \qquad ; & \text{otherwise} \end{cases}$$

which is a truncated version of the second derivative of a Gaussian.

Step 2*:* Select the scaling function φ_2 and the analysis function φ_1. Here it is advantageous to use B-splines because of their attractive features:

(i) They are the shortest functions for a given order of approximation.

(ii) They are symmetrical.

(iii) They have a simple analytical form which makes them easy to manipulate.

(iv) They are smooth well behaved functions.

To keep things as simple as possible but still maintain a high degree of accuracy, we will use the zero degree B-spline for φ_1 and the cubic B-spline for φ_2. The advantages of this particular selection are:

(i) The FIR filters $\left\{q_{\alpha_i}(k)\right\}_{i=o,\ldots M-1}$ are determined analytically by simple integration of the wavelet $\psi(t/\alpha_i)$:

$$q_{\alpha_i}(k) = \int_{k-1/2}^{k+1/2} \psi(t/\alpha_i)\, dt \qquad (6)$$

(ii) The FIR filters are shorter and the IIR filter is of a lower order as compared to the orthogonal projection into $V(\varphi_2)$.

(iii) The approximation error is very close to the minimum which is achieved by the orthogonal projection.

In this case, we gain in simplicity and speed with little loss in accuracy (cf. error graph in Fig. 2).

Other properties to consider are the approximation power of the function and the number of vanishing moments desired in the projected wavelet. The latter is closely tied to the first and second step and the former is related to the third step.

Step 3*:* Select an acceptable error level, and hence a fine scale resolution α_0. This step should actually be performed in conjunction with the selection of the order of the scaling function since its power of approximation determines the exact error characteristics. For a particular error threshold, one can compute a plot such as that shown in Fig. 2 to determine an order of approximation and a fine scale resolution. We selected an error threshold of 0.01 which is shown as a dash line in Fig. 2. This places our fine scale resolution at 1.41.

Step 4*:* Select the number of scales per octave. This will set the number of FIR filters. We used P=12 scales per octave where $\alpha_i = \alpha_o 2^{i/12} \quad i = 0,\ldots,11$ which corresponds to the musical notes $\left(A, A^{\#}, B, C, C^{\#}, \ldots\right)$.

With these design steps completed, Eq. (6) provides the means for determining the filter coefficients for the FIR filter bank. The remaining FIR filter is the refinement filter $h(k)$. For a B-spline of degree n, $h(k)$ is the binomial filter

$$h(k) = \begin{cases} \dfrac{1}{2^n}\begin{pmatrix} n+1 \\ k+(n+1)/2 \end{pmatrix} ; & |k| \le (n+1)/2 \\ 0 \qquad ; & \text{otherwise.} \end{cases}$$

The IIR filter q_{12} is implemented in a fast recursive fashion. In the B-spline case, q_{12} is the filter $\left(b^{n_1+n_2+1}\right)^{-1}(k)$ where n_1 and n_2 are the degrees of φ_1 and φ_2 respectively. The details of this filter are contained in [12].

We implemented the algorithm in MATLAB. The impulse response of the system over four octaves is shown in Fig. 3. We compared the performance of our algorithm to an FFT-based method which used a radix-2 algorithm when the signal length was a power of 2 and a mixed radix method for other signal lengths (MATLAB's FFT algorithm). The input was an electroencephalograph signal (EEG). Fast algorithms for analyzing such signals are of interest for applications that require real time detection of brain seizures.

The length of the signal was varied and the time required to compute four octaves is shown as a function of the signal length in Fig. 4. The long dash line represents the FFT algorithm when the signal was padded to a power of two. The solid line with the points is the FFT algorithm for various signal lengths. The short dash line, which is the oblique projection algorithm, clearly demonstrates the $O(N)$ characteristics of the method.

5. CONCLUSION

We have introduced a method for the rapid computation of the CWT. The algorithm has the following properties:

• It has a complexity of $O(N)$ per scale, which is the lowest order of complexity.

• The FIR filter coefficients are obtained analytically by integration of the wavelet.

• The approximation error is simple to control by adjusting the fine scale resolution or increasing the order of approximation.

• The method is flexible enough to approximate a variety of wavelet shapes and achieves an arbitrarily fine sampling of the scale axis.

REFERENCES

[1] P. Abry and A. Aldroubi, "Designing multiresolution analysis-type wavelets and their fast algorithms", *Journal of Fourier Analysis and Applications*, To appear.

[2] P. Dutilleux, "An Implementation of the algorithme à trous to compute the wavelet transform", *Proc. Wavelets: Time-Frequency Methods and Phase Space*, 1989, pp. 298-304.

[3] D.L. Jones and R.G. Baraniuk, "Efficient approximation of continuous wavelet transforms", *Elect. Lett.*, Vol. 27, No. 9, 1991, pp. 748-750.

[4] S. Maes, "A Fast Quasi-Continuous Wavelet Transform Algorithm", *Proc. Workshop on Time, Frequency, Wavelets and Multiresolution Theory*, INSA-Lyon, 1994, pp. 31.1-31.4.

[5] S.G. Mallat, "A theory of multiresolution signal decomposition: the wavelet representation", *IEEE Trans. PAMI*, Vol. PAMI-11, No. 7, 1989, pp. 674-693.

[6] O. Rioul and P. Duhamel, "Fast algorithms for discrete and continuous wavelet transforms", *IEEE Trans. Info. Th.*, Vol. IT-38, No. 2, 1992, pp. 569-586.

[7] M.J. Shensa, "The discrete wavelet transform : wedding the à trous and Mallat algorithms", *IEEE Trans. Sig. Proc.*, Vol. 40, No. 10, 1992, pp. 2464-2482.

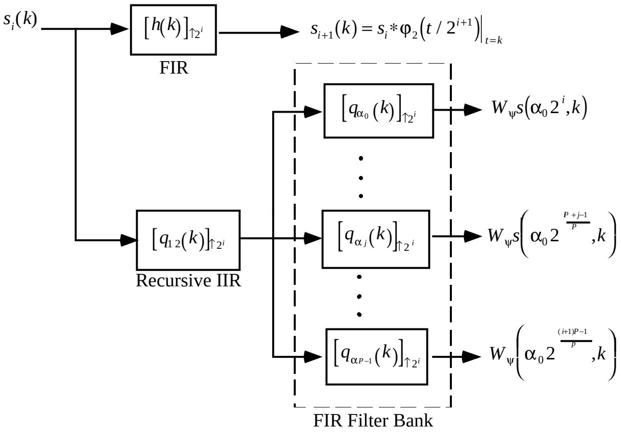
[8] G. Strang and G. Fix, "A Fourier analysis of the finite element variational method", in: *Constructive Aspect of Functional Analysis*, Edizioni Cremonese, Rome, 1971, pp. 796-830.

[9] M. Unser, "Fast Gabor-like windowed Fourier and continuous wavelet transforms", *IEEE Sig. Proc. Lett.*, Vol. 1, No. 5, 1994, pp. 76-79.

[10] M. Unser and A. Aldroubi, "A general sampling theory for non-ideal acquisition devices", *IEEE Trans. Sig. Proc.*, Vol. 42, No. 11, 1994, pp. 2915-2925.

[11] M. Unser, A. Aldroubi and S.J. Schiff, "Fast implementation of the continuous wavelet transform with integer scales", *IEEE Trans. Sig. Proc.*, Vol. 42, No. 12, 1994, pp. 3519-3523.

[12] M.J. Vrhel, C. Lee and M. Unser, "Fast Continuous Wavelet Transform", *Proc. IEEE ICASSP 95*, Detroit, MI, 1995, pp. 1165-1168.



$s_i(k)$
$[h(k)]_{\uparrow 2^i}$
FIR
$s_{i+1}(k) = s_i * \varphi_2(t / 2^{i+1})\big|_{t=k}$
$[q_{\alpha_0}(k)]_{\uparrow 2^i}$
$W_\psi s(\alpha_0 2^i, k)$
$[q_{12}(k)]_{\uparrow 2^i}$
Recursive IIR
$[q_{\alpha_j}(k)]_{\uparrow 2^i}$
$W_\psi s\left(\alpha_0 2^{\frac{P+j-1}{P}}, k\right)$
$[q_{\alpha_{P-1}}(k)]_{\uparrow 2^i}$
$W_\psi\left(\alpha_0 2^{\frac{(i+1)P-1}{P}}, k\right)$
FIR Filter Bank

Fig. 1. System block diagram.

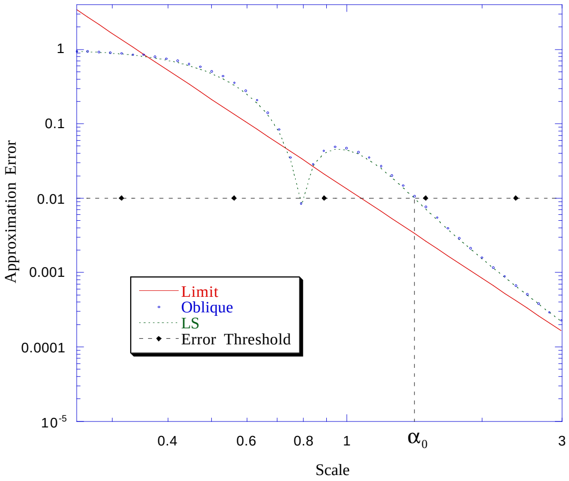


Fig. 2. Approximation errors

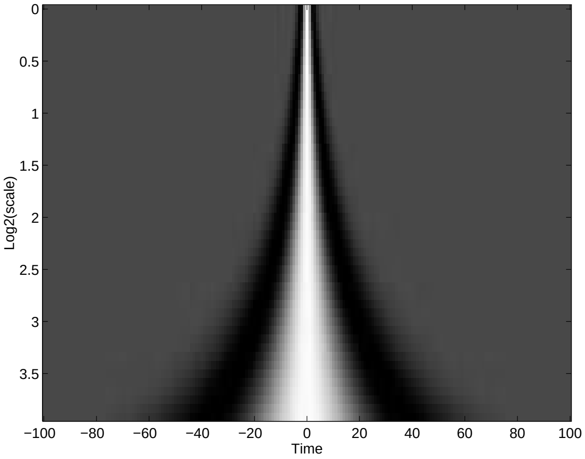


Fig. 3. Impulse response of the system

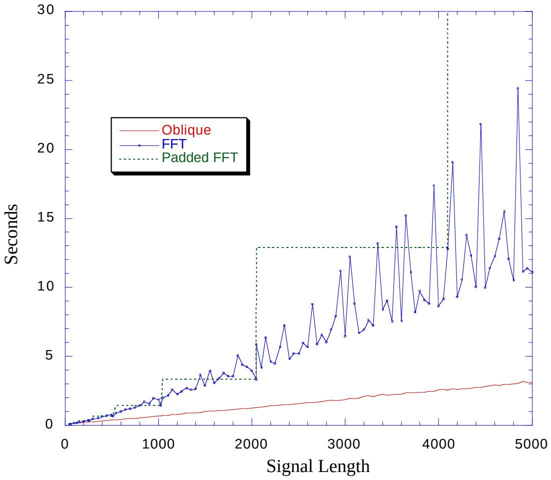


Fig. 4. Speed comparisons