Projection Based Prefiltering for Multiwavelet Transforms

Michael J. Vrhel and Akram Aldroubi

Abstract— **We introduce a method for initializing the multiwavelet decomposition algorithm. The initialization procedure is the orthogonal projection of the input signal into the space defined by the multiscaling function. The approach will always have a solution, places no restrictions on the input (except that it be contained within L_2), and can be implemented in a fast algorithm. We present the details of our approach and compare it with another proposed method of prefiltering.**

I. Introduction

A multiwavelet algorithm provides flexibility in terms of wavelet design over a traditional uniwavelet algorithm allowing simultaneously compactness, orthogonality, symmetry, and regularity in addition to providing superior energy compaction [1]. Mathematically, the multiwavelet decomposition can be expressed as follows. Let the vector

$$\Phi = (\varphi^1(x), \varphi^2(x), \ldots, \varphi^r(x))^T$$

be the multiscaling function that generates the multiresolution spaces

$$V_j = \left\{ \sum_{i=1}^{r} \sum_{k \in Z} c_j^i(k) 2^{-j/2} \varphi^i\left(\frac{x}{2^j} - k\right); c_j^i \in l_2, i = 1, \ldots, r \right\}.$$

The multiwavelet transform decomposes a signal $f(x) \in V_0$ as

$$\begin{aligned} f(x) &= \sum_{i=1}^{r} \sum_{k \in Z} c_0^i(k) \varphi^i(x-k) \\ &= \sum_{i=1}^{r} \sum_{k \in Z} c_{J_0}^i(k) 2^{-J_0/2} \varphi^i\left(\frac{x}{2^{J_0}} - k\right) \\ &\quad + \sum_{i=1}^{r} \sum_{J_0 \le j < 0} \sum_{k \in Z} d_j^i(k) 2^{-j/2} \psi^i\left(\frac{x}{2^j} - k\right) \end{aligned}$$

where $\Psi = (\psi^1(x), \psi^2(x), \ldots, \psi^r(x))^T$ is the multiwavelet associated with the multiresolution spaces V_j [1]–[5].

Much of the work on multiwavelet decompositions has involved the design of the multiscaling function Φ and an associated multiwavelet Ψ. The typical assumption is that the coefficients $c_0^i(k)$ $i = 1, \ldots, r$ are available. In practice, however, we have available only the samples $f[k/r]$, or more realistically, we have the sampling model shown in Fig. 1.

The prefiltering problem involves determining the coefficients $c_0^i(k)$ $i = 1, \ldots, r$ from the samples $f[k/r]$. If it is assumed that the signal $f(x)$ is contained in V_0, then the initialization problem is one of interpolating the function

$$f(x) = \sum_{i=1}^{r} \sum_{k \in Z} c_0^i(k) \varphi^i(x-k). \tag{1}$$

Xia *et al.* [1] provide details on what to do for this case. Since it is necessary to compute r coefficients for each knot, $f(x)$ must be sampled at a rate of $1/r$ (i.e., $f[k/r]$); this will provide a determined system of equations. Expressing (1) in terms of the samples of $f(x)$ provides a system of convolution equations that can be expressed as the matrix-vector convolution

$$\mathbf{s}(k) = (\mathbf{B} * \mathbf{c})(k) \tag{2}$$

where the elements of the vectors and matrix are the sequences

$$\begin{aligned} [\mathbf{s}]_i(k) &= f(k + (i-1)/r), \\ [\mathbf{B}]_{ij}(k) &= \varphi^j(k + (i-1)/r), \\ [\mathbf{c}]_i(k) &= c_0^i(k) \end{aligned} \tag{3}$$

and the matrix-vector convolution operator is defined as

$$[\mathbf{s}]_i(k) := \sum_{l=1}^{l=r} \sum_{h \in Z} [\mathbf{B}]_{il}(h) [\mathbf{c}]_l(k-h). \tag{4}$$

The above matrix convolution operation is simply a mixture of matrix-vector multiplication and sequence convolution.

The prefiltering problem involves the inversion (in the convolutional sense) of the matrix-sequence $\mathbf{B}(k)$. A drawback of this interpolation approach is that the inverse is not guaranteed to exist (an example will be shown later). In addition, the method is inflexible since it requires $f(x) \in V_0$, which implies that if we change our space V_0, then the model of our (unchanged) data changes as well.

Here, we provide a solution to the problem of prefiltering that does not suffer from existence problems and does not require $f(x) \in V_0$. The case of $f(x) \notin V_0$ is more realistic since we have only samples of $f(x)$ that are usually not related to the chosen multiwavelet transform. These samples are, however, related to the impulse response of the acquisition device (cf., Fig. 1), which can provide a model for the signal.

Since we want to decompose the signal in terms of Φ, we would like to find the signal in V_0 that best approximates $f(x)$. For this reason, we approach the prefiltering problem as one of computing the orthogonal projection of the signal into the space defined by the multiscaling function [10]. The attractive features of this approach are that a solution always exists (unlike the interpolation approach described in [1], we obtain the exact signal if it is already contained within V_0, as is assumed in [1]), and if the signal is not contained in V_0, we will obtain the best approximation of $f(x)$ in the least squares sense.

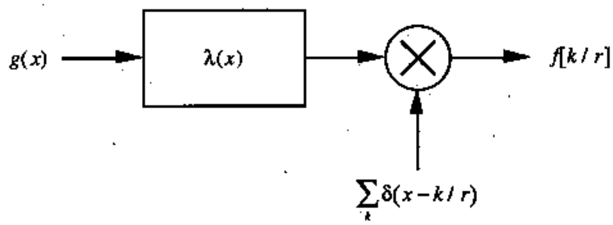

Fig. 1. Typical sample acquisition model.

II. Orthogonal Projection Method

Again, we assume that we have available samples $f[k/r]$ of the signal $f(x)$. Unlike the interpolation approach, however, we have the flexibility to assume that $f(x)$ does not belong to V_0 but is contained in some other subspace $S(\lambda_{1/r})$. This subspace should in principle depend on the impulse response of the device used to

Manuscript received April 11, 1997; revised March 20, 1998. The associate editor coordinating the review of this paper and approving it for publication was Dr. Truong Q. Nguyen.

M. J. Vrhel is with Color Savvy Systems Limited, Springboro, OH 45036 USA (e-mail: mvrhel@colorsavvy.com).

A. Aldroubi is with the Department of Mathematics, Vanderbilt University, Nashville, TN 37240 USA.

Publisher Item Identifier S 1053-587X(98)07816-7.

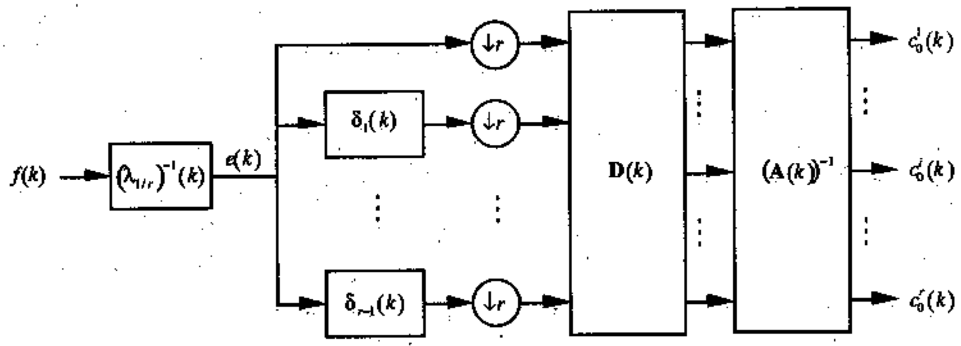

Fig. 2. System diagram of orthogonal projection initialization.

sample $f(x)$ (cf., Fig. 1). For example, $f(x)$ may be bandlimited and then sampled, in which case, $\lambda = \text{sinc}$ would be an appropriate model [6]. If the impulse response of the measuring device is not available, a model for $f(x)$ can be prescribed such as a spline model, in which case, $\lambda = \beta^n$. The subspace $S(\lambda_{1/r})$ must have the property that an interpolation basis exists. If the subspace does not have this property, then without additional information about the signal, it is not possible to obtain the necessary coefficients by linear filtering. The interpolation property of the space $S(\lambda_{1/r})$ will allow us to obtain, from samples of $f(x)$, the coefficients $e(k)$ associated with the decomposition [7]

$$f(x) = \sum_{k\in Z} e(k)\lambda(xr - k) \tag{5}$$

where here, we assume that $S(\lambda_{1/r})$ is defined by a single generating function $\lambda(x)$. Since we want to decompose the signal in terms of Φ, we would like to find the signal in V_0 that best approximates $f(x)$. If our criterion is least squares, then the solution is the orthogonal projection of $f(x)$ into V_0.

The approximation that we wish to find is given by

$$f_a(x) = \sum_{i=1}^{r}\sum_{k\in Z} c_0^i(k)\varphi^i(x-k). \tag{6}$$

The actual signal can be expressed as

$$\begin{aligned} f(x) &= \sum_{i=1}^{r}\sum_{k\in Z}[(e*\delta_{i-1})]_{\downarrow r}(k)\lambda(rx - rk - (i-1)) \\ &= \sum_{i=1}^{r}\sum_{k\in Z}[(e*\delta_{i-1})]_{\downarrow r}(k)\lambda_{1/r}(x - k - (i-1)/r) \end{aligned}$$

where $[b]_{\downarrow r}(k) = b(kr)$, $\lambda_{1/r}(x) = \lambda(rx)$, and $\delta_i(k)$ denotes the unit impulse sequence located at $k = i$.

The approximation $f_a(x)$ (or equivalently $c_0^i(k)$ $i = 1,\ldots,r$) is found by solving the orthogonality condition

$$\langle f(x) - f_a(x), \varphi^i(x-k)\rangle = 0 \quad i = 1,\ldots,r, \quad k \in Z.$$

These equations are expressed in the matrix-vector convolution format

$$(\mathbf{A} * \mathbf{c})(k) = (\mathbf{D} * \mathbf{e})(k)$$

where

$$\begin{aligned} &[\mathbf{A}]_{ij}(k) = (\varphi^j * \varphi^{i\vee})(k); \quad [\mathbf{D}]_{ij}(k) = (\lambda_{1/r}^j * \varphi^{i\vee})(k) \\ &[\mathbf{c}]_i(k) = c_0^i(k); \quad [\mathbf{e}]_i(k) = [(e*\delta_i)]_{\downarrow r}(k) \\ &\varphi^{i\vee}(x) = \varphi^i(-x); \quad \lambda_{1/r}^j(x) = \lambda_{1/r}(x - (j-1)/r). \end{aligned} \tag{7}$$

The coefficients $c_0^i(k)$ are obtained by inverting the matrix sequence $\mathbf{A}(k)$. Unlike the interpolation approach, the convolution-inverse of $\mathbf{A}(k)$ is guaranteed to exist since Φ is a Riesz basis of V_0 [8]. Thus, the orthogonal projection is

$$\mathbf{c}(k) = ((\mathbf{A})^{-1} * \mathbf{D} * \mathbf{e})(k). \tag{8}$$

A system diagram is shown in Fig. 2. The filter $(\lambda_{1/r})^{-1}(k)$ is the convolution inverse of the discrete filter $\lambda_{1/r}(k)$ [7].

It is often desirable to return to the signal coefficients $e(k)$ from $c_0^i(k)$ $i = 1,\ldots,r$ (i.e., postfiltering). This operation can be achieved by performing the inverse of the above filter. If the inverse of $\mathbf{D}(k)$ does not exist, then the postfiltering operation can be implemented by projecting into $S(\lambda_{1/r})$ from V_0. As in the prefiltering case, the projection operator is guaranteed to exist.

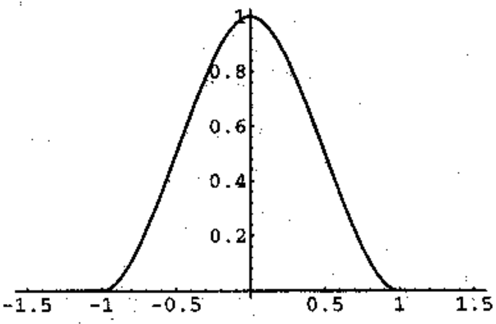

Fig. 3. $\varphi^1(x)$ of the Hermite cubic spline multiresolution.

III. Implementation and Results

The procedure to obtain the initialization by projection is as follows.

1) Choose a model for $f(x)$ as described by (5).
2) Choose the analyzing multiresolution V_0; thus, the multiscaling function $\Phi = (\varphi^1(x), \varphi^2(x), \ldots, \varphi^r(x))^T$ as desired.
3) Compute the convolution operators $\mathbf{A}$ and $\mathbf{D}$ using (7).
4) Compute the coefficients $\mathbf{c}$ using (8).

To demonstrate the usefulness of our method, we consider the initialization of a multiwavelet transform associated with the Hermite cubic spline multiresolution [9]. The two chosen ($r = 2$) scaling functions are shown in Figs. 3 and 4. These functions have the property that if

$$f(x) = \sum_{k\in Z} c_0^1(k)\varphi^1(x-k) + c_0^2(k)\varphi^2(x-k)$$

then the coefficients are given by $c_0^1(k) = f(x)|_{x=k}$ and $c_0^2(k) = f'(x)|_{x=k}$. In other words, the samples $c_0^1(k)$ are the signal samples, and the samples $c_0^2(k)$ are the samples of the signal derivative.

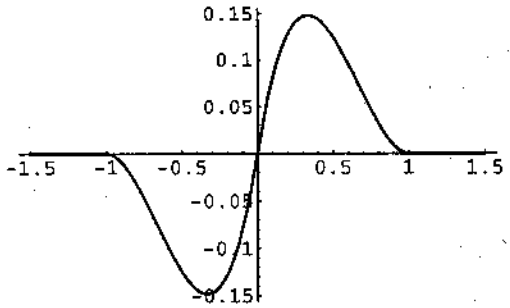

Fig. 4. $\varphi^2(x)$ of the Hermite cubic spline multiresolution.

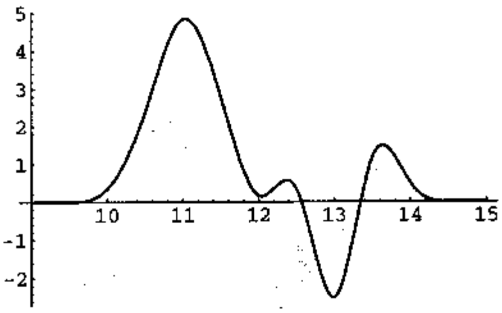

Fig. 5. Signal contained in $S(\beta_{1/2})$ but not V_0.

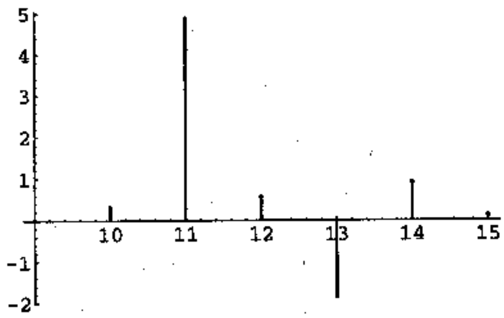

Fig. 6. Computed $c_0^1(k)$ coefficients for the signal in Fig. 5.

A. Matrix-Sequence Inverses

Since the convolution inverse of a matrix sequence is not a traditional signal processing operation, a brief discussion of a method for implementation is appropriate. Let us express the sequence $\mathbf{A}(k)$ in terms of its coefficients $[\mathbf{A}]_{ij}(k) = a_{ij}(k)$, which for the case $r = 2$ gives us

$$\mathbf{A}(k) = \begin{bmatrix} a_{11}(k) & a_{12}(k) \\ a_{21}(k) & a_{22}(k) \end{bmatrix}.$$

The convolution inverse of the matrix sequence $\mathbf{A}(k)$ is given by a matrix sequence

$$\mathbf{A}^{-1}(k) = \begin{bmatrix} \alpha_{11}(k) & \alpha_{12}(k) \\ \alpha_{21}(k) & \alpha_{22}(k) \end{bmatrix}$$

which satisfies the condition

$$(\mathbf{A} * \mathbf{A}^{-1})(k) = \begin{bmatrix} \delta(k) & 0 \\ 0 & \delta(k) \end{bmatrix} \tag{9}$$

where the matrix-matrix convolution operation $\mathbf{D}(k) = (\mathbf{B} * \mathbf{C})(k)$ is defined by

$$[\mathbf{D}]_{ij}(k) = \sum_{l=1}^{r} \sum_{h \in Z} [\mathbf{B}]_{il}(h)[\mathbf{C}]_{lj}(k-h).$$

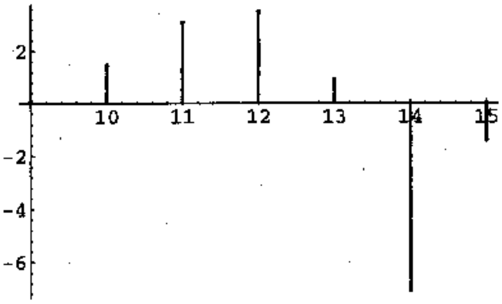

Fig. 7. Computed $c_0^2(k)$ coefficients for the signal in Fig. 5.

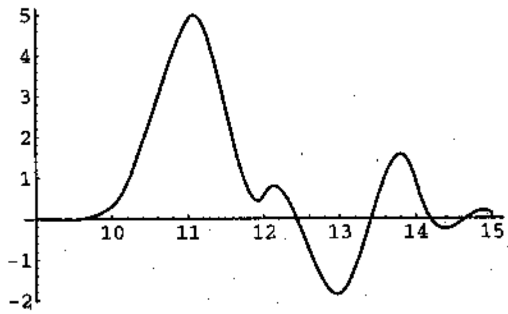

Fig. 8. Orthogonal projection of the signal in Fig. 5 into V_0.

Now, let us consider the Fourier transform of $\mathbf{A}(k)$, which we define as the Fourier transform of the scalar sequences of each element of the matrix, that is

$$\hat{\mathbf{A}}(f) = \begin{bmatrix} \hat{a}_{11}(f) & \hat{a}_{12}(f) \\ \hat{a}_{21}(f) & \hat{a}_{22}(f) \end{bmatrix}.$$

From the above two definitions, it is simple to show that the matrix convolution operation in the time domain is equivalent to computing matrix products in the frequency domain. In other words

$$\mathbf{D}(k) = (\mathbf{B} * \mathbf{C})(k) \Leftrightarrow \hat{\mathbf{D}}(f) = \hat{\mathbf{B}}(f)\hat{\mathbf{C}}(f). \tag{10}$$

Therefore, in the Fourier domain, (9) becomes

$$\hat{\mathbf{A}}(f)\hat{\mathbf{A}}^{-1}(f) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad f \in [0, 1]. \tag{11}$$

Equations (10) and (11) have the implication that the convolution inverse of the matrix $\mathbf{A}(k)$ exists if and only if the matrix $\hat{\mathbf{A}}(f)$ is nonsingular for $f \in [0, 1]$. The above equation leads to a method for approximating the sequence $\hat{\mathbf{A}}^{-1}(k)$ similar to that used for computing the convolution inverse for scalar sequences. We first sample the Fourier transform $\hat{\mathbf{A}}(f)$, producing a matrix sequence $\hat{\mathbf{A}}(l/M)$ $l = 0, \ldots, M$. Each of these matrices are inverted (in the conventional sense), resulting in a sequence $\hat{\mathbf{A}}^{-1}(l/M)$. An approximation of the convolution inverse of $\mathbf{A}(k)$ is computed from the inverse discrete Fourier transform of the matrix sequence $\hat{\mathbf{A}}^{-1}(l/M)$ $l = 0, \ldots, M$. The accuracy of the inverse can be assessed by computing $(\mathbf{A} * \mathbf{A}^{-1})(k)$ and observing how close the resulting matrix comes to the identity matrix. For example, one measure that could be used is

$$\sum_k \left\| (\mathbf{A} * \mathbf{A}^{-1})(k) - \begin{bmatrix} \delta(k) & 0 \\ 0 & \delta(k) \end{bmatrix} \right\|_F < \varepsilon$$

where $\| \|_F$ is the Frobenius norm, and ε is related to the computer precision.

Note that if $\mathbf{A}(k)$ is FIR, then $\mathbf{A}^{-1}(k)$ will decay exponentially fast. In addition, $\mathbf{A}^{-1}(k)$ can be efficiently implemented by a recursive algorithm.

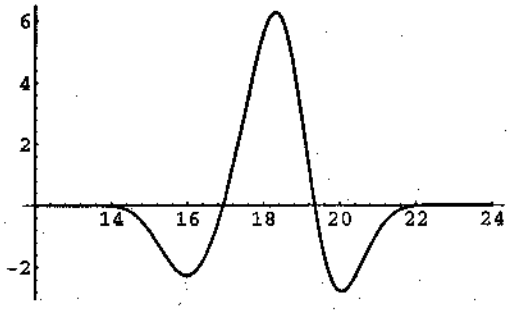

Fig. 9. Signal contained in $S(\beta)$ and V_0.

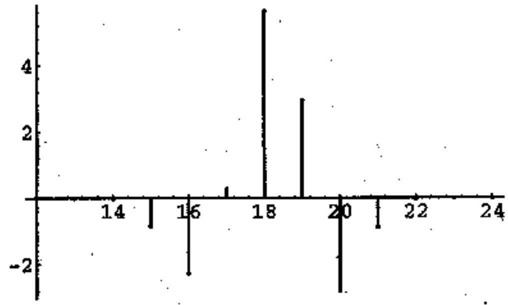

Fig. 10. Computed $c_0^1(k)$ coefficients for the signal in Fig. 9.

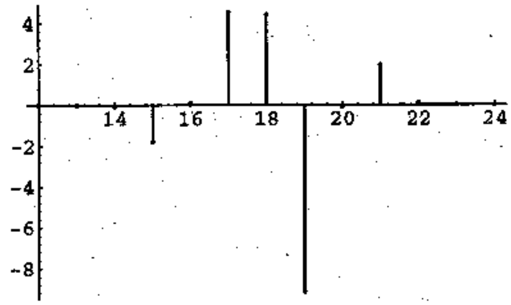

Fig. 11. Computed $c_0^2(k)$ coefficients for the signal in Fig. 9.

B. Interpolation Approach

For the Hermite cubic spline multiscaling function, an interpolating set of functions does not exist. In other words, the convolutional inverse of matrix $\mathbf{B}(k)$ [cf., (2)] does not exist. The singularity can be easily shown by computing the Fourier transform of the filter

$$\mathbf{B}(k) = \begin{bmatrix} (1,0) & (0,0) \\ (1/2,1/2) & (-1/8,1/8) \end{bmatrix}$$

which is given by the matrix

$$\hat{\mathbf{B}}(f) = \begin{bmatrix} 1 & 0 \\ (1/2)(1+e^{-j2\pi f}) & (1/8)(e^{-j2\pi f}-1) \end{bmatrix}.$$

As mentioned, the matrix $\hat{\mathbf{B}}(f)$ must be nonsingular for $f \in [0,1]$, which is not true since the matrix $\hat{\mathbf{B}}(0)$ is singular.

C. Orthogonal Projection Approach

Unlike the interpolation approach, a solution always exists for our projection-based method. In addition, the signal need not be contained within V_0. However, if the signal is within V_0, then the approximation will be exact since we are performing a projection operation into V_0.

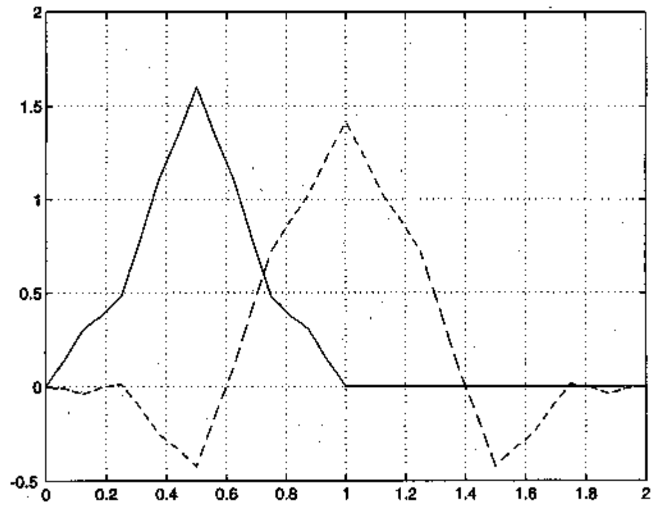

Fig. 12. Geronimo, Hardin, and Massopust multiscaling function: $\varphi^1(x)$ solid and $\varphi^2(x)$ dashed.

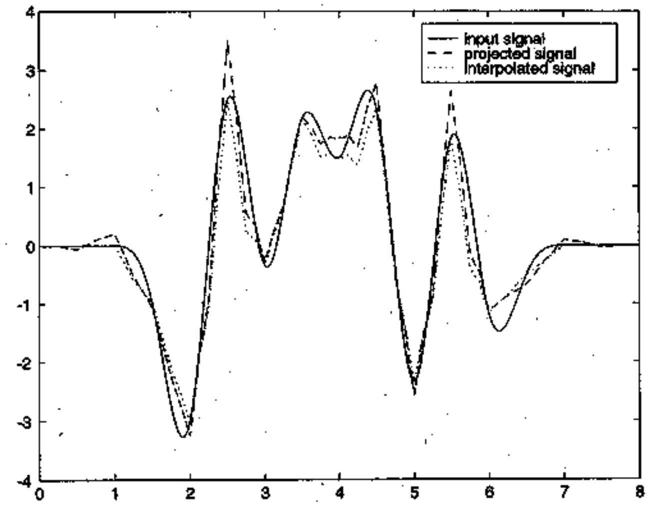

Fig. 13. Comparison of prefiltering with projection-based method and intepolation-based method.

Example 1—Input Signal Not in V_0: As an example, let us assume that our signal is contained within the space $S(\beta_{1/2})$ defined by a 1/2 scaled cubic B-spline function. That is

$$f(x) = \sum_{k \in Z} e(k)\beta(2x-k). \tag{6}$$

The coefficients $e(k)$ are easily computed from the samples $f(k/2)$ by use of the inverse filter $(\beta_{1/2})^{-1}(k)$ [7]. Note that since $f(x)$ is not guaranteed to have C^∞ smoothness at the points $x = (2k+1)/2$, the space $S(\beta_{1/2})$ is not contained in the Hermite cubic spline space V_0. The signal shown in Fig. 5 is contained in $S(\beta_{1/2})$ but not V_0. The exact coefficients starting at $k=0$ are given by $e(k) = [0,\ldots,0,2,6,3,-1,2,-5,3,0,0,\ldots]$ where the first 20 coefficients are zero. Fig. 8 displays the orthogonal projection of this signal into the space V_0. The coefficients $c_0^1(k) = [0,\ldots,0,0.309,4.88,0.572,-1.85,0.915,0.118,0.032,0.009,0,\ldots,0]$ and $c_0^2(k) = [0,\ldots,0,1.445,3.088,3.507,0.905,-7.086,-1.413,-0.394,-0.111,0,\ldots,0]$, which are shown in Figs. 6 and 7, are the sample values of the projection and the sample values of the projection's derivative, respectively.

Example 2—Input Signal in V_0: For this example, we use a function in the cubic B-spline space $S(\beta)$. One such signal can be expressed in terms of (6) with coefficients $e(k) = 1/8 \quad [0, \ldots, 0, -1, -6, -15, -20, -14, 1, 23, 48, 54, 25, -13, -26, -17, -6, -1, 0, 0, \ldots]$, where the first 30 coefficients are zero. It is not difficult to show that $S(\beta) \subset V_0$. Filtering this signal, shown in Fig. 9 through the system in Fig. 2 provides the coefficients $c_0^1(k) = [0, \ldots, 0, -0.0192, -0.832, -2.27, 0.261, 5.62, 2.941, -2.807, -0.868, -0.0198, 0, \ldots, 0]$, and $c_0^2(k) = [0, \ldots, 0, -0.087, -1.79, 0.103, 4.52, 4.42, -9.131, -0.079, 1.95, 0.091, 0, \ldots, 0]$ displayed in Figs. 10 and 11. Since the signal is already in the space V_0, our projection approach provides coefficients that are exactly the sample values of the signal and its derivative. For this example, the direct interpolation approach [1] cannot obtain $c_0^1(k)$ and $c_0^2(k)$, even though the signal is in V_0. The reason is that the matrix sequence $(\mathbf{B})^{-1}(k)$ does not exist as mentioned at the beginning of this section.

D. Interpolation versus Least Squares

To compare the interpolation-based method with our method, it is necessary to select a multiscaling function for which the interpolation based approach will work. One such multiscaling function was introduced by Geronimo *et al.* [2]. The multiscaling function is shown in Fig. 12. A signal that is contained in the space $S(\beta_{1/2})$ is shown in Fig. 13 with the least squares and interpolation approximations. The exact coefficients for the input signal are given by $e(k) = [0, \ldots, 0, -6, 6, -3, 4, 0, \ 5, -6, 5, -3, 0, 0, \ldots]$ where the first nine coefficients are zero. Prefiltering the signal with the interpolation method gives $c_0^1(k) = [0, 0, 0, 0, -1.187, 0.938, 1.573, 1.302, 0.490, -0.531, \ 0, 0, 0]$, and $c_0^2(k) = [0, 0, 0, 0, -2.122, -0.236, 1.061, -1.65, -0.825, 0, 0, 0, 0]$, whereas the projection method gives $c_0^1(k) = [0, 0, 0, 0, \ -1.239, 1.52, 1.67, 1.60, 0.958, -0.612, 0, 0, 0]$, and $c_0^2(k) = [0, 0, 0, 0.145, -2.29, -0.195, 1.30, -1.816, -0.798, 0.071, 0, 0, 0]$. The interpolating method and the projection method produce square errors of 1.13 and 0.84, respectively. Both methods were implemented via matrix sequence convolutions.

IV. Conclusion

We have introduced a new method for initializing (by prefiltering) the multiwavelet decomposition algorithm. The approach has the following attractive features:

- The projection filter will always exist, unlike the interpolation filter.
- If the signal is contained within the space defined by the multiscaling function, then the projection solution will be an exact approximation of the signal.
- The method is more flexible and general than previous methods since it provides a least squares solution when the original signal is not contained in the space defined by the multiscaling function.
- If the matrix sequence $\mathbf{A}(k)$ is FIR, then the initialization algorithm is a fast filtering algorithm.

References

[1] X. Xia, J. S. Geronimo, D. P. Hardin, and B. W. Suter, "Design of prefilters for discrete multiwavelet transforms," *IEEE Trans. Signal Processing*, vol. 44, pp. 25–35, Jan. 1996.

[2] J. S. Geronimo, D. P. Hardin, and P. R. Massopust, "Fractal functions and wavelet expansions based on several scaling functions," *J. Approx. Theory*, vol. 78, pp. 373–401, 1994.

[3] G. Strang and T. Nguyen, *Wavelets and Filter Banks*. Wellesley, MA: Wellesley-Cambridge, 1996.

[4] A. Aldroubi, "Oblique and hierachical multiwavelet bases," *Appl. Comput. Harmon. Anal.*, vol. 4, pp. 231–263, 1997.

[5] A. Aldroubi and J. McGowan, "Oblique and biorthogonal multiwavelet spaces with optimal properties," *Mathematical Imaging: Wavelet Applications in Signal and Image Processing*. Bellingham, WA: SPIE, 1995, vol. 2569, pp. 15–26.

[6] M. Unser and A. Aldroubi, "A general sampling theory for nonideal acquisition devices," *IEEE Trans. Signal Processing*, vol. 42, pp. 2915–2925, Nov. 1994.

[7] A. Aldroubi and M. Unser, "Sampling procedures in function spaces and asymptotic equivalence with Shannon's sampling theory," *Numer. Funct. Anal. Optimiz.*, vol. 15, no. 1, pp. 1–21, 1994.

[8] A. Aldroubi, "Oblique projections in atomic spaces," in *Proc. Amer. Math. Soc.*, 1996, vol. 124, no. 7, pp. 2051–2060.

[9] C. Heil, G. Strang, and V. Strela, "Approximation order by translates of refinable functions," preprint, 1995.

[10] M. J. Vrhel and A. Aldroubi, "Pre-filtering for the initialization of multi-wavelet transforms," in *Proc. IEEE ICASSP*, to be published.

1053–587X/98$10.00 © 1998 IEEE