Stanford Exploration Project, Report 75, November 17, 1997, pages 399–410

Stanford Exploration Project, Report 75, November 17, 1997, pages 399–410

Wavelets and seismic processing

Ray Abma and Francis Muir[1]

keywords: *Fourier transform, wavelet*

ABSTRACT

Various wavelet transforms are compared and the fundamentals of wavelet transforms are examined. While wavelet transforms appear to provide much promise for signal processing, we found no apparent application other than some limited data compression possibilities. The absence of useful properties within the transformed domain, such as the derivative property in the Fourier transform, account for the present lack of applications.

INTRODUCTION

Wavelet transforms appear to hold much promise for seismic processing. One important advantage is that while the familiar Fourier transforms are constrained to work globally, or on an entire time sequence, wavelet transformations allow localization of the transform in time.

The basic features of the application of wavelet transforms are translation and dilation of a basic wavelet. Daubechies (1992) reviews these features, including orthonormal bases and frames. Strang (1989) introduces these features in a shorter presentation.

We examined wavelet transforms for applications to seismic processing and found that these transforms now appear to have limited application because of the lack of useful properties within the transformed domain. There are some data compression applications, but seismic data are not well suited to these.

The sliding Fourier transform, the Haar transform, and Daubechies D_4 wavelet are examined and applied. Examples of these transforms are shown to allow a better understanding of the limits of each technique.

[1]**email:** not available

SAMPLE WAVELET APPLICATIONS

Sliding Fourier Transform

The sliding Fourier transform is widely used for extracting time-dependent spectra from time series and has localization properties similar to those of the wavelet transformations. It is described here to provide a contrast to the wavelet transforms. To create a sliding Fourier transform, a time series is cut into a series of smaller series, which are then individually Fourier transformed. This windowing creates a transformation that is localized in time.

To provide an example of this localizing feature of the sliding Fourier transform, the trace with two spikes shown in Figure 1 is shown broken up into overlapping windows in Figure 2. Each window is then Fourier transformed. This process creates the sliding Fourier transform displayed in Figure 3.

Figure 1: A trace with two spikes. ray2-spike [NR]

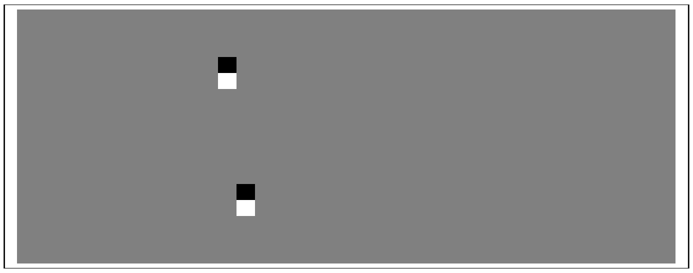

Figure 2: The trace in the previous figure windowed. Notice that the windows are overlapped and the spikes appear in two of the windows. The vertical axis is time (increasing upward) and the horizontal axis is the time corresponding to the window location, increasing to the right. ray2-wspike [NR]

Figures 4 to 6 show the same sequence applied to a seismic trace. Figure 4 shows a single seismic trace with a high amplitude zone and a weak amplitude zone. Figure 5 shows this trace windowed into overlapping windows to allow the localization of the Fourier transform. The high amplitude zone is now separated from the low amplitude zone, allowing the Fourier transforms to operate within limited time ranges. Figure 6 shows the resulting sliding Fourier transform. The figure shows a time separation into

Figure 3: The traces in the previous figure Fourier transformed. This is the sliding Fourier transform. The vertical axis is frequency (increasing upward) and the horizontal axis is the time corresponding to the window location. ray2-fspike [NR]

zones of low amplitude and high amplitude and shows a zone of high frequency between the two zones. This result may be compared to the standard Fourier transform, which operates globally on the trace and does not allow the separation of transformed events in time.

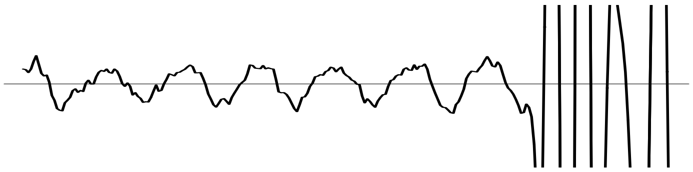

Figure 4: A seismic trace with a low-energy zone and a high-energy zone. ray2-seis [NR]

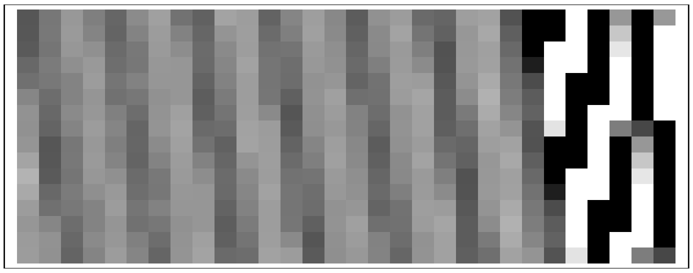

Figure 5: The trace in the previous figure windowed. Notice that the windows are overlapped and the high amplitude zone appears on the right side. ray2-wseis [NR]

As the windows become smaller in the sliding Fourier transform, the frequency resolution becomes poorer. Resolution in time may be increased, but only at the expense of resolution in frequency. Another problem with the sliding Fourier transform is that the output is much larger than the input. The sliding Fourier transform is useful for analyzing frequency with respect to time, but applying an operation in the

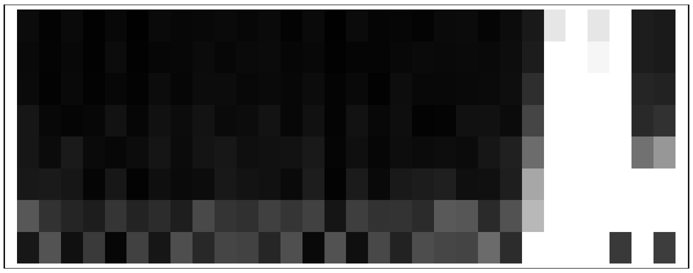

Figure 6: The traces in the previous figure Fourier transformed. This is the sliding Fourier transform. The vertical axis is frequency (increasing upward) and the horizontal axis is the time corresponding to the window location. ray2-fseis [NR]

sliding-Fourier domain requires many more operations than the same application in the Fourier domain.

The Haar Transform

The Haar transform is the simplest of the wavelet transforms. This transform cross-multiplies a function against the wavelet shown in Figure 7 with various shifts and stretches, much like the Fourier transform cross-multiplies a function against a sine wave with two phases and many stretches.

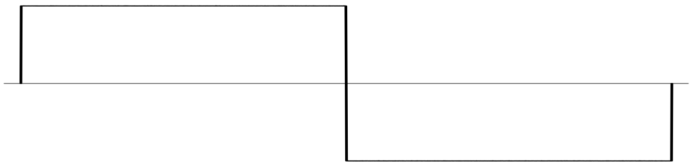

Figure 7: The Haar wavelet. ray2-haarplot [NR]

As an example of a Haar transform, consider transforming the single seismic trace shown in Figure 8. The Haar transform of this trace is shown in Figure 9. Notice that the trace consists of two zones, a weak zone on the left side, and a strong zone on the right side. We see this pattern of strong and weak zones reflected as octave zones within the transform shown in Figure 9.

The samples in the Haar transform shown in Figure 9 are coefficients that describe the decomposition of the trace in Figure 8. A simple example of the Haar decomposition is taken from Strang(1989). If f is a 4-sample trace, then

$$f = \begin{pmatrix} 9 \\ 1 \\ 2 \\ 0 \end{pmatrix} = 3 \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} + 2 \begin{pmatrix} 1 \\ 1 \\ -1 \\ -1 \end{pmatrix} + 4 \begin{pmatrix} 1 \\ -1 \\ 0 \\ 0 \end{pmatrix} + 1 \begin{pmatrix} 0 \\ 0 \\ 1 \\ -1 \end{pmatrix}, \tag{1}$$

and the coefficients of the transform are $\begin{pmatrix} 3 \\ 2 \\ 4 \\ 1 \end{pmatrix}$.

The first sample in Figure 9 contains the coefficient that describes the D.C. component of the trace. The next sample contains the coefficient that describes how a single Haar wavelet shown in Figure 7 cross-multiplies the entire trace. Then, the next two samples describe the two Haar wavelets that cross-multiply two-halves of the trace; one cross-multiplying the first half of the trace, the other cross-multiplying the last half of the trace. This halving of the wavelet and of the trace continues until the Haar wavelets are two samples long, and the number of coefficients required to describe the cross-multiplication are half the trace length. On the right side of Figure 9, in the last half of the trace, the amplitude pattern of the original trace is clearly reflected. This pattern may also be seen within the other octave zones.

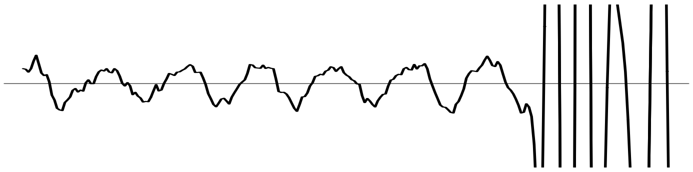

Figure 8: A seismic trace with a low-energy zone and a high-energy zone. ray2-trace [NR]

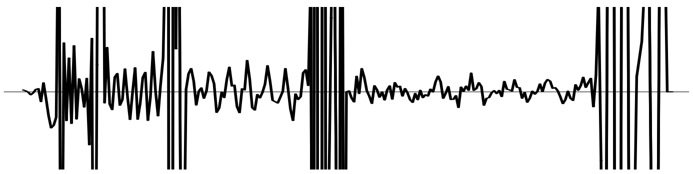

Figure 9: The Haar transform of the seismic trace shown in the previous figure. ray2-haartrace [NR]

An alternate and perhaps more understandable method of displaying the results of a wavelet transform is in the two-dimensional display shown in Figure 9. The vertical axis is the time axis corresponding to the time axis of the trace. The horizontal axis corresponds to the sizes of the Haar wavelets. Where the longer operators have barely visible coefficients in Figure 9, Figure 10 gives more emphasis to the long-operator coefficients and maintains the relative time scales within each set of coefficients.

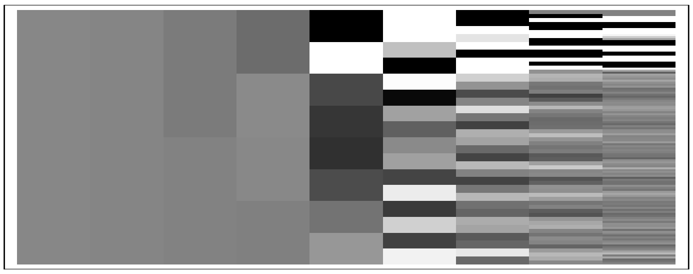

Figure 10: The alternate display form of the Haar transform. The horizontal axis corresponds to the Haar wavelets sizes, and the vertical axis corresponds to the time axis of the trace. Notice that time increases vertically, opposite to the standard seismic display convention. This figure shows the same information as Figure 9. ray2-dhaartrace [NR]

The algorithm is simple, however there is not an obvious use for the transformed data. The data is localized in time, but the frequency separation is poor when compared to the sliding Fourier transform.

MALLAT'S TREE

Mallat's Tree, shown in Figure 11, may be considered in three distinct ways.

- An algorithm for transforming data
- An algorithm for generating filters
- An algorithm for generating the roots of filters

In this section we discuss these three aspects.

The tree as a data algorithm

A fast algorithm for wavelet transformations uses the Mallat tree as seen in Figure 11. The algorithm involves two wavelets, low-pass and high-pass, with the one derived from the other by changing the sign of the even coefficients.

In Figure 11, the input data series is represented by the circle marked 'input'. The result referred to as Lo-1 is from an application of the low-pass filter with a decimation. The low-pass filter for the Haar transform is the 2-term low-pass operator in Figure 12, and the low-pass filter for the D_4 transform is 4-term low-pass operator in Figure

13. Hi-1 comes from filtering with the corresponding high-pass filters. These filters are illustrated by the 2-term high-pass operator in Figure 12, and the 4-term high-pass operator in Figure 13. Hi-1 is now set aside, and the data in Lo-1 becomes the input for the next step. Elsewhere in this report, Schwab expands on how the filters are calculated (Schwab, 1992).

At each step, the high-pass output is saved and becomes part of the transformed output, and the final lo-pass output makes up the rest of the output.

The tree as a filter-generating algorithm

We have seen that the output at any stage is produced by a sequence of identical filtering and decimating operations. An alternative algorithm is to calculate the single long filter and the single multiple decimation that would produce exactly the same result. Here we are not operating on data but on the wavelet filters themselves. At each stage, the new low-pass filter is the old low-pass filter convolved with the basic low-pass filter decimated by a factor of 2^n, where n is the stage. Similarly the new high-pass filter is the old low-pass filter convolved with the multiply-decimated basic high-pass filter.

The tree as a root-generating algorithm

Looking at the roots of sequences of polynomials is always interesting, and our sequence of filters just described is no exception. Again, Mallat's Tree can be used with these three rules:

- Convolving two polynomials is equivalent to forming the union of their roots.
- Filtering decimated data is equivalent to filtering the undecimated data with a zero-padded filter followed by decimating the output.
- The roots of a filter zero-padded between every coefficient are the square roots of the roots of the same unpadded filter.

With these rules in place, the roots of the new low-pass filter are the roots of the old low-pass filter in union with the roots of the basic low-pass wavelet zero-padded by the number of stages. This is illustrated in Figures 12 and 13.

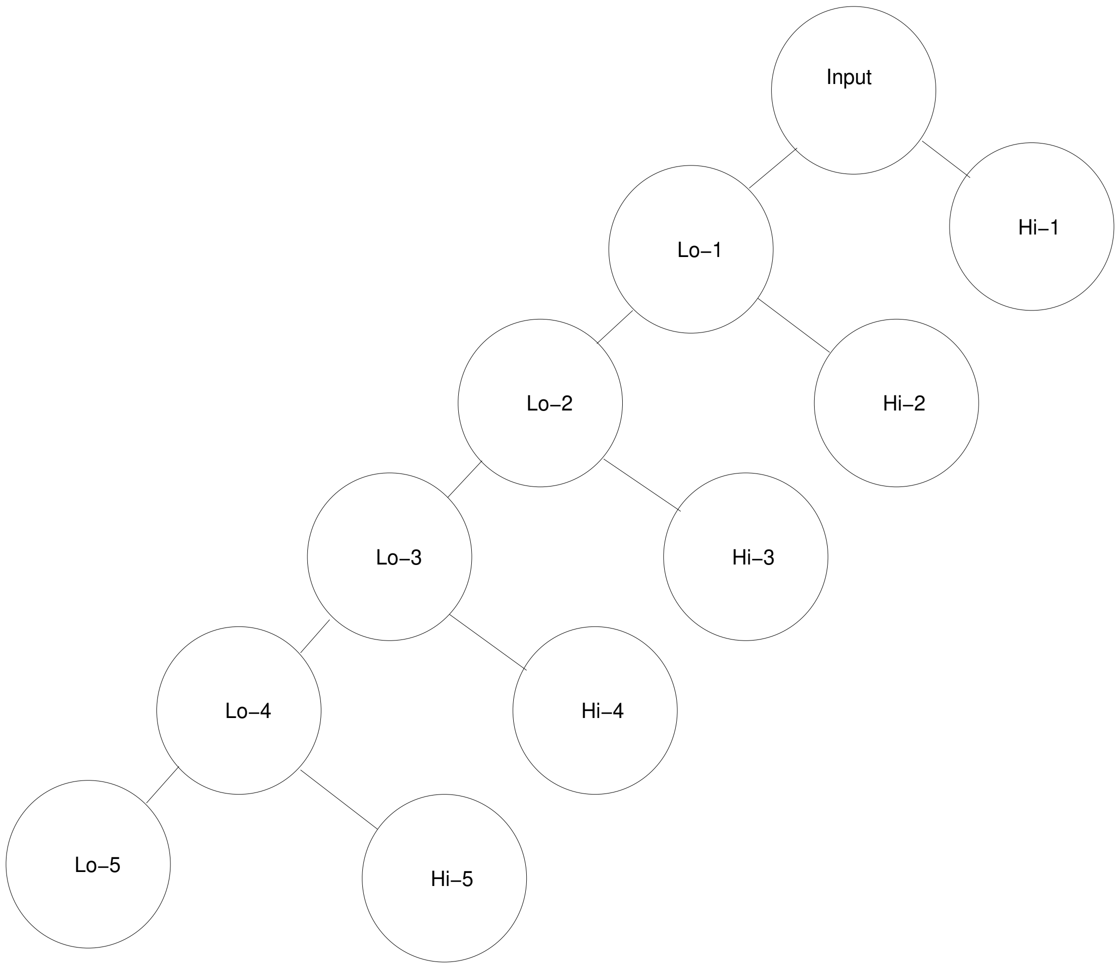

Figure 11: Mallat's tree. ray2-tree [NR]

CONCLUSIONS

We now have a better understanding of wavelets, and if applications come along, we'll be ready to implement them.

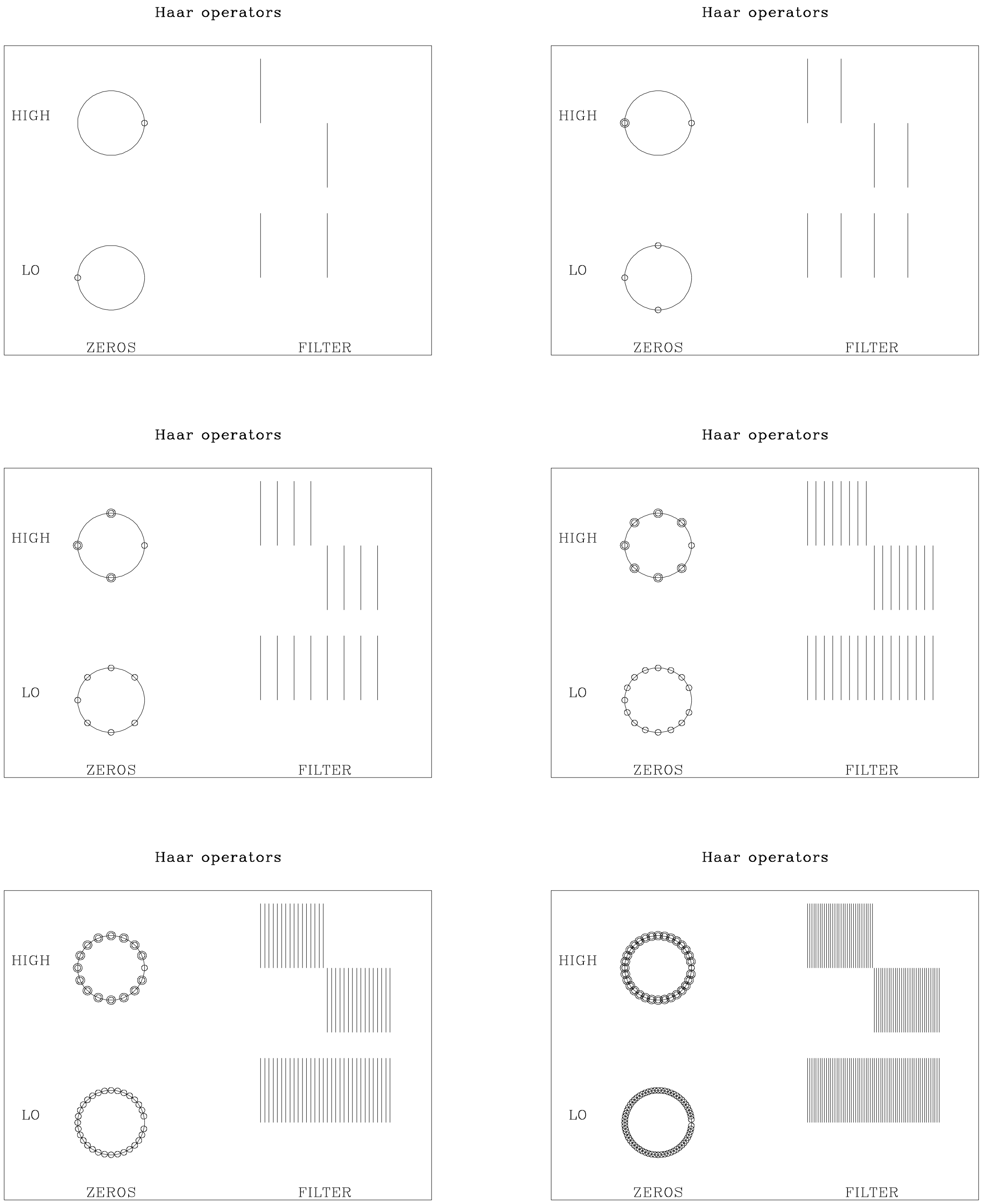

Figure 12: The wavelets and the zeros for the Haar wavelets. In each plot, the top row shows the high-pass Haar wavelet on the left and its zeros on the complex plane on the right. The bottom row shows the low-pass Haar wavelet and its zeros. Multiple zeros have multiple circles drawn around them. In the electronic version of this paper, pressing the button starts a movie of these figures. ray2-haarwave [ER]

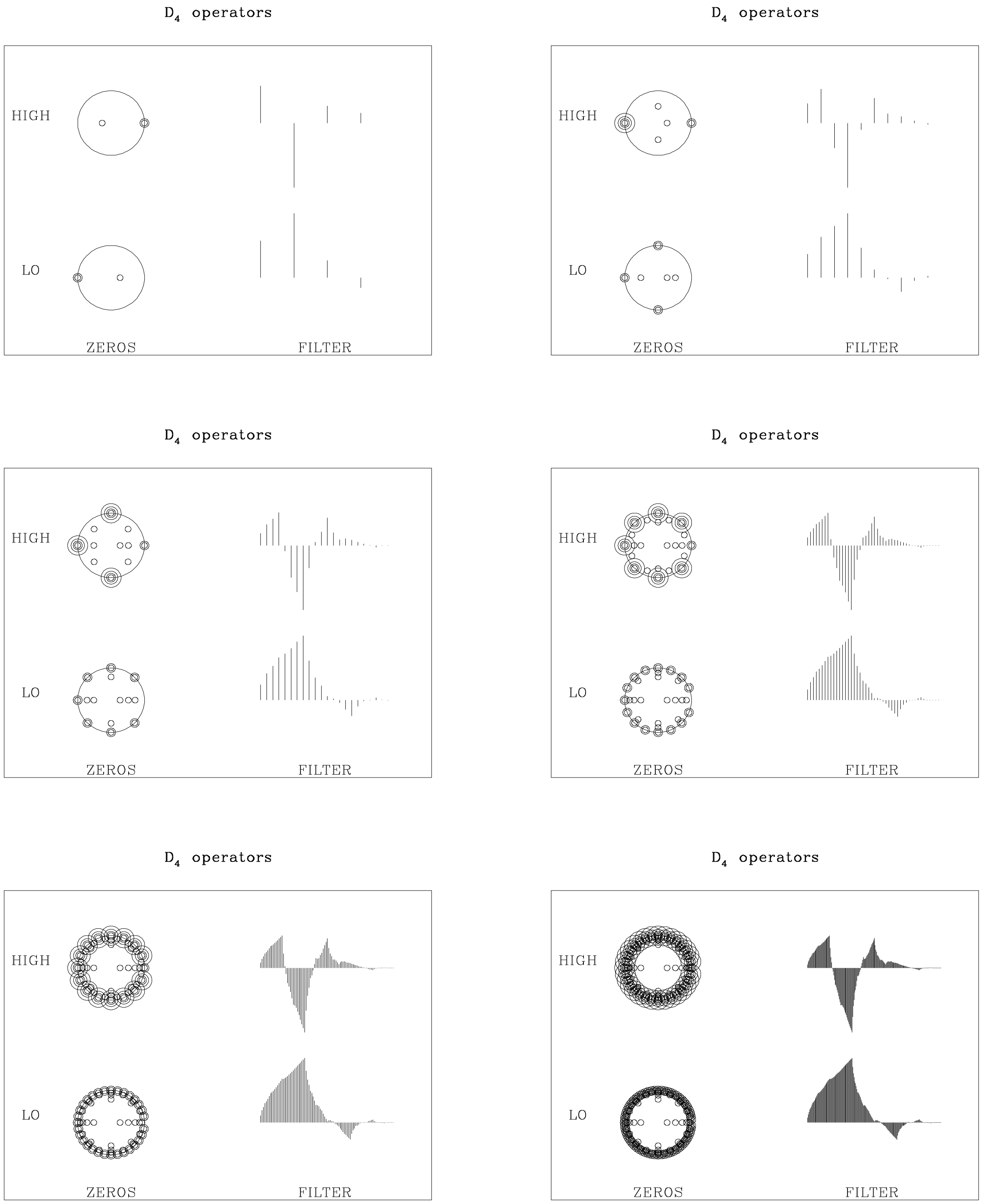

Figure 13: The wavelets and the zeros for various forms of D_4. In each plot, the top row shows the high-pass D_4 wavelet on the left and its zeros on the complex plane on the right. The bottom row shows the low-pass D_4 wavelet and its zeros. Multiple zeros have multiple circles drawn around them. In the electronic version of this paper, pressing the button starts a movie of these figures. $\boxed{\text{ray2-wave}}$ [ER]

ACKNOWLEDGEMENTS

Lin Zhang was generous in taking the time to show us many of the tricks of the vplot plotting software.

REFERENCES

Daubechies, I., 1992, Ten lectures on wavelets: Society for Industrial and Applied Mathematics.

Schwab, M., 1992, Transformation of discrete arrays by Daubechies wavelets: SEP–**75**.

Strang, G., 1989, Wavelets and Dilation Equations: A brief introduction: SIAM Review, **31**, No. 4, 614–627.