Daubechies Wavelets and Interpolating Scaling Functions and Application on PDEs

R. Schneider
F. Krüger

TUB - Technical University of Berlin

November 22, 2007
Conditions that fulfill the Daubechies wavelets:

1. They perform an orthonormal basis.
2. They have $p$ vanishing moments for a given $p \in \mathbb{N}$.
3. They have the minimal support of all functions that fulfill the first two conditions.
4. They are rather smooth.
The Fourier transform of a function $f$ is defined by

$$\hat{f}(\omega) = \int_{\mathbb{R}} f(x) e^{-i\omega x} \, dx$$

Given a scaling function $\varphi$ and its corresponding wavelet $\psi$ with its refinement equations

$$\varphi = \sum_k h_k \varphi^1_k$$
$$\psi = \sum_k g_k \varphi^1_k.$$
In Fourier space it gets to

\[
\hat{\varphi}(\omega) = \frac{\sqrt{2}}{2} h\left(\frac{\omega}{2}\right) \hat{\varphi}\left(\frac{\omega}{2}\right)
\]

\[
\hat{\psi}(\omega) = \frac{\sqrt{2}}{2} g\left(\frac{\omega}{2}\right) \hat{\varphi}\left(\frac{\omega}{2}\right)
\]

with

\[
h(\omega) = \sum_{k \in \mathbb{Z}} h_k e^{-ik\omega}
\]

\[
g(\omega) = \sum_{k \in \mathbb{Z}} g_k e^{-ik\omega}.
\]

By successive application of (1) we get

\[
\hat{\varphi}(\omega) = \hat{\varphi}(0) \prod_{k=1}^{\infty} \frac{h(2^{-k}\omega)}{\sqrt{2}}
\]

if \( \varphi \) is continuous at 0.
There is no analytical formula for the Daubechies scaling functions and wavelets. But you can compute exactly the filters $h$ and $g$. They are tabulated in the internet, for example at wikipedia.
Computation of the Daubechies Wavelets

With the filters you can use the formulae in the Fourier space.

\[
\hat{\varphi}(\omega) = \prod_{k=1}^{\infty} \frac{h(2^{-k}\omega)}{\sqrt{2}}
\]

\[
\hat{\psi}(\omega) = \frac{1}{\sqrt{2}} g\left(\frac{\omega}{2}\right) \hat{\varphi}\left(\frac{\omega}{2}\right)
\]

with

\[
h(\omega) = \sum_{k \in \mathbb{Z}} h_k e^{-ik\omega}
\]

\[
g(\omega) = \sum_{k \in \mathbb{Z}} g_k e^{-ik\omega}.
\]
Figure: Daubechies scaling functions $\varphi$ with $p = 2, 3, 4$ and 5.
Figure: Daubechies wavelets $\psi$ with $p = 2, 3, 4$ and $5$. 
Vanishing Moments

A very important property of the Daubechies wavelets are the vanishing moments

$$\int_{\mathbb{R}} x^k \psi(x) \, dx = 0, \quad k = 0, \ldots, p - 1.$$  

That also means that every polynomial of degree at most $p - 1$ is in $V_0$ in a compact subset $K$ of $\mathbb{R}$. For example is $f = 1_K \in V_0|_K$. The series $(\sum_{k=-n}^{n} \varphi_k(x))_{n \in \mathbb{N}}$ is converging pointwise to the unity.
In the picture is shown the function

\[ f = \sum_{k=0}^{10} \varphi_k \]

with \( \varphi \) the Daubechies scaling function of order \( p = 2 \).
An interpolating scaling function $\phi$ fulfills

$$\phi(k) = \delta_k, \quad k \in \mathbb{Z}.$$  

Since the Daubechies scaling functions $\varphi_k$ are orthonormal we get an interpolating scaling function $\phi$ (Interpolet) by folding the Daubechies function with itself.

$$\phi(t) = \int_{\mathbb{R}} \varphi(u)\varphi(u - t)du = \varphi \star \overline{\varphi}(t),$$

where

$$\overline{\varphi}(t) = \varphi(-t).$$
The interpolation property can be easily seen:

\[ \phi(k) = \int_{\mathbb{R}} \varphi(u)\varphi(u-k)du = \langle \varphi, \varphi_k \rangle = \delta_k. \]

For the filter \( h^I \) corresponding to \( \phi \) we get

\[ h^I_k = (h \ast \overline{h})_k. \]

To hold the interpolating property for \( j \in \mathbb{Z} \) we omit the normalisation factor:

\[ \phi^j_k(x) = \phi(2^j x - k). \]
Figure: Interpolating scaling function $\phi$ for $p = 2, 3, 4$ and $5$. 
With Interpolets we have a very simple projection $P^J$ onto $V^J$:

$$P^J f = \sum_{k \in \mathbb{Z}} f(x_k) \phi^J_k, \quad x_k = 2^{-J} k$$

This projection is called interpolation projection or collocation projection.
Moments

We want to compute the moments

\[ M_k = \int_{\mathbb{R}} x^k \varphi(x) \, dx = \int_0^{2p-1} x^k \varphi(x) \, dx. \]

By using the refinement equation we get

\[ M_k = \sqrt{2} \sum_l h_l \int_{\mathbb{R}} x^k \varphi(2x - l) \, dx \]

\[ = \frac{\sqrt{2}}{2^{k+1}} \sum_l h_l \int_{\mathbb{R}} (x + l)^k \varphi(x) \, dx \]

\[ = \frac{\sqrt{2}}{2^{k+1}} \sum_l h_l \sum_{j=0}^{k} \binom{k}{j} l^j M_{k-j}. \]
Using the fact that $M_0 = 1$ we get the recursion formula

$$
M_k = \frac{1}{\sqrt{2}(2^k - 1)} \sum_{l=0}^{2p-1} h_l \sum_{j=1}^{k} \binom{k}{j} l^j M_{k-j} \quad \text{for } k > 0.
$$

With the $M_k$ we can easily compute

$$
\langle x^k, \varphi^j \rangle = \int_{\mathbb{R}} x^k \varphi^j(x) \, dx
$$

$$
= 2^{-j k - j/2} \int_{\mathbb{R}} (x + l)^k \varphi(x) \, dx
$$

$$
= 2^{-j k - j/2} \sum_{m=0}^{k} \binom{k}{m} l^m M_{k-m}.
$$
Therefore for any polynomial $\pi$ the integral

$$\langle \pi, \varphi^j_l \rangle = \int_{\mathbb{R}} \pi(x) \varphi^j_l(x) dx$$

is exactly computable. For computing

$$\langle f, \varphi^j_l \rangle$$

where you know the function values of some nodes of $f$, you can do a polynomial interpolation with a polynomial $\pi_f$ and integrate

$$\int_{\mathbb{R}} \pi_f(x) \varphi^j_l(x) dx \approx \int_{\mathbb{R}} f(x) \varphi^j_l(x) dx.$$
Computing $\langle D\varphi^j, D\varphi_k^j \rangle$

$\varphi'$ satisfies a refinement equation, too:

$$D\varphi = \sum_k h_k D\varphi_k^1.$$

We can use this to compute

$$a_k = \langle D\varphi, D\varphi_k \rangle.$$
Computing Integrals

\[ a_k = \sum_l \sum_m h_l h_m \langle D\varphi_l^1, D\varphi_{2k+m}^1 \rangle \]

\[ = 4 \sum_l \sum_m h_l h_{m-2k} \langle D\varphi, D\varphi_{m-l} \rangle \]

\[ = 4 \sum_l (h_l \sum_m h_{l+m-2k}) a_l \]

If you set \( H_{k,l} = 4h_l \sum_m h_{l+m-2k} \) you have the eigen equation

\[ H a = a \]

i.e. you have to compute an eigenvector for the eigenvalue 1. It can be shown that this eigenvector is unique up to normalisation. To find the correct normalisation of the \( a'_k \)'s we use the equation

\[ \sum_k k^2 a_k = 2 \]

which holds for \( p \geq 3 \).
For $a^j_k = \langle D\varphi^j, D\varphi^j_k \rangle$ you have

$$a^j_k = 2^{2j} \langle D\varphi, D\varphi_k \rangle.$$

**Remark:** It holds

$$a^j_k = -\Delta \varphi^j_k(0).$$
Computing $\langle \varphi, \phi_k \rangle$

With the same trick as before you can compute

$$b_k = \langle \varphi, \phi_k \rangle$$

$$= \sum_l \sum_m h_l h_m^I \langle \varphi_l^1, \phi_{2k+m}^1 \rangle$$

$$= 2^{-1/2} \sum_l \sum_m h_l h_{m-2k}^I \langle \varphi, \phi_{m-l} \rangle$$

$$= 2^{-1/2} \sum_l \left( h_l \sum_m h_{l+m-2k}^I \right) b_l.$$
With \( \tilde{H}_{k,l} = 2^{-1/2} \sum_m h^I_{l+m-2k} \) we have the equation

\[ \tilde{H} b = b, \]

i.e. we search an eigenvector of \( \tilde{H} \) for the eigenvalue 1. Here you also have to find a further information to achieve a unique solution. For \( b^j_k = \langle \varphi^j, \phi^j_k \rangle \) we get

\[ b^j_k = 2^{-j/2} b_k. \]
Until now we only considered one dimensional functions. If $\varphi$ is a scaling function we define a $d$-dimensional scaling function $\Phi$ as follows.

$$\Phi(x_1, \ldots, x_d) = \prod_{i=1}^{d} \varphi(x_i)$$

For $k = (k_1, \ldots, k_d) \in \mathbb{Z}^d$ and $j \in \mathbb{Z}$ we define

$$\Phi^j_k(x_1, \ldots, x_d) = \prod_{i=1}^{d} \varphi^{j}_{k_i}(x_i).$$

Then with

$$V^j = \text{span}\{\Phi^j_k | k \in \mathbb{Z}^d\} = \bigotimes_{i=1}^{d} V^j$$

we get a MRA for $L_2(\mathbb{R}^d)$. 

R. Schneider  F. Krüger

Daubechies Wavelets and Interpolating Scalind

November 22, 2007  28 / 1
The wavelet spaces are a little bit more complicated. With $W_0^j = V^j$ and $W_1^j = W^j$ we have

\begin{align*}
\mathbf{V}^J &= \bigotimes_{i=1}^{d} V^J \\
\quad &= \bigotimes_{i=1}^{d} (V^{J-1} \oplus W^{J-1}) \\
\quad &= \bigoplus_{e \in \{0,1\}^d} \bigotimes_{i=1}^{d} W_{e_i}^{J-1}.
\end{align*}

That means that in $d$ dimensions we have $2^d - 1$ different wavelets. For example, for $d = 3$, we have

\[
\begin{align*}
\psi_1 &= \phi \otimes \phi \otimes \psi, \\
\psi_2 &= \phi \otimes \psi \otimes \phi, \\
\psi_3 &= \phi \otimes \psi \otimes \psi, \\
\psi_4 &= \psi \otimes \phi \otimes \phi, \\
\psi_5 &= \psi \otimes \phi \otimes \psi, \\
\psi_6 &= \psi \otimes \psi \otimes \phi, \\
\psi_7 &= \psi \otimes \psi \otimes \psi.
\end{align*}
\]
If we pass to more levels, we can do it in two ways.

1. **isotrop:**

\[
V^J = \bigoplus_{i=1}^{d} V^0 \oplus \bigoplus_{j=0}^{J-1} \bigoplus_{e \in \{0,1\}^d \setminus \{0\}} W^j_{e_i}
\]

2. **anisotrop:** With the notation \( W^{-1} = V^0 \)

\[
V^J = \bigoplus_{j_1=-1}^{J-1} \bigoplus_{j_2=-1}^{J-1} \cdots \bigoplus_{j_d=-1}^{J-1} \bigoplus_{i=1}^{d} W^j_i
\]
Isotropic Decomposition

\[ V^3 \otimes V^3 \]
### Isotropic Decomposition

\[
\begin{array}{c|c}
V^2 \otimes W^2 & W^2 \otimes W^2 \\
V^2 \otimes V^2 & W^2 \otimes V^2 \\
\end{array}
\]
### Isotropic Decomposition

<table>
<thead>
<tr>
<th>$V^2 \otimes W^2$</th>
<th>$W^2 \otimes W^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V^1 \otimes W^1$</td>
<td>$W^1 \otimes W^1$</td>
</tr>
<tr>
<td>$V^1 \otimes V^1$</td>
<td>$W^1 \otimes V^1$</td>
</tr>
<tr>
<td></td>
<td>$W^2 \otimes V^2$</td>
</tr>
</tbody>
</table>
Isotropic Decomposition

\[
\begin{array}{cc}
V^2 \otimes W^2 & W^2 \otimes W^2 \\
V^1 \otimes W^1 & W^1 \otimes W^1 \\
V^0 \otimes W^0 & W^0 \otimes W^0 \\
V^0 \otimes V^0 & W^0 \otimes V^0 \\
\end{array}
\]
Anisotropic Decomposition

\( V^3 \otimes V^3 \)
**Anisotropic Decomposition**

<table>
<thead>
<tr>
<th>$V^2 \otimes W^2$</th>
<th>$W^2 \otimes W^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V^2 \otimes V^2$</td>
<td>$W^2 \otimes V^2$</td>
</tr>
</tbody>
</table>
Anisotropic Decomposition

\[
\begin{array}{ccc}
V^1 \otimes W^2 & W^1 \otimes W^2 & W^2 \otimes W^2 \\
V^1 \otimes W^1 & W^1 \otimes W^1 & W^2 \otimes W^1 \\
V^1 \otimes V^1 & W^1 \otimes V^1 & W^2 \otimes V^1 \\
\end{array}
\]
## Anisotropic Decomposition

<table>
<thead>
<tr>
<th>$V^0 \otimes W^2$</th>
<th>$W^0 \otimes W^2$</th>
<th>$W^1 \otimes W^2$</th>
<th>$W^2 \otimes W^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V^0 \otimes W^1$</td>
<td>$W^0 \otimes W^1$</td>
<td>$W^1 \otimes W^1$</td>
<td>$W^2 \otimes W^1$</td>
</tr>
<tr>
<td>$V^0 \otimes V^0$</td>
<td>$W^0 \otimes V^0$</td>
<td>$W^1 \otimes V^0$</td>
<td>$W^2 \otimes V^0$</td>
</tr>
</tbody>
</table>
Given an equation

\[ Au - f = 0 \]  

for a given function \( f \) and operator \( A \). We are looking for the solution \( u \) in a Hilbert space \( H \). (??) is equivalent to

\[ \langle Au - f, v \rangle = 0 \quad \text{for all } v \in H. \]
Galerkin Scheme

We want to approximate the solution $u$ in a $N$-dimensional subspace of $H$ and denote the approximated solution by $u_N$. Let \( \{v_1, \ldots, v_N\} \) be an orthonormal basis for $V_N$. We are getting the equations

\[
\langle Au_N - f, v_i \rangle = 0, \quad i = \{1, \ldots, n\}.
\]

We make the ansatz $u_N = \sum_{k=1}^{N} \lambda_k v_k$. We define the matrix $A$ and the vector $f$ by

\[
A_{k,l} = \langle Av_k, v_l \rangle, \quad f_k = \langle f, v_k \rangle.
\]

Then we have to solve the linear system

\[
A \lambda = f.
\]
Collocation Scheme

We make the ansatz

\[ u_N = \sum_{k} u(x_k) v_k. \]

Here the \( v_k \) don't have to be orthonormal. Here are the Interpolets \( \phi_{k}^{j} \) senseful. This leads to the equations

\[ (Au_N - f)(x_k) = 0, \quad k = 1, \ldots, N. \]

By defining \( A \) and \( f \) by

\[ A_{k,l} = (Av_l)(x_k), \quad f_k = f(x_k) \]

we get the linear system

\[ Au = f, \quad \text{with } u_k \approx u(k). \]
References
