BIORTHOGONAL BUTTERWORTH WAVELETS

Valery A. Zheludev, Amir Z. Averbuch.

School of Computer Science
Tel Aviv University
Ramat Aviv, Tel Aviv, 69978, ISRAEL

Alexander B. Pevnyi

Department of Mathematics,
Syktyvkar University,
Syktyvkar, RUSSIA

ABSTRACT

In the paper we present a new family of biorthogonal wavelet transforms and a related library of biorthogonal symmetric wavelets. For the construction we use the interpolatory discrete splines which enable us to design perfect reconstruction filter banks related to the Butterworth filters. The construction is performed in a "lifting" manner. The proposed scheme is based on interpolation and, as such, it involves only samples of signals and it does not require any use of quadrature formulas. These filters have linear phase property. The filters yield perfect frequency resolution.

1. INTRODUCTION

The discrete splines yield a natural tool for processing of discrete-time signals. In this work we employed the interpolatory discrete splines ([**?**]) as a tool for devising a discrete biorthogonal wavelet scheme. The construction is based on the "lifting scheme" presented by Sweldens in [**?**]. The lifting scheme allows custom design and fast implementation of the transforms. Briefly, the idea of the computation is that values of the signal located at odd positions are predicted by values in the midpoints of the spline that interpolates even values of the signal. Then, the odd subarray is replaced by the difference between the current and the predicted subarrays.Then we employ the new odd subarray for updating the existing even subarray in order to smooth the even subarray and thus reduce the aliasing which is a consequence of the decimation. Based upon the above strategy, we constructed a new family of biorthogonal wavelet transforms and a related library of biorthogonal symmetric wavelets. The transforms are implemented using perfect reconstruction filter banks that have linear phase property. Our investigation revealed an interesting relation between the decomposition splines and the popular Butterworth filters. The filter banks constructed in the paper comprise filters which act as a bi-directional (forward and backward) half-band Butterworth filters. The frequency response of Butterworth filters are maximally flat and we succeeded in construction of the dual filters with similar property. The corresponding wavelets we name the Butterworth wavelets. We present explicit formulas for the construction of wavelets with arbitrary number of vanishing moments. The computations in our scheme are conducted in the time domain using recursive filtering. The filters which we use are symmetric and allow fast cascade or parallel implementation.

2. PRELIMINARIES

Discrete splines. We outline briefly the properties of discrete splines ([**?**]) and recall the notion of the Butterworth filter. Sequences $\{a(k)\}_{k=-\infty}^{\infty}$, which belong to the space l_1, we will call the discrete-time signals. We will use the $z-$transform of signals assuming that $|z|=1 \Longleftrightarrow z=e^{i\omega}$.

We will use the discrete splines of order $p=2r,\ r\in\mathbb{N}$ builded on the grid with the step equal to 2 samples. These splines are defined as linear combinations, with real-valued coefficients, of shifts of the central B-spline of order $2r$: $S_r(k) = \sum_{l=-\infty}^{\infty} c(l)Q_r(k-2l)$, where the central B-spline $Q_r(j)$ of order $2r$ is the signal whose $z-$transform is $Q_r(z)=z^r(1+z^{-1})^{2r}$.

Let $\{e(k)\},\ k\in\mathbb{Z}$ be a given sequence.The discrete spline S_r is called the interpolatory spline if the following relations hold: $S_r(2k)=e(k),\ k\in\mathbb{Z}$. The points $\{2k\}$ are called the nodes of the spline. The interpolatory splines of any order can be explicitly constructed but for further development we need to know only the values of the splines in the midpoints between the nodes, which we denote as $\sigma(k) = S_r(2k+1),\ k\in\mathbb{Z}$. The $z-$transform of the sequence σ is $\sigma(z)=zU(z)e(z^2)$, where

$$U(z) \triangleq \frac{\left(1+z^{-1}\right)^{2r} - (-1)^r\left(1-z^{-1}\right)^{2r}}{\left(1+z^{-1}\right)^{2r} + (-1)^r\left(1-z^{-1}\right)^{2r}}. \tag{1}$$

Discrete-time Butterworth filters. We recall briefly the notion of Butterworth filter. For details we refer to [**?**]. The digital Butterworth filter is the filter with the maximally flat frequency response. The magnitude squared frequency response $F_l(\omega)$ of the digital low-pass Butterworth filter of order r is $|F_l(\omega)|^2 = \left[1+(\tan\frac{\omega}{2}/\tan\frac{\omega_c}{2})^{2r}\right]^{-1}$. where ω_c is the so-called cutoff frequency.

We are interested in the half-band Butterworth filters that is $\omega_c = \pi/2$. In this case $|F_l(\omega)|^2 = \left[1 + (\tan\frac{\omega}{2})^{2r}\right]^{-1}$. If we put $z = e^{i\omega}$ then we obtain the magnitude squared transfer function of the low-pass filter:

$$|f_l(z)|^2 = \frac{\left(1+z^{-1}\right)^{2r}}{\left(1+z^{-1}\right)^{2r} + (-1)^r\left(1-z^{-1}\right)^{2r}}$$

Similarly we have the magnitude squared transfer function of the high-pass filter $f_h(z)$.The following important relations are true:

$$1 + U(z) = 2|f_l(z)|^2, \quad 1 - U(z) = 2|f_h(z)|^2. \qquad (2)$$

3. BIORTHOGONAL TRANSFORMS

We introduce a family of biorthogonal wavelet-type transforms that operate on the signal $\mathbf{x} = \{x(k)\},\ k \in \mathbb{Z}$, which we construct through lifting steps. We carry out the construction in the $z-$ domain and discuss the time-domain implementation in the subsequent sections.

The lifting scheme can be implemented in either a primal or dual modes.For brevity, we consider only the primal one.

Decomposition The lifting scheme for decomposition of signals consists of three steps: 1. Split. 2. Predict. 3. Update or lifting.**1.Split -** We simply split the array $\mathbf{x}$ into an even and odd sub-arrays: $\mathbf{e_1} = \{e_1(k) = x(2k)\}$, $\mathbf{d_1} = \{d_1(k) = x(2k+1)\}$. **2.Predict -** We use the spline S_r which interpolates the sequence $\mathbf{e_1}$ to predict the odd array $\mathbf{d_1}$ and redefine the array $\mathbf{d_1}$ as the difference between the existing array and the predicted one. To be specific, we predict the function $d_1(z^2)$ which is the z^2-transform of $\mathbf{d_1}$ by the function $\sigma(z) = zU(z)e(z^2)$. We have for the new $d-$array:

$$d_1^u(z^2) = d_1(z^2) - zU(z)e_1(z^2). \qquad (3)$$

¿From now on the superscript u means an *update* operation of the array. **3.Lifting -** We update the even array using the new odd array:

$$e_1^u(z^2) = e_1(z^2) + \beta(z)z^{-1}\, d_1^u(z^2). \qquad (4)$$

The goal of this step is to eliminate aliasing which appears while downsampling the original signal $\mathbf{x}$ into $\mathbf{e}_1$. By doing so we have that $\mathbf{e_1}$ is transformed into a low-pass filtered and downsampled replica of $\mathbf{x}$. We can achieve this effect by the choice $\beta(z) = U(z)/2z$, but many more options to choose β exist which allow the custom design of the transform.

Reconstruction The reconstruction of the signal $\mathbf{x}$ from the arrays $\mathbf{e_1^u}$ and $\mathbf{d_1^u}$ is implemented in reverse order:**1.Undo Lifting -** We restore immediately the even array: $e_1(z^2) = e_1^u(z^2) - \beta(z)z^{-1}\, d_1^u(z^2)$..**2.Undo Predict -** We restore the odd array: $d_1(z^2) = d_1^u(z^2) + zU(z)e_1(z^2)$. **3.Unsplit -** The last step represents the standard restoration of the signal from its even and odd components. In the $z-$ domain it looks as $x(z) = e_1(z^2) + z^{-1}d_1(z^2)$.

4. FILTER BANKS

Relation to the Butterworth filters Lifting schemes, that were presented above, yield efficient algorithms for the implementation of the forward and backward transform of $\mathbf{x} \longleftrightarrow \mathbf{e_1^u} \cup \mathbf{d_1^u}$. But these operations can be interpreted as transformations of the signals by a filter bank that possesses the perfect reconstruction properties.

Denote $B_{l,r}(z) \stackrel{\Delta}{=} (1+U(z))/2, B_{h,r}(z) \stackrel{\Delta}{=} (1-U(z))/2.$. ¿From (**??**) it is clear that the linear phase filters $B_{l,r}(z)$, $B_{h,r}(z)$ are equal to the magnitude squared frequency-response functions of the discrete-time low- and high-pass half-band Butterworth filters of order r respectively. It means that application of these filters on a signal is equivalent to application of two passes (forward and backward) of the corresponding Butterworth filters. We call these filters the bidirectional Butterworth filters. Define the filter functions

$$\tilde{g}(z) \stackrel{\Delta}{=} \frac{2}{z}B_{h,r}(z),\ \tilde{h}_\beta(z) \stackrel{\Delta}{=} 1 + 2\beta(z)B_{h,r}(z), \qquad (5)$$

$$h(j) = 2B_{l,r}(z), \quad g_\beta(z) = \frac{2}{z}B_{l,r}(z). \qquad (6)$$

Theorem 1 *The decomposition and reconstruction formulas can be represented as follows:*

$$e_1^u(z^2) = \frac{1}{2}\left(\overline{\tilde{h}_\beta(z)}\, x(z) + \overline{\tilde{h}_\beta(-z)}\, x(-z)\right) \qquad (7)$$

$$d_1^u(z^2) = \frac{1}{2}\left(\overline{\tilde{g}(z)}\, x(z) + \overline{\tilde{g}(-z)}\, x(-z)\right) \qquad (8)$$

$$\hat{x}(z) = h(z)e_1^u(z^2) + g_\beta(z)\, d_1^u(z^2). \qquad (9)$$

We call the functions $\{\tilde{h}_\beta(z)\}$ and $\{\tilde{g}(z)\}$ the low-pass and high-pass decomposition filters, respectively. We call the functions $\{h(z)\}$ and $\{g_\beta(j)\}$ the low-pass and high-pass reconstruction filters, respectively. These four filters form a perfect reconstruction filter bank. ([**?**]).

The filter functions are linked in the following way: $\tilde{g}(z) = z^{-1}h(-z)$, $g_\beta(z) = z^{-1}\,\overline{\tilde{h}_\beta(-z)}$

Recursive implementation of the transforms First we note that (**??**) implies that the function $zU(z)$ depends actually on z^2 and we denote it as

$$F_r(z^2) \stackrel{\Delta}{=} zU(z) = z\frac{(1+z)^{2r} - (-1)^r\,(z-1)^{2r}}{(1+z)^{2r} + (-1)^r\,(z-1)^{2r}}.$$

Let us consider the primal decomposition procedure. Since $zU(z) = F_r(z^2)$, the predicting formula (**??**) is equivalent to the following $d_1^u(z) = d_1(z) - F_r(z)e_1(z)$. The latter relation means that, to obtain the detail array $\mathbf{d}_1^u$, we must process the even array $\mathbf{e}_1$ by the filter F_r with the transfer function $F_r(z)$ and extract the filtered array from the odd array $\mathbf{d}_1$.It can be done in the recursive mode. Moreover, the filter can be decomposed into the product or the sum of elementary recursive filters and, by this means allow the cascade or parallel implementation.

If the order $r = 2p + 1$ then we denote

$$\alpha_k^r \triangleq \cot^2 \frac{(p+k)\pi}{2r} < 1, \gamma_k^r \triangleq \cot^2 \frac{(2p+2k+1)\pi}{4r} < 1,$$

$k = 1, \ldots, p$. If $r = 2p$ then
$\alpha_k^r \triangleq \cot^2 \frac{(2p+2k-1)\pi}{4r} < 1,\ k = 1, \ldots, p$. and
$\gamma_k^r \triangleq \cot^2 \frac{(p+k)\pi}{2r} < 1,\ k = 1, \ldots, p-1$.

Theorem 2 *If $r = 2p + 1$ then the function $F_r(z)$ is represented as follows:*

$$F_r(z) = A_r(1+z) \prod_{k=1}^{p} R_r(z,k) \prod_{k=1}^{p} R_r(z^{-1},k),$$

where $$A_r \triangleq \frac{\alpha_1^r \alpha_2^r \ldots \alpha_p^r}{2r\gamma_1^r\gamma_2^r \ldots \gamma_p^r},\quad R_r(z,k) \triangleq \frac{1+\gamma_k^r z}{1+\alpha_k^r z}.$$

If $r = 2p$ then

$$F_r(z) = A_r(1+z) \prod_{k=1}^{p} R_r(z,k) \prod_{k=1}^{p} R_r(z^{-1},k),$$

where $$A_r \triangleq \frac{2r\alpha_1^r \alpha_2^r \ldots \alpha_p^r}{\gamma_1^r\gamma_2^r \ldots \gamma_{p-1}^r},\quad R_r(z,p) \triangleq \frac{1}{1+\alpha_p^r z},$$

$$R_r(z,k) \triangleq \frac{1+\gamma_k^r z}{1+\alpha_k^r z},\ k = 1, \ldots, p-1.$$

The theorem implies that both the filters F_{2p} and F_{2p+1} can be split into a cascade of p elementary forward recursive filters $\overrightarrow{R_r(k)}$ with the transfer function $R_r(z^{-1},k)$, p elementary backward recursive filters $\overleftarrow{R_r(k)}$ with the transfer function $R_r(z,k)$ and the the FIR filter Q with the transfer function $1+z$. On the other hand, the filters can be decomposed into sums of elementary recursive filters. Such a decomposition allows parallel implementation of the transform. The filters are acting as follows:

$$\begin{aligned}
\mathbf{y} &= \overrightarrow{R_r(k)}\mathbf{x} \Longleftrightarrow y(l) = x(l) + \gamma_k^r x(l-1) - \alpha_k^r y(l-1),\\
\mathbf{y} &= \overleftarrow{R_r(k)}\mathbf{x} \Longleftrightarrow y(l) = x(l) + \gamma_k^r x(l+1) - \alpha_k^r y(l+1),\\
\mathbf{y} &= Q\mathbf{x} \Longleftrightarrow y(l) = x(l) + x(l+1).
\end{aligned}$$

If the control filter is chosen as $\beta_1(z) = U(z)/2$ then the implementation of the update step of decomposition (see (**??**)) is just similar to the previous step. To be specific, to obtain the smoothed array $\mathbf{e}_1^u$, we must process the detail array $\mathbf{d}_1^u$ by the filter Φ_r with the transfer function $\Phi_r(z) = z^{-1}F_r(z)/2$ and add the filtered array to the even array $\mathbf{e}_1$. But the filter Φ_r differs from $F_r/2$ only by one-sample delay and is acting just similarly.

Since the reconstruction in the lifting scheme differs from the decomposition only by the order of operations, its implementation is completely explained above.

Example. Let $r = 2$.In this case $\alpha_1^2 = 3-2\sqrt{2} \approx 0.172$ and

$$F_2(z) = 4\alpha_1^2 \frac{1+z}{(1+\alpha_1^2 z)(1+\alpha_1^2 z^{-1})}.$$

The filter can be implemented as the following cascade:

$$\begin{aligned}
x_1(k) &= 4\alpha_1^2 x(k) - \alpha_1^2 x_1(k-1),\\
y(k) &= x_1(k) + x_1(k+1) - \alpha_1^2 y(k+1).
\end{aligned}$$

Another option stems from the following decomposition of the function $F_2(z)$:

$$F_2(z) = \frac{4}{1+\alpha_1^2}\left(\frac{1}{1+\alpha_1^2 z^{-1}} + \frac{z}{1+\alpha_1^2 z}\right).$$

Then the filter is implemented in parallel mode:

$$\begin{aligned}
y_1(k) &= x(k) - \alpha_1^2 y_1(k-1)\\
y_2(k) &= x(k+1) - \alpha_1^2 y_2(k+1)
\end{aligned} \quad y = \frac{4\,(y_1+y_2)}{1+\alpha_1^2}.$$

We note that elementary filters which produce y_1 and y_2 are operating in the opposite directions.

5. BASES FOR THE SIGNAL SPACE

The perfect reconstruction filter banks, that were constructed above, are associated with the biorthogonal pairs of bases in the space $\mathcal{S}$ of discrete signals.

In the previous section we introduced a family of filters by their transfer functions $h(z)$, $g_\beta(z)$, $\tilde{h}_\beta(z)$, $\tilde{g}(z)$. We denote by $\varphi^1(k)$, $\psi_\beta^1(k)$, $\tilde{\varphi}_\beta^1(k)$, $\tilde{\psi}^1(k)$ the impulse response functions of the corresponding filters. It means that, for example

$$h(z) = \sum_{k\in\mathbb{Z}} z^{-k}\varphi^1(k)$$

and similarly for the other functions.

Theorem 3 *The shifts of functions $\varphi^1(k)$, $\psi_\beta^1(k)$, $\tilde{\varphi}_\beta^1(k)$, $\tilde{\psi}^1(k)$, form a biorthogonal pairs of bases in the space $\mathcal{S}$. This means that any signal $\mathbf{x} \in \mathcal{S}$ can be represented as:*

$$x(l) = \sum_{k\in\mathbb{Z}} e_1^u(k)\,\varphi^1(l-2k) + \sum_{k\ in\mathbb{Z}} d_1^u\,\psi_\beta^1(l-2k).$$

The coordinates $e_1^u(k)$ and $d_1^u(k)$ can be represented as the inner products:

$$\begin{aligned}
e_1^u(k) &= \langle \mathbf{x}, \tilde{\varphi}_{\beta,k}^1\rangle, \quad \textit{where}\ \tilde{\varphi}_{\beta,k}^1(l) = \tilde{\varphi}_\beta^1(l-2k)\\
d_1^u(k) &= \langle \mathbf{x}, \tilde{\psi}_k^1\rangle, \quad \textit{where}\ \tilde{\psi}_k^1(l) = \tilde{\psi}^1(l-2k).
\end{aligned}$$

The theorem justifies the following definition: The functions φ^1 and ψ^1_β, which belong to the space $\mathcal{S}$, are called the low-frequency and high-frequency reconstruction wavelets of the first scale, respectively. The functions $\tilde{\varphi}^1_\beta$ and $\tilde{\psi}^1$, which belong to the space $\mathcal{S}$, are called the low-frequency and high-frequency decomposition wavelets of the first scale respectively. Moreover, the following biorthogonal relations hold $\forall l,\ k$:

$$\langle \tilde{\varphi}^1_{\beta,k}, \varphi^1_l \rangle = \langle \psi^1_{\beta,k}, \tilde{\psi}^1_l \rangle = \delta^l_k,\ \langle \tilde{\varphi}^1_{\beta,k}, \psi^1_{\beta,l} \rangle = \langle \tilde{\psi}^1_l, \varphi^1_k \rangle = 0.$$

We say that a wavelet ψ has m vanishing moments if the following relations hold

$$\sum_{k\in\mathbb{Z}} k^s \psi(k) = 0, \quad s = 0, 1, \ldots, m-1.$$

Proposition 1 *The high-frequency decomposition and reconstruction wavelets of order r have $2r$. vanishing moments*

The filters which are used in our wavelet transform are combination of the bi-directional Butterworth filters. Therefore, it is appropriate to name the corresponding wavelets the *Butterworth wavelets.*

6. MULTISCALE WAVELET TRANSFORMS

Repeated applications of the transform can be achieved in an iterative way. It can be implemented as either a linear invertible transform of a wavelet type or as a wavelet packet type transform which results in an overcomplete representation of the signal. We explain one multiscale advance of the wavelet transform.

In this transform we store the array $\mathbf{d}^u_1$ and decompose the array $\mathbf{e}^u_1$. The transformed arrays $\mathbf{e}^u_2$ and $\mathbf{d}^u_2$ of the second decomposition scale are derived from the even and odd sub-arrays of the array $\mathbf{e}^u_1$ by the same lifting steps as those described in Section **??**. The transform is implemented using the recursive filters presented in Section **??**. As the result we have the signal $\mathbf{x}$ transformed into three subarrays: $\mathbf{x} \leftrightarrow \mathbf{d}^u_1 \bigcup \mathbf{d}^u_2 \bigcup \mathbf{e}^u_2$. The reconstruction is implemented in the reverse order.

Again the transform leads to expansions of the signal with biorthogonal pairs of bases as follows

$$x(l) = \sum_{k\in\mathbb{Z}} e^u_2(k)\, \varphi^2(l-4k) + \sum_{k\in\mathbb{Z}} d^u_2(k)\, \psi^2_\beta(l-4k) + \sum_{k\in\mathbb{Z}} d^u_1(k)\, \psi^1_\beta(l-2k)$$

where low- and high-frequency reconstruction wavelets of the second scale are defined as
$\varphi^2(l) = \sum_{k\in\mathbb{Z}} \varphi^1(k)\varphi^1(l-2k)$,
$\psi^2_\beta(l) = \sum_{k\in\mathbb{Z}} \psi^1_\beta(k)\varphi^1(l-2k)$.

The coordinates are inner products with 4-sample shifts of the decomposition wavelets of the second scale:
$\tilde{\varphi}^2_\beta(l) = \sum_{k\in\mathbb{Z}} \tilde{\varphi}^1_\beta(k)\tilde{\varphi}^1_\beta(l-2k)$,
$\tilde{\psi}^2_\beta(l) = \sum_{k\in\mathbb{Z}} \tilde{\psi}^1(k)\tilde{\varphi}^1_\beta(l-2k)$. Namely,

$$e^u_2(k) = \langle \mathbf{x}, \tilde{\varphi}^2_{\beta,k} \rangle, \quad \text{where } \tilde{\varphi}^2_{\beta,k}(l) = \tilde{\varphi}^2_\beta(l-4k)$$
$$d^u_2(k) = \langle \mathbf{x}, \tilde{\psi}^2_{\beta,k} \rangle, \quad \text{where } \tilde{\psi}^2_{\beta,k}(l) = \tilde{\psi}^2_\beta(l-4k).$$

Unlike the mechanism in the wavelet transform, in the wavelet packet transform both sub-arrays $\mathbf{e}^u_1$ and $\mathbf{d}^u_1$ of the first scale are subject to decomposition that produces four second-scale sub-arrays. In turn, these four arrays produce eight sub-arrays for the third scale, and so on. All sub-arrays which are related to a certain scale are stored.

7. CONCLUSIONS

We presented a new family of biorthogonal wavelet transforms and the related library of biorthogonal periodic symmetric waveforms. For the construction we used the interpolatory discrete splines which enabled us to design a library of perfect reconstruction filter banks. These filter banks are intimately related to Butterworth filters. The construction is performed in a "lifting" manner that allows more efficient implementation and provides tools for custom design of the filters and wavelets. The difference with the conventional lifting scheme [**?**] is that all the transforms are implemented using recursive IIR filters related to the digital Butterworth filters. The filters are symmetric and allow fast cascade or parallel implementation. We established explicit formulas for the construction of wavelets with arbitrary number of vanishing moments. For the equal number of vanishing moments, the computational complexity of the proposed transform is remarkably lower than the complexity of the transforms with the compactly supported wavelets. The filters have linear phase property and the wavelets are symmetric. In addition, these filters yield perfect frequency resolution.

8. REFERENCES

[1] A. V. Oppenheim, R. W. Shafer, Discrete-time signal processing, Englewood Cliffs, New York, Prentice Hall, 1989.

[2] A. B. Pevnyi and V. A. Zheludev, "On the interpolation by discrete splines with equidistant nodes", J. Appr. Th., vol. 102, pp. 286-301, 2000.

[3] G. Strang, and T. Nguen, Wavelets and filter banks, Wellesley-Cambridge Press, 1996.

[4] W. Sweldens "The lifting scheme: A custom design construction of biorthogonal wavelets", Appl. Comput. Harm. Anal. vol. 3(2), pp. 186-200, 1996.