Wavelet Transforms and Simple Applications

If one looks at the vast variety of approaches to wavelets, it appears to be a vague concept with matrix transforms and functions for the use in compression. This is only the tip of the iceberg. In reality, wavelets offer significant advantages in the field of numerical analysis. Some methods are good for showing theoretical proofs. Other methods are good for computing out solutions for frequency and filtering problems.

Goals for this introductory document into wavelets is to visit some approaches to defining what a wavelet is, some basic theorems, and practical examples. The wavelet definitions approaches a wavelet from both the matrix and convolution methods. Most of the basic theorems presented here are based on the convolution approach to wavelets. It is not to say that the matrix approach is not valid, rather it is simply more complicated than this introduction should be. Lastly the practical examples are based on the Haar Wavelet applied to 1-D and 2-D sources. Such examples are also practical for Daubechies wavelet transform as well.

■ Wavelet Transform Definitions:

As stated, the wavelet transform can be viewed from both a matrix representation which allows many linear algebra and abstract algebra methods to prove theorems about wavelets. Another form is a convolution type of operation. The compact form used by James Walker is simply a special case of the convolution method, and thus James Walker's principles apply to the convolution methods.

In the matrix form, a wavelet matrix is defined as a generalization of a square orthogonal or unitary matrices which are a subset of a larger class of rectrangular matrices. The matrices contain information to define an associate wavelet system. Such matrices are defined in terms of their rank and genus.

Furthermore, Alfred Haar himself pointed out in Zur Theorie der orthogonalen Funktionensystem Math Annual qualities of a Haar Wavelet Matrix. In these cases, the Haar Wavelet Matrix has a genus of one. Such matrices have a one to one mapping to a general wavelet matrix. Such Haar matrices are called the characteristic Haar Matrix. Also a rank 2 Haar Matrix suffices to show all geometric considerations.

The thing that makes the matrix representation difficult is the use of the Laurent series. Also, computationally, any matrix operation is inherently a $O(n^2)$ operation which is not desired in compuationaly intense programs. Convolution operatated versions may offer an improvement.

James S. Walker identified eight properties of wavelets in convolution, which some of them hold in general. These properties covered the following areas: small fluctuations, compaction of energy, scaling squared, linear fluctuations, quadratic fluctations, polynomial fluctuations. These properties allow frequency components to be analyized in a wavelets space as opposed to pure frequency space like Fourier and Laplace transforms allow.

■ Wavelets via Convolution

Wavelets can be defined in a matrix form as $(A \mid D_1 \mid D_2 \mid \ldots \mid D_n)$ where n is the resolution of the wavelet, A is the average component, and D_i is the difference component. This implies that there is the ability to transform a signal into this form and to recover it as well.

Many such as Walker, use a form that eliminates half of the values. This method makes the total number values in the wavelet transform and the original the same. However, there are other means. One method (an over-qualified method) ensures that each component of the wavelet matrix is of the same length as the orignal. It is hoped that properties of both methods can reveal useful mathematical properties.

Another useful property of wavelets via convolution is the simplisity of the operation. The convolution is $O(n(\log(n)))$ in complexity. Where as a matrix multiply is inherently $O(n^2)$. Obviously few operations lead to less complexity.

■ Convolution Operation:

The convolution algorithm is presented as follows:

```
∀i∈[0,M)
        ∀j ∈[0,i)
                n=i-j
                if (0≤n<M)
                        y_i += x_n * h_j
```

This filter simply equates to the mathematical function: $y = x * h = \sum_{k=-\infty}^{\infty} x_{n-k}\, h_n$, which is the convolution operation. As we can see the operation is slightly less than $O(n^2)$. For practical use, the filter is made smaller than the actual signal being analyized. In some cases, the filter may be much smalled than the signal. Size of the filter matters in the features that are being extracted from original signal.

In the case of this convolution operator, the limit is acutally M, not ∞. The value of M is the size of the original signal. Since the resulting vector is the same as the original, the vector is said to be fully qualified. Some forms are made half size discounting half of the information.

To perform a wavelet transform via convolution, each signal is convolved twice. $A_i = A_{i-1} * H_v$ and $D_i = A_{i-1} * H_w$ such that H_v is the Haar scaling vector and H_w is the Haar wavelet vector, respectively. Also, this applies to i∈[2,∞). For i=1, there is a special case, $A_1 = f * H_v$ and $D_1 = f * H_w$.

■ Walker Method and Convolution Method

There is a comparison of this method to James S. Walker's method. This point is to be illustrated by the Haar wavelet. By definition, the difference vector elements are defined:

$d_k = f \cdot W_k$

d_k is the difference value

W_k is the wavelet vector shifted by (k-1) * 2

$k \in [1, \frac{m}{2}]$ s.t. m is the length of f
f is the original signal.

The average vector elements are defined by:
$a_k = f \cdot V_k$
a_k is the average value
V_k is the scaling vector shifted by (k-1)*2
$k \in [1, \frac{m}{2}]$ s.t. m is the length of f
f is the original signal.

Furthermore, the inverse of the wavelet transform defines the original vector as:
$f_{2k} = a_k - d_k$
$f_{2k-1} = a_k + d_k$

This example implies that each A^i and D^i is half the size of the previous elements. In otherwords A^1 and D^1 are both half the size of f in the James S. Walker method. Likewise, A^2 and D^2 are half the size of A^1.

For the convolution method for zero indexed vectors, the even indexed values equated to the a_k and d_k values in Walker's method. Thus the inverse via convolution is:

$f_i = a_i + d_i$ ∀i∈[0,m) that is even
$f_i = a_{i-1} - d_{i-1}$ ∀i∈[0,m) that is odd

■ Example:

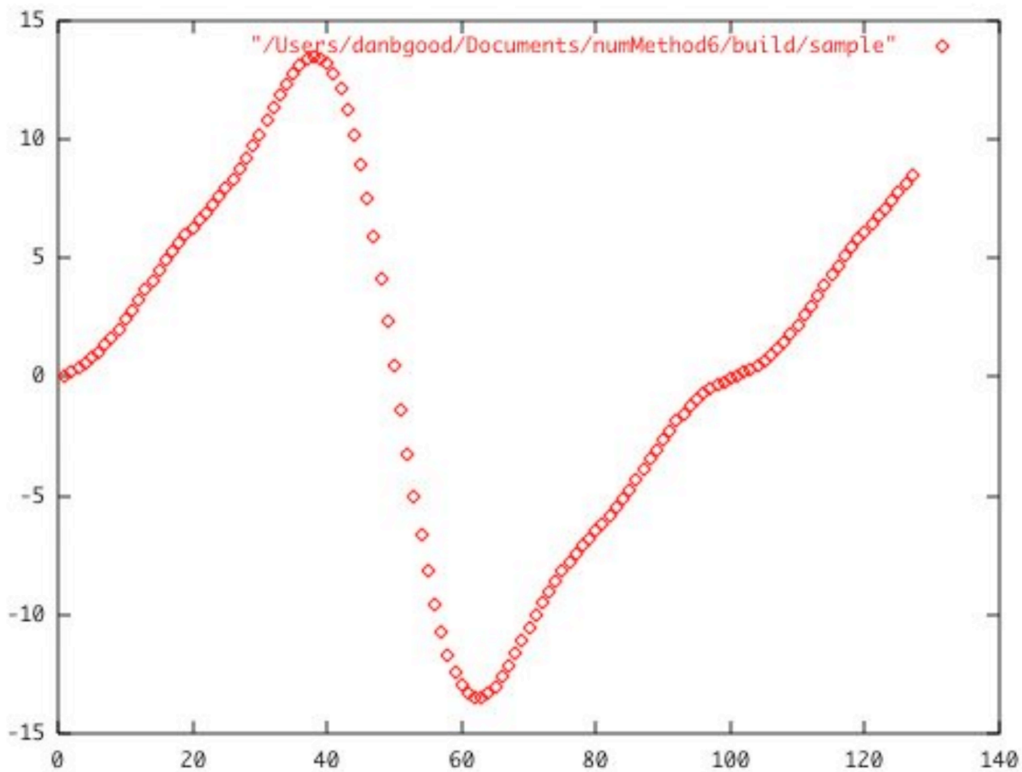
To test these idea, a simple program was made to run this convolution operator on simple functions. Haar matrices were generated by one class. Convolution was placed in another class. Last, the representation in another class. Special classes were provided for the test of the algorithm and plotting of the vectors.

The values chosen for the wavelet and scaling vectors were as followes:
$V = (\sqrt{(\frac{1}{2})})\begin{pmatrix} 1 \\ 1 \end{pmatrix}$
$W = (\sqrt{(\frac{1}{2})})\begin{pmatrix} 1 \\ -1 \end{pmatrix}$

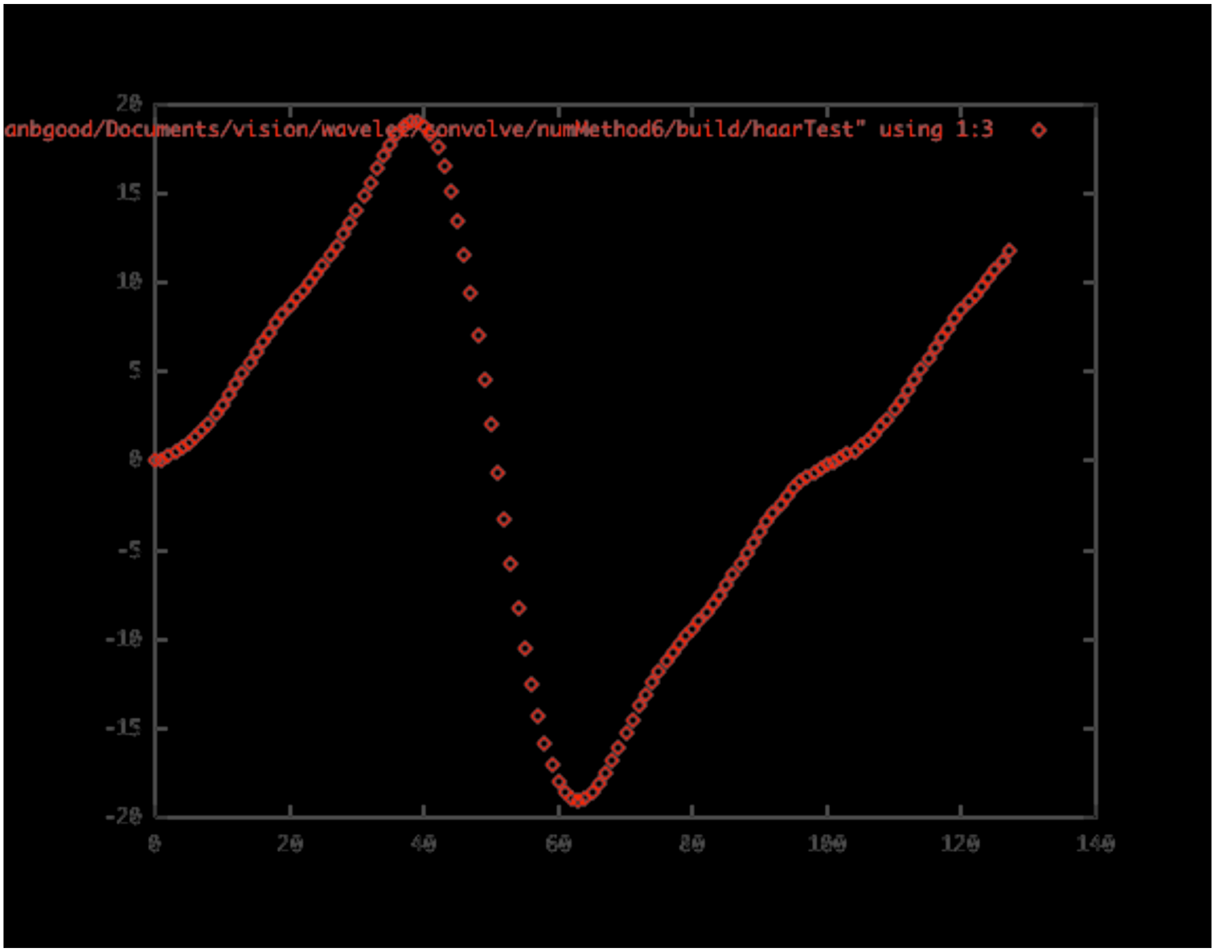
The start of the transform convolves the original signal into vectors R_A and R_D. From this a loop performs the operation on subsequent R_A vectors. In principle this should work. However, R_A has a tendency to grow.

The original signal chosen for test was:
$f(t) = 10 \sin(t) - 5 \sin(2t) + 2 \sin(3t) - \sin(4t)$

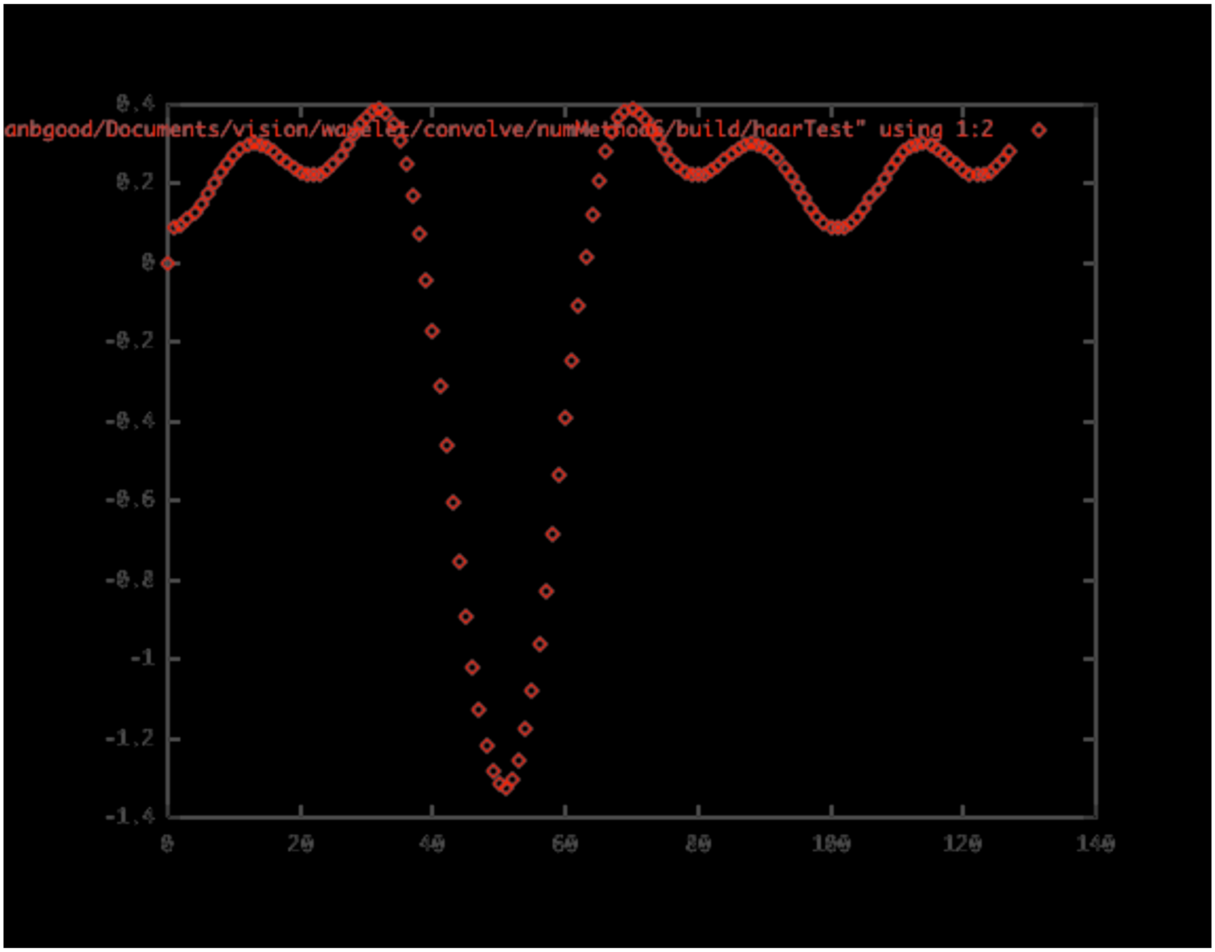
∀t∈[0,128) represented on horizontal axis
f(t) represented on the vertical axis.



The first average maintains the shape, but gains some strength:



Differences are notices mostly ∀t∈(40,60). The rest are not so remarkable.



One curious aspect of this example is that the average signal grows in each iteration. In time is much larger in magnitude than the original. However, the shape generally remains the same. Why is this? The first idea that comes to mind is the scaling value on the scaling vector. Most filters are supposed have loss of energy. However, there is an actual gain of energy between the average versus the previous original. Why?

Walker points out that a majority of the energy is contained with in the average. Subsequent energy would be found in the average of the average of the average ..... Unfortunately, these vectors are much the same size as the original.

Another reason could be the size of the convolution filter. This filter is only 2 elements long. Where as the signal is 64 times its size. Could this be contributing to this difference?