Differential operators 1: The first derivative

John C. Bancroft

ABSTRACT

Differential operators are used in many seismic data processes such as triangle filters to reduce aliasing, finite difference solutions to the wave equation, or wavelet correction when modelling with diffractions or migrating with Kirchhoff algorithms. Short operators (two to nine samples) may be quite accurate when the data are restricted to locally low order polynomials, but may be inaccurate in other applications.

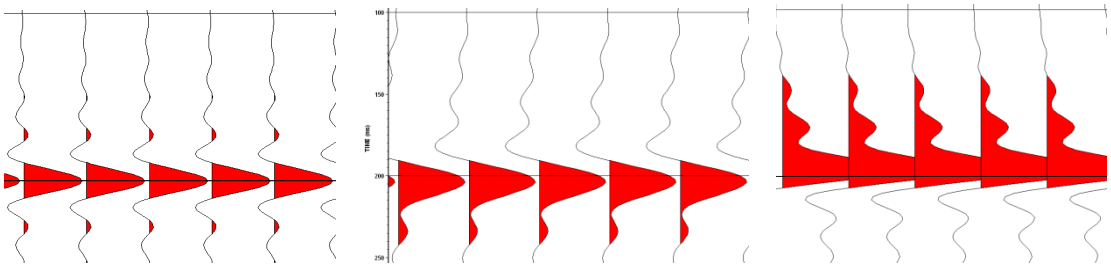
Because of the size of the contents, this topic is divided into three papers, the first reviews operators for the first derivative, the second paper examines the second derivative, and the third paper examines the square-root derivative that is also referred to as the rho filter. Theses papers examine various methods for computing the operators and compare their size with their accuracy.

Four approximations to the derivative are evaluated; an ad hoc windowed spectral definition, a polynomial approximation, the Remez approximation, and a recursive approximation. Significant details of the derivations have been included for clarification.

APPLICATIONS

Rho filter

Modelling and migration should produce no change in a horizontal event, however some of these processes distort the wavelet. This distortion is illustrated in Figure 1 that shows in (a) a portion of a horizontal event created with a zero-phase wavelet. After modelling with diffractions this horizontal event should remain the same as the input, but there is a phase distortion as illustrated in (b). When the event in (a) is migrated with a 2D Kirchhoff algorithm, the event becomes that illustrated in (c). These wavelet distortions are corrected using a Rho filter (operator) that corrects the shape of the wavelet back to the zero phase shape of (a). The Rho filter modifies the phase and applies a taper to the amplitude spectrum.



a) **b)** **c)**

FIG. 1. An input event in a) is shown in b) after modelling with diffractions, and in c) after 2D Kirchhoff migration.

Fast filtering

In this application, fast filtering is performed in the time (or space) domain with simple operators that are much faster than the equivalent convolution. A specific application applies to Kirchhoff migrations where the same data may be locally filtered many times with different size filters to prevent aliasing. An ideal antialiasing filter would be a box-car shape in the frequency domain or a sin x/x shape in the time domain. To save run time, practical applications may use a filter with the shape of a box car, a triangle, or a bell shape made from segments of a quadratic polynomial.

Fast filtering is a process that simplifies the convolutional filter to one that sums a few samples. The filter operator must be composed of segments constructed with the same order of a polynomial such that differentiation by n times, reduces the operator to a number of impulses or delta functions. The input trace is then integrated n times and then convolved with the delta functions, which becomes the summation of a few samples. The result is identical to conventional convolution.

Consider a triangular shaped filter operator in Figure 2 that spans samples from i = -7 to + 7, with 15 samples whose amplitudes increase linearly from i =-7 to 0, then decrease from i =1 to 7. Convolution requires the multiplication of the input samples by the 15 amplitudes of the filter samples and the results summed to produce one filtered sample. However, the triangular shaped wavelet may be differentiated once to produce two boxcars with amplitude +1.0 from i =-7 to 0 and -1.0 from i = 1 to 7. Differentiating again produces three delta functions +1.0 at i = -7, -2 at i = 0 and +1 at i = 7. Convolution now requires only summing the input samples that correspond to -7, 0, and +7 of the filter operator. Note that the triangle filter can be of any length, but will always be reduced to three delta functions.

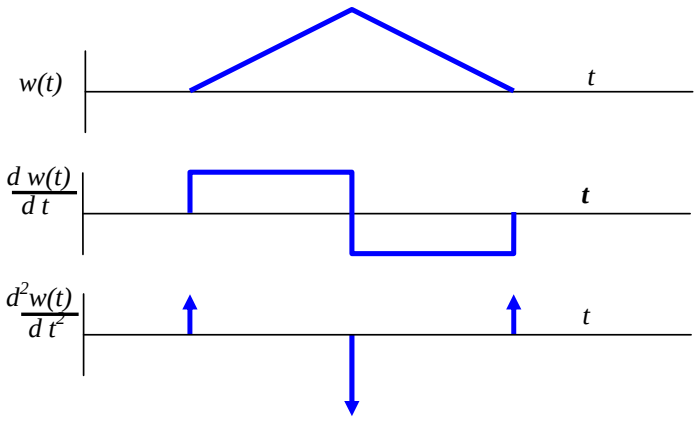


FIG. 2. A triangle wavelet with its first and second derivative.

For the triangular filter, the traces of input data are integrated twice, but the impulse filters may be applied may times at all location and each application may have a different sized filter.

Finite difference solutions

Digital solutions to various forms of the wave-equation require accurate finite difference approximations to the first and second order differentials. The finite difference approximations to the second order differential are usually well behaved, however, the finite difference approximations to the first order differential may provide a greater challenge.

Consider the two point approximation to the derivative (-1, +1) that is typically used. The differentiated sample should be located mid way between the two input samples. Assuming we use the same sample locations of the input trace, the differentiated sample can be located at either the first or second location, (forward or backward solutions) with both locations producing a linear phase error. We could go to a (-1, 0, 1) operator to give us the correct phase, but what happens to the frequency response?

SOME BASIC ASSUMPTIONS

I assume an array of data f_n, that I call a trace, is sampled from a continuous function $f(t)$ defined in time, and that the discrete Fourier transform is F_n. The trace can be in either time or distance transforming into the frequency or wavenumber domains. I will assume the trace to be in the time domain and refer to the transform domain parameters as frequency.

I will assume that the input data has been sampled with a Nyquist frequency well above the maximum frequencies in the trace. The data is then locally smoothed and can be fitted locally with a low order polynomial.

The complex Fourier transform will be used with the imaginary part of the trace zero. The resulting two sided transform will be displayed with the positive frequencies occupying the first part of the display, and the negative frequencies displayed in the second half. This allows examination of the data at the Nyquist location. Many of the displays use sample point numbers for the horizontal axis, especially with the number of complex samples N_{fft} = 512. Sample numbers from 1 to 256 will roughly correspond to frequencies from 0 to 250 Hz.

Factors such as vectorized code do affect the computational times, however old rules of thumb used to claim that the time domain filter was faster when there are less than 30 samples in the convolution wavelet, or that the Fourier transform domain method may be faster if there are more than 50 samples in the wavelet. I will assume that a convolution operator (wavelet) with less than nine samples will be a faster in the time domain, and estimating these wavelets is the focus of this paper.

Derivatives will assume unit sample interval in the time domain.

I use $\sqrt{-1} \equiv j$, and $\omega = 2\pi f_{req}$.

The displayed results are created using MATLAB with code

```
\2008-Matlab\DifferentialOperatorFirst.m
\20008-Matlab\DifferentialRecursiveOperator.m
```

THE FIRST DERIVATIVE

Differentiation is a filtering process and can be conducted in time (space) domain or in the Fourier transformed domain. Typically a short operator is considerably faster in the time domain, however the transformed domain may be more accurate or even faster when combining other processes that require the frequency domain.

Defining a short operator that is accurate for a given purpose may be challenging and a number of methods will be considered. The methods are:

- defined accurately in the transformed domain, then inverse transformed back to the time domain where it is then windowed to a few samples.
- defined using a polynomial assumption combined with the Taylor series,
- using the REMEZ exchange algorithm, and
- using a recursive filter.

I will assume that the derivative in the discrete format is only accurately defined in the frequency domain. Consequently, the short operators are compared and evaluated by comparing their amplitude spectrums.

Derivatives in the Fourier transformed domain

Differentiation is well understood in the continuous realm. Assume the trace $f(t)$ is Fourier transformed into $F(\omega)$, i.e.,

$$\mathrm{F}\{f(t)\} \Leftrightarrow F(\omega). \tag{1}$$

Texts on signal processing show that taking the Fourier transform of the derivative of $f(t)$ will give

$$\mathrm{F}\left\{\frac{d\ f(t)}{dt}\right\} \Leftrightarrow j\omega F(\omega), \tag{2}$$

where $j\omega$ is a filter that performs the derivative in the frequency domain by multiplication. The complex number $j=\sqrt{-1}$ produces a ninety degree phase shift and ω provides amplification of the amplitudes of $F(\omega)$ proportional to the frequency.

The $j\omega$ filter is has only imaginary values with the real components all zero. The blue line in Figure 3 illustrates these imaginary values. The samples are rotated in a form suitable for using the Fast Fourier transform (FFT) and in this figure, N_{fft} = 512. The positive frequency samples are defined from 0 to $N_{fft}/2$, the Nyquist sample is at $1+N_{fft}$, and the "negative" frequencies are defined from $2+N_{fft}/2$ to N_{fft}.

This example uses 512 complex samples that is quite close to a common geophysical sampling frequency of 500 Hz, enabling an approximation in the lower half of the spectrum that matches sample number with frequency in Hz., i.e. the maximum frequency at sample number 256 is 250 Hz.

The sharp discontinuity of the spectrum at the Nyquist frequency, causes excessive oscillations in the time domain response, so a taper has been applied to the higher frequencies to minimize this effect. A quarter cycle of a cosine was used as a taper that started at 80% of the Nyquist frequency f_{nyq}. The taper function is scaled by 100, and shown in Figure 3 as the green line. The resulting tapered $j\omega$ is shown by the red "+" signs. The taper could start at a lower frequency, say 50% f_{nyq} as typical seismic data would be below 100 Hz, but I will leave it at 80% to allow applications with data that have a higher frequency content.

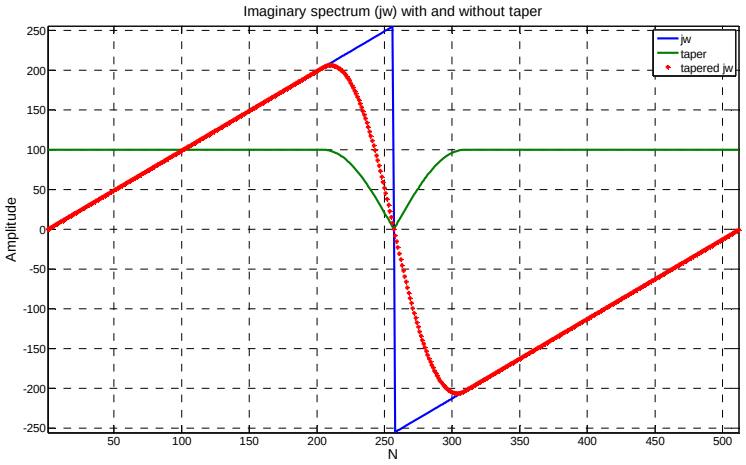


FIG. 3. Imaginary spectrum of $j\omega$, a tapered version of $j\omega$, and the taper scaled to 100.

A discrete Fourier transform, that is complex in both time and frequency will be used. The time domain trace will have N_{fft} complex samples, but only the real part of the complex value will form samples of the trace. That will allow viewing both sides of the complex spectrum, especially at the Nyquist frequency.

I also assume the frequency domain definition of the derivative to be more accurate than a finite difference approximation in the time domain. Note however, that a short time window definition may be more accurate and or applicable when applied to wavelet type data that forms our seismic data.

DIFFERENTIAL OPERATORS DERIVED FROM THE SPECTRAL DEFINITION.

Figure 4 shows the defined spectrums with a smaller N_{fft} = 64. It is easier to identify the samples with this smaller number however the wraparound effect is more prominent. (An actual operators could defined with , N_{fft} = 1024 to minimize the wraparound effect.)

The inverse Fourier transform (IFT) of the frequency domain definition of the derivative $j\omega$ is shown in Figure 5. In this figure, the data is shifted with time zero, t = 0, at the central location, with the sample shown with green + signs. Note:

- the ringing nature of this time domain operator,
- the central three samples are (-1, 0 +1), and
- and that is in a form ready for convolution.

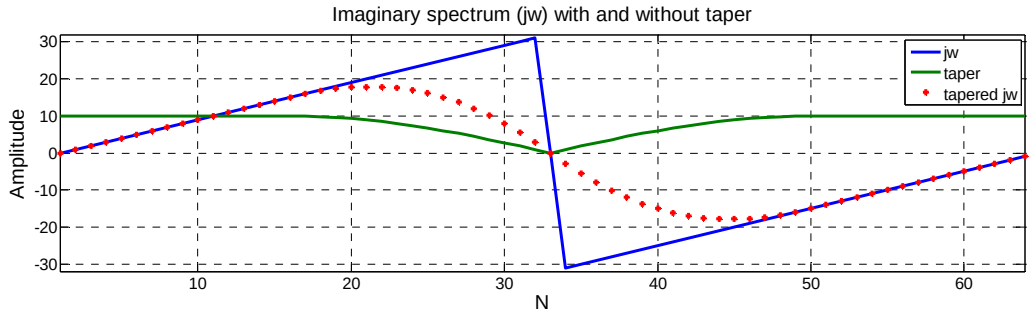


FIG. 4. Plot of the derivative $j\omega$ in the frequency domain. The blue line represents the amplitude of the imaginary values. The Nyquist sample is in the center, and Nfft = 64.

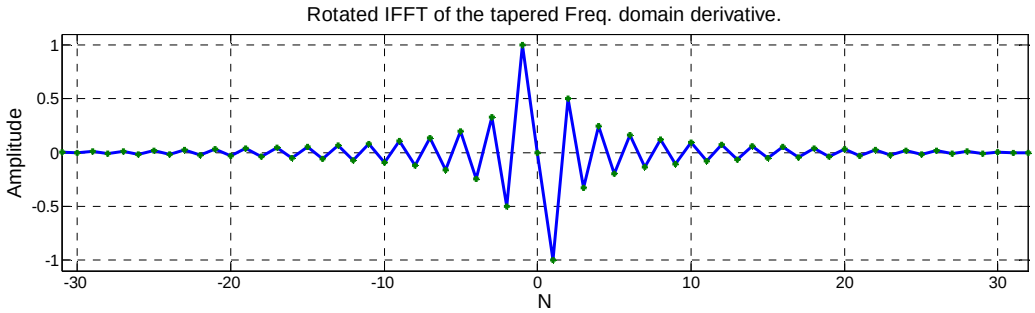


FIG. 5. Inverse Fourier transform of the derivative that was defined in the frequency domain.

The central values for N_{fft} = 1024 with no taper and no window are:

-0.2500	0.3333	-0.5000	1.0000	0	-1.0000	0.5000	-0.3333	0.2500

The ringing nature of the time domain operator is due to the large change in amplitudes of the frequency samples on either side of the Nyquist frequency. A cosine taper was applied to the upper half of the frequency samples as illustrated by the red "+" signs in Figure 4. Note that the frequency samples tend to zero around the Nyquist frequency. The resulting IFT of the tapered spectrum is shown in Figure 6.

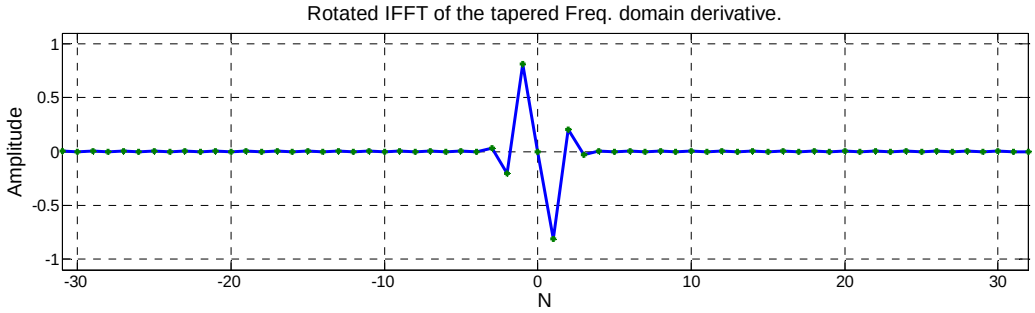


FIG. 6. Inverse Fourier transform of the derivative defined using the tapered frequency domain, Nfft = 64 and taper starting at 50% of the Nyquist frequency.

Note the smoother operator in Figure 6 and that the sample tend to zero much faster than in Figure 5. The amplitudes of the two sample about the Nyquist frequency have also been reduced. These amplitudes are virtually independent of the size of the number

of sample used as illustrated in Figure 7 where N_{fft} = 1024 but displayed with samples from -25 to +25. The taper used in this example started at 50% of the Nyquist frequency, and an "eyeball" evaluation indicated that the seven central samples may produce an adequate operator.

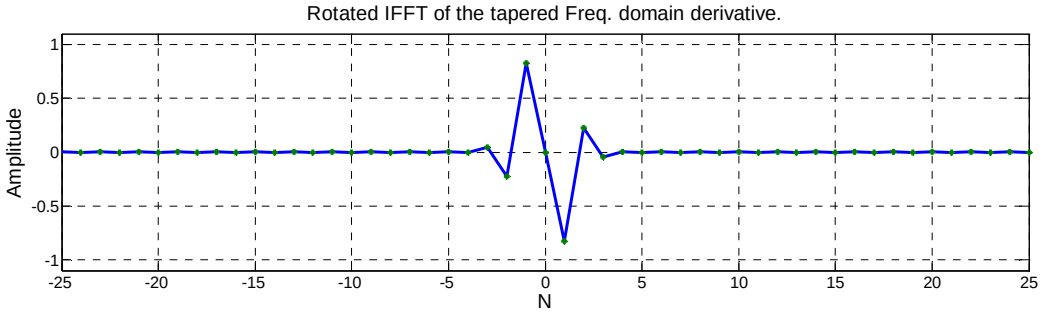
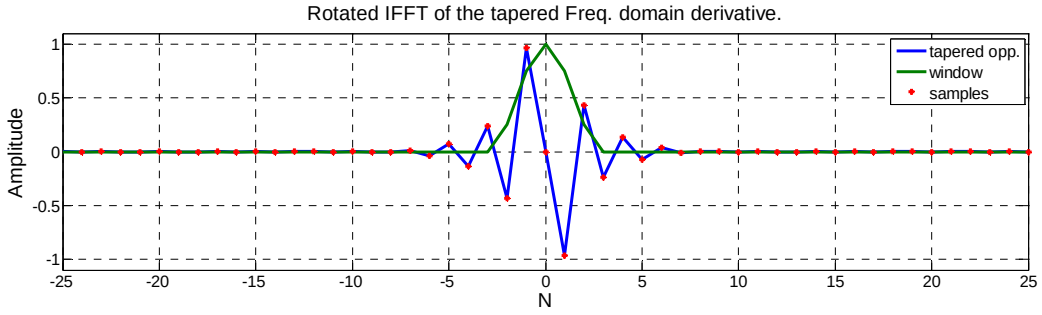
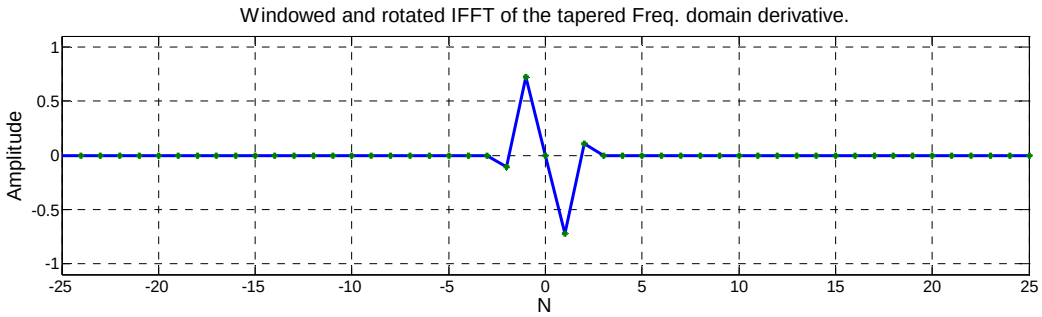


FIG. 7. Inverse Fourier transform of the derivative defined using the tapered frequency, Nfft = 1024 and taper starting at 50% of the Nyquist frequency.

Figure 8a shows the result of changing the taper to 80% of the Nyquist frequency. There is more ringing and thirteen samples may provide an adequate operator. A five sample was applied to the operator in (b), with the result shown in (c).



a)



b)

FIG. 8. Inverse Fourier transform of the derivative in a) defined using the tapered frequency, Nfft = 1024, taper starting at 80% of the Nyquist frequency, and a 5 point window. Part b)shows the tapered result.

The numerical values of the five point operator are shown below.

-0.1085	0.7243	0	-0.7243	0.1085

An eleven point operator with an 80% taper is shown in Figure 9 with the sample values shown below the figure.

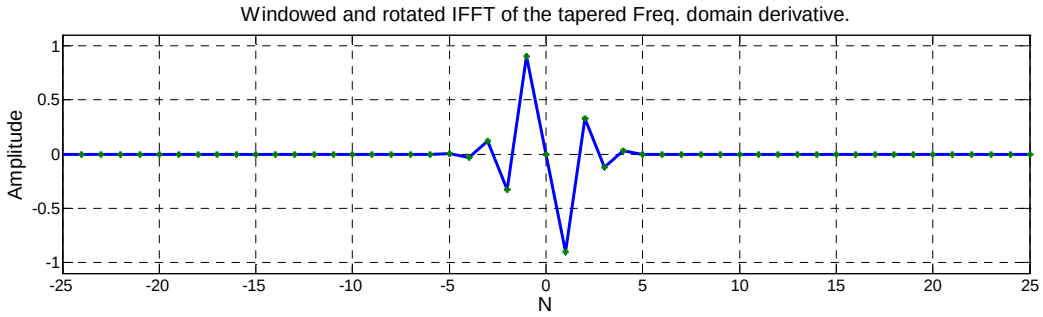


FIG. 9. Inverse Fourier transform of the derivative defined using the tapered frequency, Nfft = 1024, taper starting at 80% of the Nyquist frequency, and an 11 point window.

0.0050	-0.0344	0.1205	-0.3256	0.9011	0	-0.9011	0.3256	-0.1205	0.0344	-0.0050

TIME DOMAIN OPERATORS FOR THE FIRST DERIVATIVE

A starting point for estimating the derivative of a function $f(t)$ is the ratio of the rise over the run as illustrated in Figure 2, i.e.,

$$\left.\frac{d\ f(t)}{dt}\right|_{-t_7+t_8\to 0} \approx \frac{-f(t_7)+f(t_8)}{-t_7+t_8}, \tag{3}$$

where, for a continuous function, we assume the time interval $(-t_7+t_8)$ tends to zero. For a discrete time series we are restricted to times defined by the sample interval.

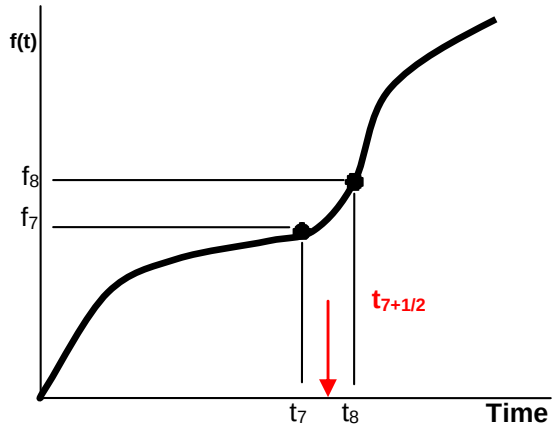


FIG. 10. Defining an approximate derivative from samples on a trace.

Most data is now in a digital form where the samples are uniform in time (or space) and we define an "operator" (-1, 1) to take the difference of two values [-*f*(7) + *f*(8)]. The scaling of the derivative is completed by dividing the difference by the time increment [$t_8 - t_7$], though this scaling is often ignored when defining the operator.

For a sampled sequence of numbers, the value of the approximate derivative should be located centrally between the two time samples, i.e.,

$$\frac{d\ f\left(t_{7+1/2}\right)}{dt} \approx \frac{-f\left(t_7\right)+f\left(t_8\right)}{-t_7+t_8}. \tag{4}$$

Since there are no fractional samples in the sampled data, we place the differentiated value at either t_7 or t_8, providing us with the forward or backwards approximations. Consequently, these approximations will include a positive or negative half sample shift of the data.

Forward difference:
$$\frac{d\ f\left(t_7\right)}{dt} \approx \frac{-f\left(t_7\right)+f\left(t_8\right)}{-t_7+t_8}. \tag{5}$$

Backward difference:
$$\frac{d\ f\left(t_8\right)}{dt} \approx \frac{-f\left(t_7\right)+f\left(t_8\right)}{-t_7+t_8}. \tag{6}$$

The defined operators are directly applied to the data, and should be considered a correlation operator if applied to all the samples in the trace. When using convolution, the defined operator will be time reversed, i.e. (1, -1).

The central differential operator (-1, 0 1) may also be familiar, but usually assumes that the trace is sampled with a Nyquist frequency that is well above the maximum frequency.

POLYNOMIAL DEFINITION OF OPERATOR

A continuous function or curve can be represented by a polynomial with a variable *x* such as

$$f\left(x\right) = ax^2 + bx + c, \tag{7}$$

and we can also define a value of the function at offset *h* by the Taylor's series

$$f\left(x+h\right) = f\left(x\right) + hf'\left(x\right) + \frac{h^2}{2!} f''\left(x\right) + \frac{h^3}{3!} f'''\left(x\right) + \ldots, \tag{8}$$

where $f'(x)$ is the first derivative, $f''(x)$ the second derivative etc. A polynomial has a limited number of derivatives, and in the case of quadratic equation (7) there are only two derivatives, after that, the higher order derivative are all zero. Choosing a polynomial to represent a curve or portion of a curve becomes convenient as we can limit the number of derivatives.

The Taylor series is rearanged to get derivative $f'(x)$, i.e.,

$$f'(x) = \frac{f(x+h) - f(x)}{h} - \frac{1}{h}\left[\frac{h^2}{2!} f''(x) + \frac{h^3}{3!} f'''(x) + \ldots\right]. \qquad (9)$$

We typically assume a locally smooth curve where the higher order derivatives are zero leaving us with the approximation

$$f'(x) = \frac{f(x+h) - f(x)}{h}, \qquad (10)$$

which is similar to the forward approximation given in equation (5).

We can also consider the Taylor series with –*h*, i.e.,

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2!} f''(x) - \frac{h^3}{3!} f'''(x) + \ldots, \qquad (11)$$

and we could manipulate this equation to define the backward difference equation (6) .

We now subtract equation (11) from (8) to get

$$f'(x) = \frac{-f(x-h) + f(x+h)}{2h} - \frac{1}{2h}\left[\frac{h^3}{3!} f'''(x) + \frac{h^5}{5!} f^5(x) + ..\right] \qquad (12)$$

which only contains the odd order of the higher derivatives. If these higher order derivatives are zero, then we get the approximation

$$f'(x) = \frac{-f(x-h) + f(x+h)}{2h}, \qquad (13)$$

which is the central difference approximation (-1, 0, 1). Texts on this subject will include the higher derivatives as error terms, but I will not do that as I will be looking at the error of the transform of the operator.

The size of a derivative operator is related to a polynomial that will fit the data. If we have a straight line, then any (-1, 1) operator will give an exact answer. If we have a parabola, then any three points will define the curve and we can get the exact derivative.

Consider again a parabola or second order polynomial,

$$f(x) = ax^2 + bx + c, \qquad (14)$$

then the Taylor's series becomes

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2!}2a \quad , \tag{15}$$

where 2*a* is the second derivative of $f(x)$. The derivative is then

$$f'(x) = \frac{f(x+h) - f(x) - h^2 a}{h}, \tag{16}$$

where *a* can be defined from three points that include $f(x+h)$, $f(x)$ and $f(x+h)$. These three points are substituted into the parabola to get

$$f(x-h) = a(x-h)^2 + b(x-h) + c \tag{17}$$

$$f(x) = a(x)^2 + b(x) + c \tag{18}$$

$$f(x+h) = a(x+h)^2 + b(x+h) + c \tag{19}$$

and we can solve for *a*, *b*, and *c*, however we only need to solve for *a*. Subtracting (17) from (18) we get

$$f(x) - f(x-h) = a\left(2xh - h^2\right) + bh, \tag{20}$$

and (18) from (19) we get

$$f(x+h) - f(x) = a\left(2xh + h^2\right) + bh. \tag{21}$$

Subtracting (20) from (21) we get

$$f(x-h) - 2f(x) + f(x+h) = a2h^2, \tag{22}$$

giving the single constant *a* which is

$$a = \frac{f(x-h) - 2f(x) + f(x+h)}{2h^2}, \tag{23}$$

that is almost the second derivative of $f(x)$. Inserting this value of *a* into (15) we get

$$f(x+h) = f(x) + hf'(x) + h^2 \frac{f(x-h) - 2f(x) + f(x+h)}{2h^2}, \tag{24}$$

and solving for the derivative we finally get

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h}, \tag{25}$$

the same central difference equation above. This is an exact solution when our data is fitted as to quadratic equation. This derivation was for the central difference of three

points, assuming a parabolic curve. This solution is valid when we are in the middle of the data but not at the end of the data. We will now consider the evaluation of the derivative at an end (boundary) of the data.

Forward (or left sided) difference estimation

We now evaluate the derivative at the beginning of an array or at a boundary. We start with three similar definition as above but now assume $f(x)$, $f(x+h)$, and $f(x+2h)$, and we desire the derivative $f'(x)$ at *x*, for example at the beginning of the array. Now, using

$$f(x) = a(x)^2 + b(x) + c, \tag{26}$$

$$f(x+h) = a(x+h)^2 + b(x+h) + c, \tag{27}$$

$$f(x+2h) = a(x+2h)^2 + b(x+2h) + c. \tag{28}$$

As above we subtract equation (26) from (27) to get

$$f(x+h) - f(x) = a\left(2xh + h^2\right) + bh \tag{29}$$

and (27) from (28) we get

$$f(x+2h) - f(x+h) = a\left(2xh + 3h^2\right) + bh. \tag{30}$$

Subtracting (29) from (30) for

$$f(x) - 2f(x+2) + f(x+2h) = a2h^2 \tag{31}$$

we get the definition for the constant *a* as

$$a = \frac{f(x) - 2f(x+h) + f(x+2h)}{2h^2}. \tag{32}$$

Inserting this value of a into (15) we get

$$f(x+h) = f(x) + hf'(x) + h^2 \frac{f(x) - 2f(x+h) + f(x+2h)}{2h^2}, \tag{33}$$

and solving for the left sided derivative we get

$$f'(x)_{\text{Left}} = \frac{-3f(x) + 4f(x+h) - f(x-h)}{2h}. \tag{34}$$

This gives an excellent estimate for the first value of the derivative in a sequence. We can also derive a similar boundary valued equation for the final value of an array or right sided derivative.

Do remember the assumptions behind this estimate of a very smooth function, i.e. a quadratic type function.

We can use this method for data that is not evenly sampled, and can estimate the derivative at any location. We estimate the coefficients of the polynomial that defines the curve, and then get the derivative at any reasonable point.

In summary, we have used a quadratic equation to get numerous finite difference solutions to the derivative. We can use higher order polynomials and will get more accurate approximations with higher number of samples.

Higher order estimates of the first derivative.

We can now assume a higher order for the polynomial such as

$$f(x) = ax^n + bx^{n-1} + cx^{n-2} + ... + q\,. \tag{35}$$

This equation can be differentiated n times to get the a component. The corresponding Taylor series will also be of order n,

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2!} f''(x) + \frac{h^3}{3!} f'''(x) + ... + \frac{h^n}{n!} f^n(x), \tag{36}$$

and we will require n+1 coefficients to define a. Subsequent parameters will also require n+1 coefficients until we get to the derivative. Examples of the central differences for higher order first derivative estimates are:

$$f'(x) = \frac{f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h)}{12h}, \tag{37}$$

and

$$f'(x) = \frac{-f(x-3h) + 9f(x-2h) - 45f(x-h) + 45f(x+h) - 9f(x+2h) + f(x+3h)}{60h}. \tag{38}$$

One can also arrive at equation (37) by assuming a third order polynomial through the points (x-2h), (x-h), (x), (x+h) and (x+2h). The second derivative in equation (9) is approximated using the approximate derivative (1, -2, 1)/h^2 centered at (x).

$$f''(x) \approx \frac{f(x-h) - 2f(x) + f(x+h)}{h^2}, \tag{39}$$

The third derivative is approximated by the central difference of the second derivatives defined at points (x-h) and (x+h), i.e.,

$$f''(x-h) \approx \frac{f(x-2h) - 2f(x-h) + f(x)}{h^2}, \tag{40}$$

and

$$f''(x+h) \approx \frac{f(x) - 2f(x+h) + f(x+2h)}{h^2}. \tag{41}$$

Using the difference between these derivatives, we approximate the third derivative with

$$f'''(x) = \frac{-f''(x+h) + f''(x+h)}{2h}, \tag{42}$$

$$f'''(x) = \frac{-\dfrac{f(x-2h) - 2f(x-h) + f(x)}{h^2} + \dfrac{f(x) - 2f(x+h) + f(x+2h)}{h^2}}{2h}, \tag{43}$$

$$f'''(x) = \frac{-f(x-2h) + 2f(x-h) - f(x) + f(x) - 2f(x+h) + f(x+2h)}{2h^3}, \tag{44}$$

$$f'''(x) = \frac{-f(x-2h) + 2f(x-h) - 2f(x+h) + f(x+2h)}{2h^3}. \tag{45}$$

Limiting equation (36) to the third order

$$f'(x) \approx \frac{f(x+h) - f(x)}{h} - \frac{1}{h}\left[\frac{h^2}{2!} f''(x) + \frac{h^3}{3!} f'''(x)\right]. \tag{46}$$

and substituting in the second derivative using equation (39) and third derivatives from equation (45) we get

$$f'(x) \approx \frac{f(x+h) - f(x)}{h} - \frac{1}{h}\left[\frac{h^2}{2!}\frac{f(x-h) - 2f(x) + f(x+h)}{h^2} + \frac{h^3}{3!}\frac{-f(x-2h) + 2f(x-h) - 2f(x+h) + f(x+2h)}{2h^3}\right], \tag{47}$$

$$f'(x) \approx \frac{f(x+h) - f(x)}{h} - \frac{f(x-h) - 2f(x) + f(x+h)}{2h} - \frac{-f(x-2h) + 2f(x-h) - 2f(x+h) + f(x+2h)}{12h}, \tag{48}$$

$$f'(x) \approx \frac{12f(x+h) - 12f(x) - 6f(x-h) + 12f(x) - 6f(x+h) + f(x-2h) - 2f(x-h) + 2f(x+h) - f(x+2h)}{12h}, \tag{49}$$

and finally

$$f'(x) \approx \frac{+f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h)}{12h}. \tag{50}$$

This equation is identical to that in equation (37) which was derived using an exact method for a third order function.

SPECTRAL COMPARISONS OF THE VARIOUS OPERATORS

The problem now becomes one of determining which operator is most suitable for a given application, and how much accuracy is really needed. For most of my career, I have used the two point operator (-1, 1) and it seems to have been quite reasonable for

seismic applications. I will now compare the results of the rather ad hoc spectrally derived operators with the polynomial derived operators by looking at the amplitude spectrums. Note that all the operators have an odd (numerically) shape that ensures a ninety degree phase shift, i.e.,

$$f(t) = -f(-t) \;\Leftrightarrow\; 90^{o}, \tag{51}$$

and that the spectrum contains only imaginary values. Consequently I will compare only the imaginary part the spectrum. The following figure is for the 11 point operator described above in Figure 9. In Figure 11, the tapered spectrum is in blue and the Fourier transform of the 11 point spectrally derived operator is in Green. I again used N_{fft} = 512 for an approximate reference to seismic frequencies for the first half of the spectrum.

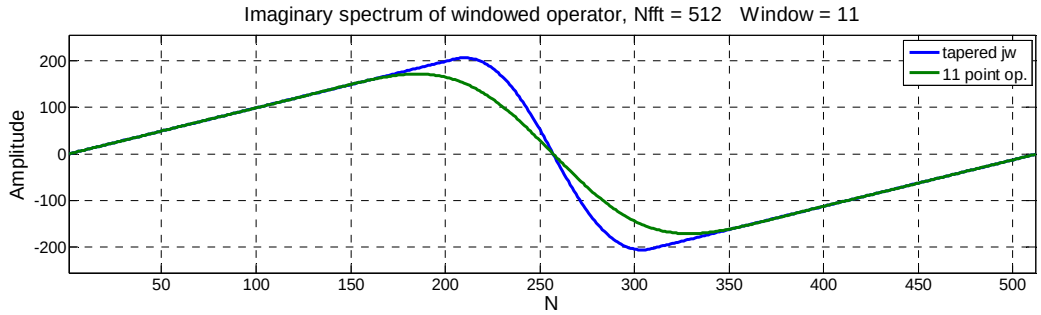


FIG. 11. Comparison of the 11 point widow spectrum with the input tapered spectrum, N_{fft} = 512, taper starting at 80% of the Nyquist frequency.

A visual inspection of the result indicates that this operator is quite good to frequencies up to 175 Hz, well above those found in reflection seismology. The numerical error can be illustrated with a percent accuracy defined as the difference of the above two curves divided by the input tapered value as displayed in Figure 12. This figure shows the range of the percentage error from +– 5%. The error in this operator is less than 0.25% for frequencies up to approximately 160 hz. This is a very small error where typical errors that range of 1% to 5% may be acceptable.

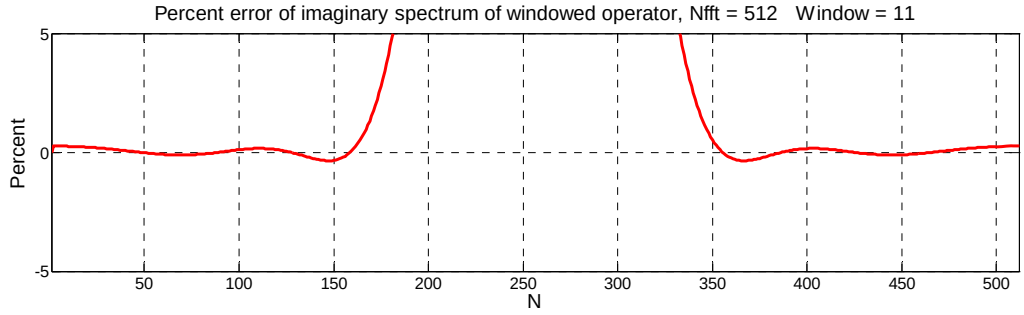


FIG. 12. Comparison of the 11 point widow spectrum with the input tapered spectrum, N_{fft} = 512, taper starting at 80% of the Nyquist frequency.

Comparison of spectral and polynomial operators

The following 3, 5, 7, and 9 point operators were derived using the spectral estimation method and plotted in Figure 13. The zoomed values show some oscillation about the theoretical value as indicated by the error of the 11 point operator in Figure 10. The operator values are:

```
Opp1= [0.4826   0   -0.48264 ];
Opp2 =[ -0.1083   0.7238   0   -0.7238   0.1083 ]
Opp3 =[ 0.0351   -0.2165   0.8238   0   -0.8238   0.2165   -0.0351]
Opp4 =[-0.0130   0.0827   -0.2834   0.87300   0   -0.8730   0.2834 -0.0827   0.0130]
```

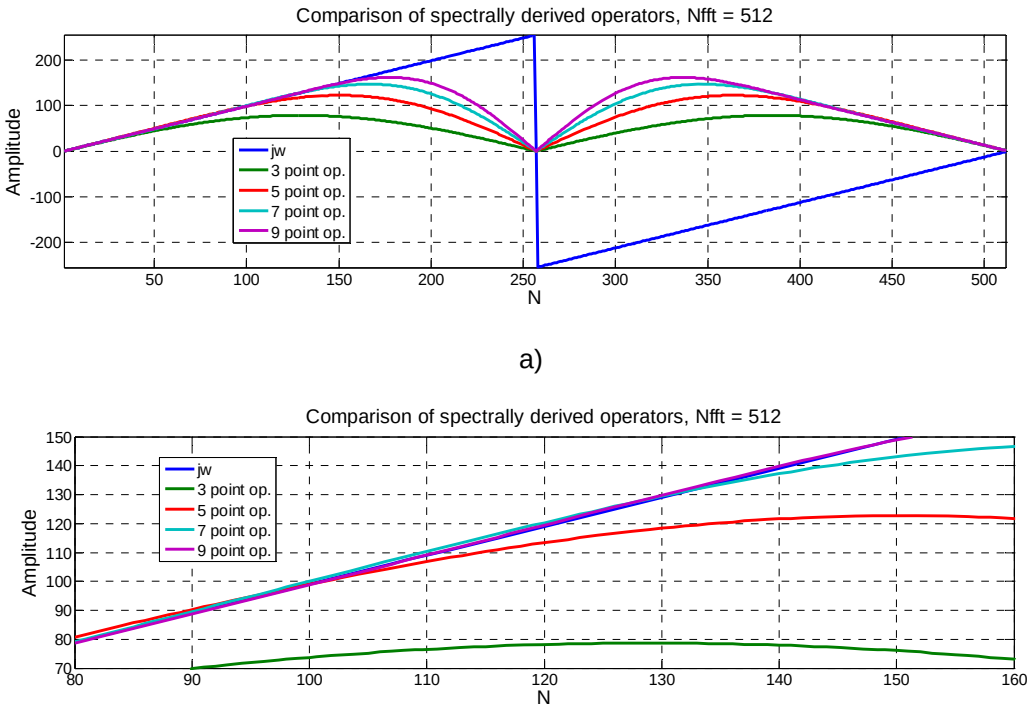


FIG. 13. Comparison a) of the amplitude spectrums for spectrally derived operators for operator sizes 3, 5, 7, and 9 points and b) a zoomed display.

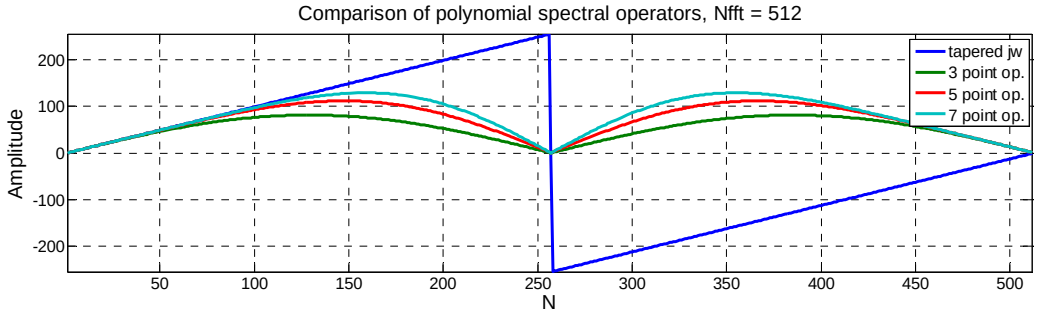
Operators derived from the polynomial method for sizes 3, 5, and 7 are shown below. The spectrum of these opserators are smooth and tangent to the theoretical values. They are:

```
Opp5 = [ -1   0   1 ]/2
Opp6 = [ 1   -8   0   8   -1 ]/12
Opp7 = [ -1   9   -45   0   45   -9   1 ]/60
```

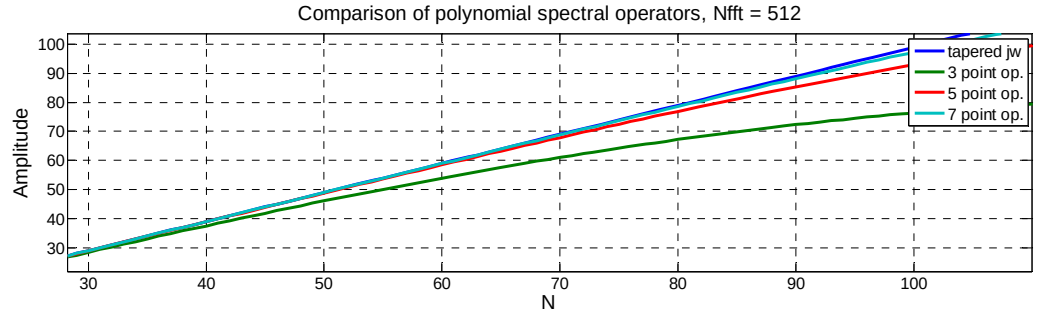
and plotted in Figure 14.

The spectrally derived operators appear superior to the polynomial derived operators. A comparison between the two different operators for size 7 are shown in Figure 13, where it is evident that the spectrally derived operator has a superior bandwidth.

The errors of the seven point operators are shown in Figure 14. The spectral operator has a larger bandwidth with an error of 1% that extend to approximately 150 Hz while that of the polynomial extends about 100 Hz with a much smaller error at the lower frequencies.

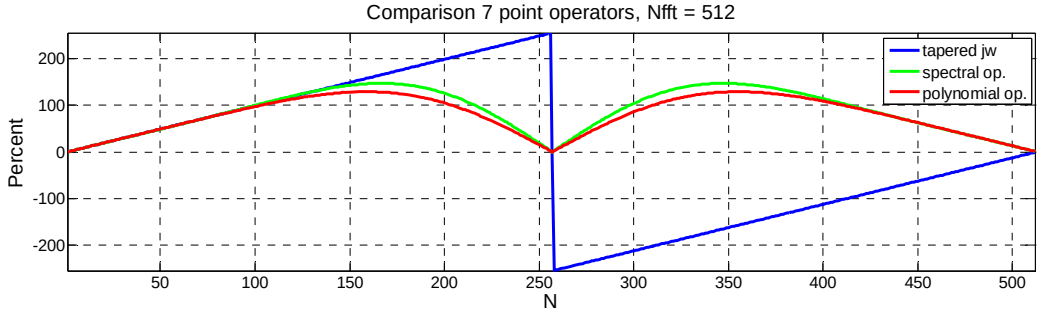


a)

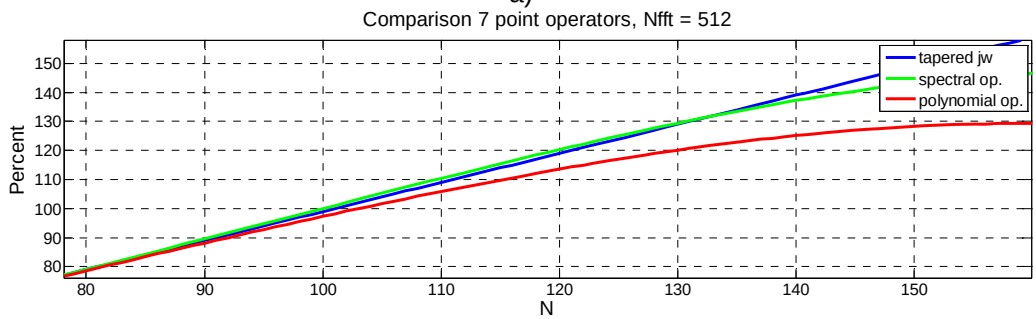


b)

FIG. 14. Comparison of a) the amplitude spectrums for polynomial derived operators for operator sizes 3, 5, and 7 points and b) a zoomed display.



a)



b)

FIG. 15. Comparison between a) the spectral and polynomial derived operators for size 7 points, and b) a zoomed display.

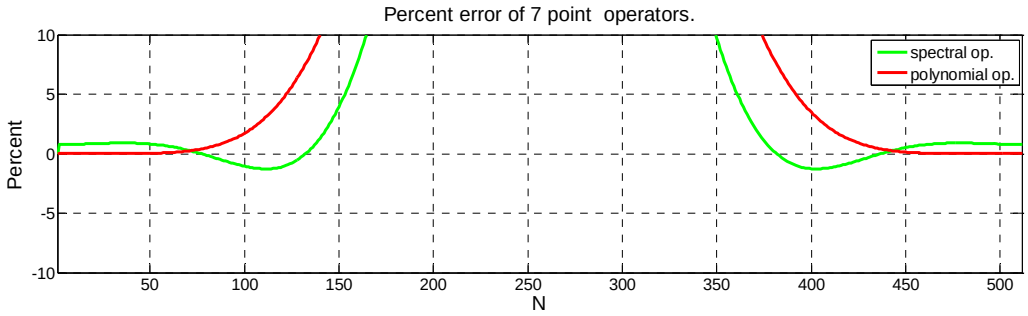


FIG. 16. Error comparison of the two 7 point operators.

REMEZ ALGORITHM

MATLAB has digital filter design software in the Signal Processing Package. This package uses the Remez exchange algorithm that was designed in the 1970's to design bandpass filters and differentiators. An example of the results for designing a differentiator are shown below for a 7 point filter. The *N* used in this filter does not count the zero value at t = 0. The code and resulting filter are:

```
>> h=firpm(6,[0 .3 .45 1],[0 1 1 0],'differentiator');h,plot(h);

h = -0.0636  0.1271  0.4769  0  -0.4769  -0.1271  0.0636
```

The following figure compares this result with the polynomial and the spectrally derived results.

I was not able to improve on the results of the Remez design and assume that is because we are using small operators. It appears that the Remez design must be limited to frequencies well below half the Nyquist frequencies or below a quarter of the sampling frequencies. A better design with these small operator sizes may be possible.

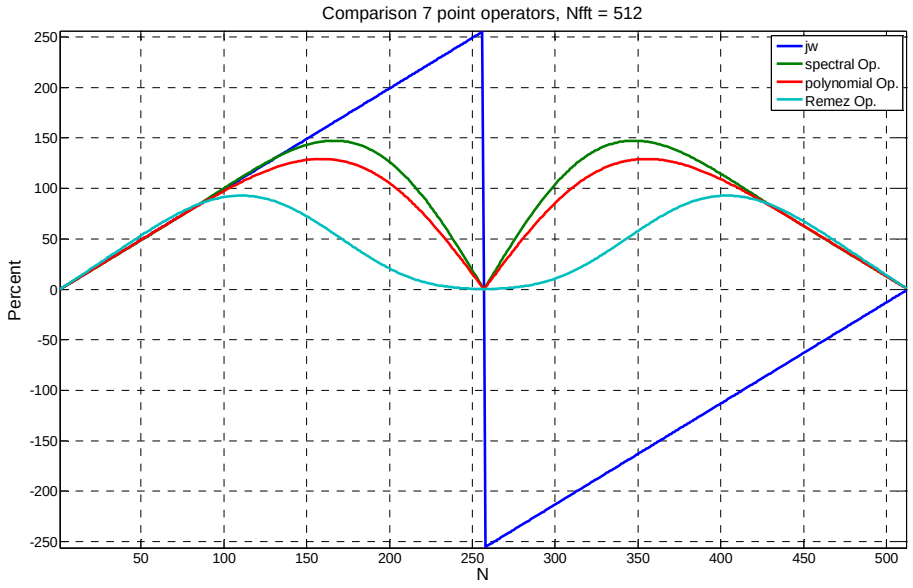


FIG. 17. Comparison of the amplitude spectrum for a spectrally derived operator, ta polynomial approximation operator, and the Remez operator, all 7 point in length.

COMPARISON OF (-1, 1) AND (-1, 0, 1)

The (-1, 1) and (-1, 0, 1) operators are popular with many processes. The (-1, 1) operator does have a higher bandwidth but has a phase shift that is 90^o at zero frequency, but then has a taper that is linear with frequency due to the half cycle shift. The (-1, 0, 1) operator does have the desired 90^o phase shift but does assume a smoother trace with a lower bandwidth. The comparison of the spectra are shown in Figure 17.

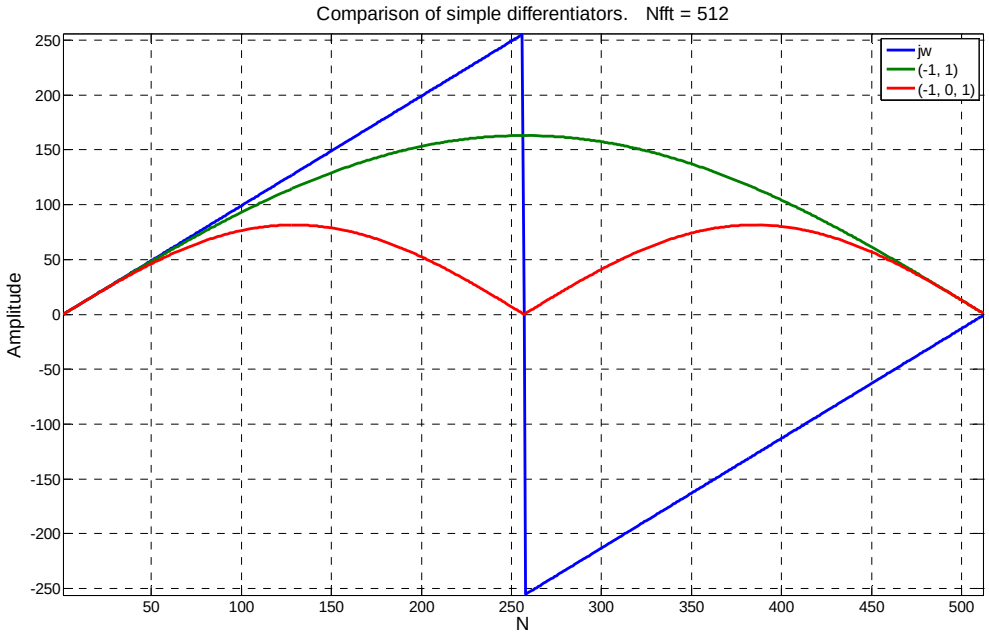


FIG. 18. Comparison of (-1,1) and (-1,0 -1) amplitude spectrums.

TWEAKING THE OPERATORS

The percent error of these two point operators is always positive as illustrate in Figure 19. The operators have a 0 to 5% error at bandwidths of 90 and 45 Hz respectively. We can scale the amplitude of the operator by a variable *x*, i.e. *x**(-1, 1), to introduce a negative error that will increase the bandwidth of the range of the percent error. We can slightly change the amplitude scaling of the operators using 1.05*(…) for a percent error that ranges from -5 to +5% to increase the bandwidths to 125 and 62.5 Hz, as illustrated below in Figure 20. These scaling factors don't affect the quality of the migration, and should only be used to establish a positive and negative range of the percent error.

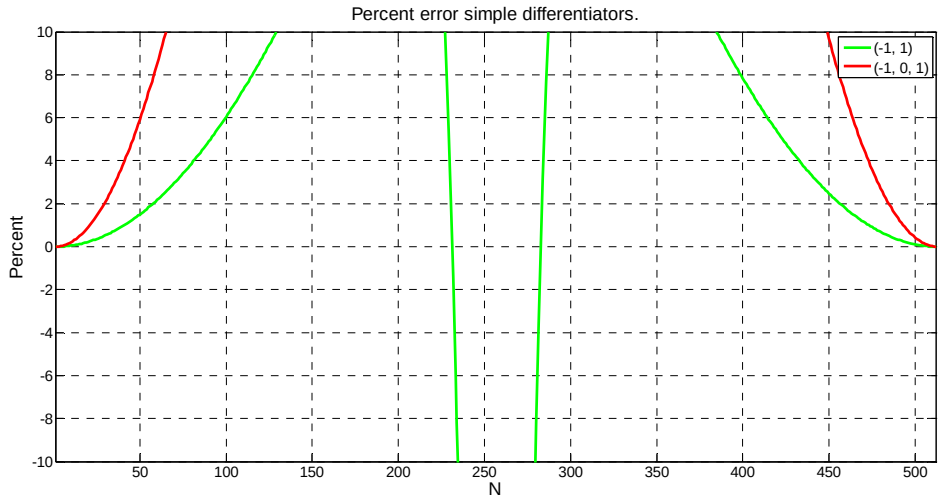


FIG. 19. Percent error of amplitude spectrum for (-1,1) and (-1,0 -1) operators.

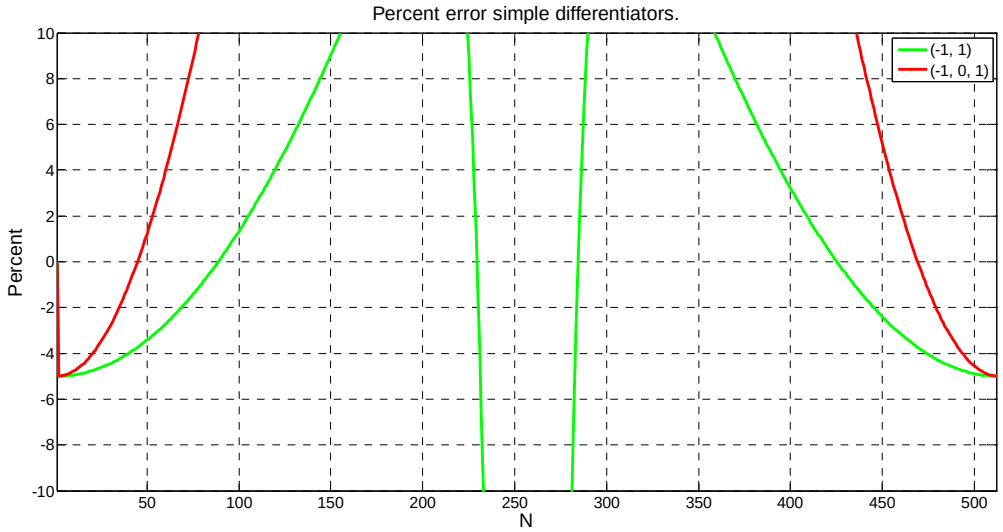


FIG. 20. Percent error of amplitude spectrum for (-1,1) and (-1,0 -1) operators that are weighted by 1.05.

Two point operator (-1, 1)

Has a 90^o phase shift at $f = 0$, with a linear phase shift of a half cycle lag.

Greater bandwidth. BW(-1, 1) > BW(-1, 0, 1) as illustrated above.

The 1% bandwidth for 1.00x(-1, 1) is 0 – 40 Hz.

The 1% bandwidth for 1.01x(-1, 1) is 0 – 55 Hz.

The 5% bandwidth for 1.05x(-1, 1) is 0 – 125 Hz.

The 5% bandwidth for 1.10x(-1, 1) is 0 – 175 Hz.

This is the better operator if the half cycle shift is not a problem,.

Three point operator (-1, 0, 1)

Exact 90^o phase shift

1% bandwidth 0 – 20Hz 1.00x(-1, 0, 1)

1% bandwidth 0 – 30Hz 1.01x(-1, 0, 1)

5% bandwidth 0 – 60Hz 1.05x(-1, 0, 1)

10% Bandwidth 0 – 85Hz 1.10x(-1, 0, 1)

This is a better operator if the half cycle shift is a problem but does require band-limited data that we normally have in a seismic trace. It will be a problem when applied to spatial data that is sampled close to the Nyquist criterion.

RECURSIVE DIFFERENTIAL OPERATOR

Z transform

The Z transform is a digital form of the continuous Laplace transform. It is more general than the Fourier transform as the exponent of the exponential "e^z" is complex, rather than just imaginary $e^{j\omega}$ as in Fourier transform. This is all I want to say about that.

A simple example of its use is in convolution, where z^n represents a delay of n samples.

Consider the samples of an input trace (a_0, a_1, a_2, a_4, a_5, …) and a filter (h_{-1}, h_0, h_1) where the subscripts define the sample number and subscript zero corresponds to time zero. We can express these arrays as the Z transforms ($a_0 + a_1z^1 + a_2z^2 + a_3z^3 + \ldots$) and ($h_{-1}z^{-1} + h_0 + h_1z^1$). Note the three point filter is centered at time zero, and that the negative time sample is represented with a negative exponent. Convolution is accomplished by multiplying the two Z transformed arrays and gathering the corresponding coefficients of z, i.e.,

$$\begin{aligned}&\left(a_0 + a_1z^1 + a_2z^2 + a_3z^3 + \ldots\right)\times\left(h_{-1}z^{-1} + h_0 + h_1z^1\right)\\ &= a_0h_{-1}z^{-1} + \left(a_0h_0 + a_1h_{-1}\right) + \left(a_0h_1 + a_1h_0\right)z^1 + \left(a_1h_1 + a_2h_0 + a_3h_{-1}\right)z^2 + \left(a_2h_1 + a_3h_0 + a_4h_{-1}\right)z^3 + \ldots\\ &= b_{-1}z^{-1} + b_0 + b_1z^1 + b_2z^2 + b_3z^3 + \ldots\end{aligned} \tag{52}$$

Consider a continuous trace $x(t)$, with samples x_n defines at discrete intervals of $x(t)$. In continuous time, the Fourier transform of $x(t)$ is

$$F\left[x(t)\right] = X(\omega), \tag{53}$$

and the Fourier transform of the derivative product of $j\omega$ the transformed function, i.e.,

$$F\left[\frac{dx(t)}{dt}\right] = j\omega X(\omega), \tag{54}$$

The Z transform is of x_n is defined as

$$Z[x_n] = X(z), \tag{55}$$

where

$$Z[x_n] = x_0 + x_1 z^1 + x_2 z^2 + x_3 z^3 + x_4 z^4 \ldots . \tag{56}$$

Note that the unit sample display is simply the product of "*z*". The Z transform of the derivative is the product of z^n with the Z transformed function, i.e.,

$$Z\left[\frac{dx_n}{dt}\right] = z^n X(z). \tag{57}$$

Recursive operator

A recursive differential operator may be derived at $n + 1/2$ by taking an average of the derivatives at n and n+1, and equating that to a single backward difference at n, i.e.,

$$\frac{dx_{n+1/2}}{dt} \approx \frac{1}{2}\left(\frac{dx_n}{dt} + \frac{dx_{n+1}}{dt}\right) \approx \frac{-x_n + x_{n+1}}{\delta t}. \tag{58}$$

We can also approximate the derivative at n+1 using an average of two backward differences, i.e.,

$$\frac{dx_{n+1}}{dt} \approx \frac{1}{2}\left(\frac{dx_n}{dt} + \frac{dx_{n+1}}{dt}\right) \approx \frac{1}{2}\left(\frac{-x_n + x_{n+1}}{\delta t} + \frac{-x_{n+1} + x_{n+2}}{\delta t}\right) \approx \frac{-x_n + x_{n+2}}{2\delta t}. \tag{59}$$

I will use the first approximation of equation (58) but either will be OK. This equation can be manipulated using the "*Z*" transform to define a new operator. Assuming the sample interval to be unity, we take the Z transform of the left derivative at times, n and n+1, i.e.,

$$Z\left[\frac{1}{2}\left(\frac{dx_n}{dt} + \frac{dx_{n+1}}{dt}\right)\right] \approx X(z)\frac{\left(z^n + z^{n+1}\right)}{2} = z^n X(z)\frac{(1+z)}{2}, \tag{60}$$

and on the right the Z transform becomes

$$Z[-x_n + x_{n+1}] = X(z)(-1+z). \tag{61}$$

Substituting back into equation (58) we get

$$z^n X(z)\frac{(1+z)}{2} \approx X(z)\left(-1+z^2\right), \tag{62}$$

or

$$z^n X(z) \approx \frac{2(-1+z)}{(1+z)} X(z), \tag{63}$$

where we now approximate the derivative product z^n with the rational function

$$h(z) = \frac{2(-1+z)}{(1+z)} \tag{64}$$

We can now do three things with this equation:

1. get the impulse response
2. use the *z* arrays to get the recursive solution in the time domain, or
3. solve the function $h(z)$ where $|z| = 1$, to get the frequency response.

The expression $|z| = 1$ describes a circle on the *z* plane, with magnitude one and with an angle that represents normalized frequencies that range from zero to π.

Impulse response of the recursive differential filter

The impulse response of the "filter" $h(z)$ is found by division to get

$$h(z) = 2(z-1)\left(1-z+z^2-z^3+z^4-+...\right), \tag{65}$$

or

$$h(z) = 2-4z+4z^2-4z^3+4z^4-+.... \tag{66}$$

This is a causal filter (2, -4, 4, -4, 4, …) that will take the derivative of an array using convolution. It does look a little suspicious with the alternating amplitudes that do extend to infinity. However, it is legitimate especially when the alternating amplitudes represent a large amplitude at the Nyquist frequency. There should be zero amplitude at the Nyquist frequency, and the oscillations should not be present in when differentiating a band limited signal. Note, convolving with a two point filter (0.5, 0.5) will completely remove the oscillations.

Recursive solution to an input impulse

The recursive solution can be found by writing the equation in the form

$$h(z)(1+z) = 2(z-1), \tag{67}$$

or, when defining the differentiated array y_n,

$$\frac{Y(z)}{X(z)}(1+z) = 2(z-1), \tag{68}$$

or

$$zY(z) = X(z)2(z-1) - Y(z).$$

Inverse transforming from the Z domain to the time gives

$$y_{n+1} = 2(x_{n+1} - x_n) - y_n. \tag{69}$$

This is now in a recursive form where the previous output y_n is now used as part of the solution for y_{n+1}.

We can look at the impulse response of this filter by assume an input delta function x = ($1_{(n=1)}$, 0, 0, 0, …), and prior samples are zero. (Note that the impulse response contains energy all frequencies, including the Nyquist frequency.) When n = 1, we can compute the response:

$y_1 = 2\times(1-0) - 0 = 2$

$y_2 = 2\times(0-1) - 2 = -4$

$y_3 = 2\times(0-0) - 4 = +4$

and we get the same as the impulse response (2, -4, 4, -4, 4, …).as in equation (66). I suggest you try the input (1, 1, 0, 0, 0, …) which will tend to cancel the oscillations.

The response of a *recursive* filter has an infinite impulse response (IIR). IIR filters may be more compact, accurate, and faster than conventional convolutional filters, however they are better suited for causal responses. The oscillations may become unstable when used with numerical approximations. A damping factor γ that is close to unity $\gamma = 0.9$ may be used to improve the stability of the recursive filter, i.e.,

$$\boxed{y_{n+1} = 2(x_{n+1} - x_n) - \gamma y_n}. \tag{70}$$

This recursive filter is very short and fast to compute and produces a reasonable response for band limited data.

Spectral response

We get the frequency response $h(\omega)$ by substituting $z = e^{j\omega}$ into equation (64) giving

$$h(\omega) = \frac{2(e^{j\omega} - 1)}{(1 + e^{j\omega})}. \tag{71}$$

We are actually defining $z = e^{j\omega} = \cos(\omega) + j\sin(\omega)$, for values $0 < \omega < \pi$ which lie on the unit circle in the Z plane. The results are plotted in Figure 21 for N = 256 that contains the desired value $j\omega$ in blue, the real component that is zero in green, the

imaginary component in red, the magnitude in cyan, and the phase in purple. The amplitudes in this figure are now normalized because the time interval was set to unity. A zoom is shown in Figure 22 where the phase can be identified as approximately -1.5, or exactly at $-\pi/2$. The fit is good to approximately 100 hz, similar to the (-1, 1) operator that was used to define the recursive operator in equation (58). The time domain response could be improved by convolving with an operator, defined in the frequency domain similar to that used in spectral approximations.

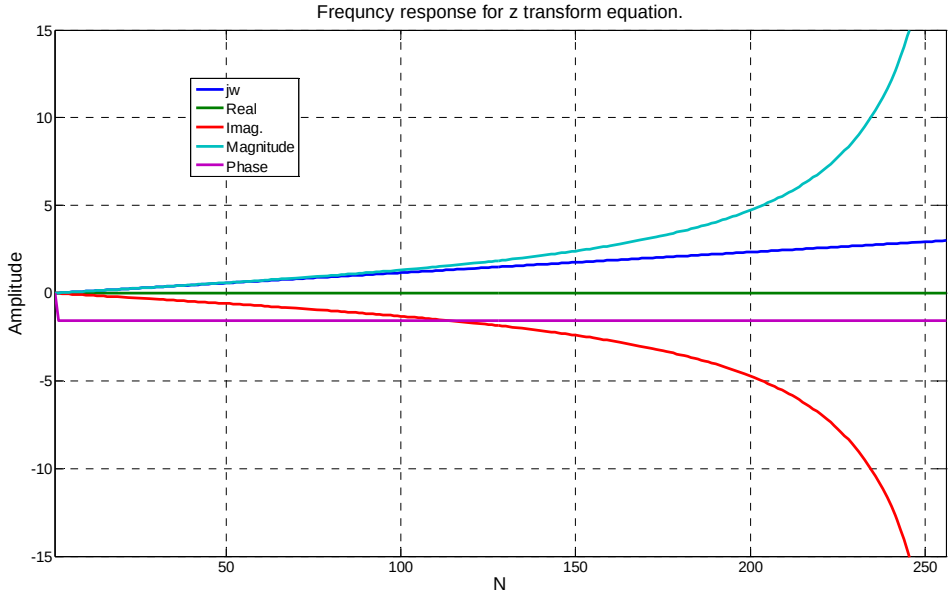


FIG. 21. The frequency response for a recursive filter.

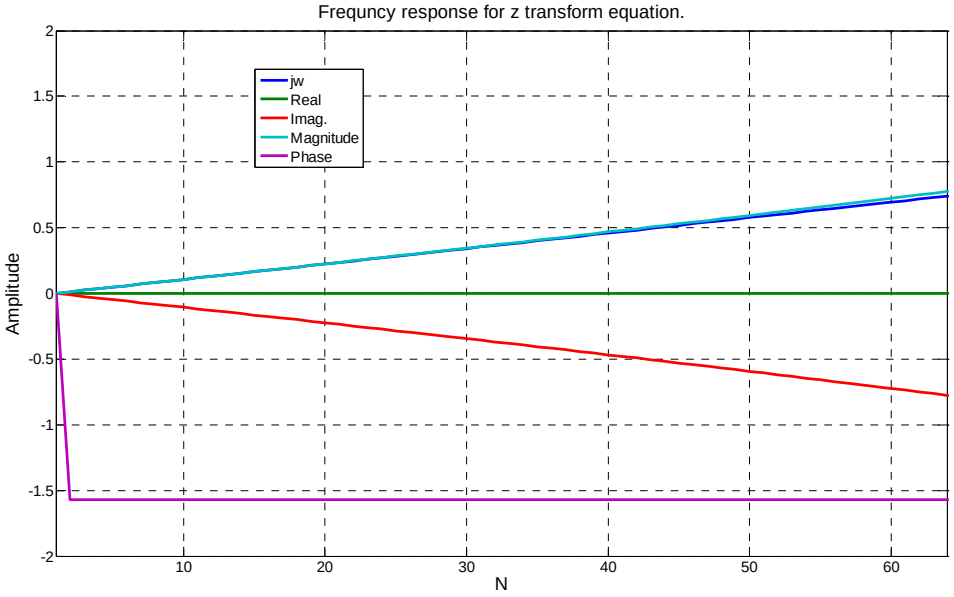


FIG. 22. A zoom of the recursive filter.

COMMENTS

The following figure contains numerous approximations to the first derivative.

Formula	Error term
$f'(x_i) \approx \frac{f(x_{i-2})-4f(x_{i-1})+3f(x_i)}{2h}$	$\frac{1}{3}h^2 f^{(3)}$
$f'(x_i) \approx \frac{f(x_{i+1})-f(x_{i-1})}{2h}$	$\frac{1}{6}h^2 f^{(3)}$
$f'(x_i) \approx \frac{-3f(x_i)+4f(x_{i+1})-f(x_{i+2})}{2h}$	$\frac{1}{3}h^2 f^{(3)}$
$f'(x_i) \approx \frac{3f(x_{i-4})-16f(x_{i-3})+36f(x_{i-2})-48f(x_{i-1})+25f(x_i)}{12h}$	$\frac{1}{5}h^4 f^{(5)}$
$f'(x_i) \approx \frac{-f(x_{i-3})+6f(x_{i-2})-18f(x_{i-1})+10f(x_i)+3f(x_{i+1})}{12h}$	$\frac{1}{20}h^4 f^{(5)}$
$f'(x_i) \approx \frac{f(x_{i-2})-8f(x_{i-1})+8f(x_{i+1})-f(x_{i+2})}{12h}$	$\frac{1}{30}h^4 f^{(5)}$
$f'(x_i) \approx \frac{-3f(x_{i-1})-10f(x_i)+18f(x_{i+1})-6f(x_{i+2})+f(x_{i+3})}{12h}$	$\frac{1}{20}h^4 f^{(5)}$
$f'(x_i) \approx \frac{-25f(x_i)+48f(x_{i+1})-36f(x_{i+2})+16f(x_{i+3})-3f(x_{i+4})}{12h}$	$\frac{1}{5}h^4 f^{(5)}$
$f'(x_i) \approx \frac{10f(x_{i-6})-72f(x_{i-5})+225f(x_{i-4})-400f(x_{i-3})+450f(x_{i-2})-360f(x_{i-1})+147f(x_i)}{60h}$	$\frac{1}{7}h^6 f^{(7)}$
$f'(x_i) \approx \frac{-2f(x_{i-5})+15f(x_{i-4})-50f(x_{i-3})+100f(x_{i-2})-150f(x_{i-1})+77f(x_i)+10f(x_{i+1})}{60h}$	$\frac{1}{42}h^6 f^{(7)}$
$f'(x_i) \approx \frac{f(x_{i-4})-8f(x_{i-3})+30f(x_{i-2})-80f(x_{i-1})+35f(x_i)+24f(x_{i+1})-2f(x_{i+2})}{60h}$	$\frac{1}{105}h^6 f^{(7)}$
$f'(x_i) \approx \frac{-f(x_{i-3})+9f(x_{i-2})-45f(x_{i-1})+45f(x_{i+1})-9f(x_{i+2})+f(x_{i+3})}{60h}$	$\frac{1}{140}h^6 f^{(7)}$
$f'(x_i) \approx \frac{2f(x_{i-2})-24f(x_{i-1})-35f(x_i)+80f(x_{i+1})-30f(x_{i+2})+8f(x_{i+3})-f(x_{i+4})}{60h}$	$\frac{1}{105}h^6 f^{(7)}$
$f'(x_i) \approx \frac{-10f(x_{i-1})-77f(x_i)+150f(x_{i+1})-100f(x_{i+2})+50f(x_{i+3})-15f(x_{i+4})+2f(x_{i+5})}{60h}$	$\frac{1}{42}h^6 f^{(7)}$
$f'(x_i) \approx \frac{-147f(x_i)+360f(x_{i+1})-450f(x_{i+2})+400f(x_{i+3})-225f(x_{i+4})+72f(x_{i+5})-10f(x_{i+6})}{60h}$	$\frac{1}{7}h^6 f^{(7)}$

FIG. 23. Differential operators for three, five, and seven points.

The figure was taken from the internet with the following address:

`http://documents.wolfram.com/mathematica/Built-inFunctions/AdvancedDocumentation/DifferentialEquations/NDSolve/PartialDifferentialEquations/TheNumericalMethodOfLines/SpatialDerivativeApproximations.html`

Consider the seven different seven point approximation. Each estimate the derivative at different locations relative to the seven input points. This is very useful when using linear algebra to perform the derivative of an array of data. The derivative can be estimated at the first sample using a seven point operator. The second point can also be estimated with a different operator and similarly for the third point. The fourth point will use the central equation that will be identical for all point up to near the end where other equations can compute the final three point.

A cartoon of five point operators in matrix form is shown in Figure 24. Note the roll in of the operators, the repeated central difference operators and the final roll out operators.

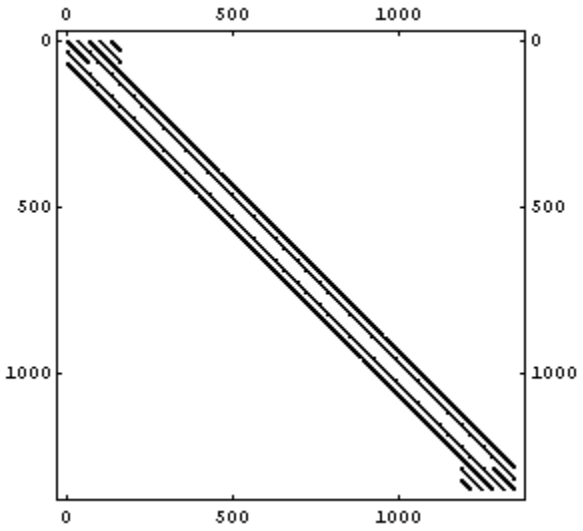


FIG. 24. Differential operators in matrix form illustrating roll-in and rollout at the boundaries.

CONCLUSIONS

Numerous short approximations to the derivative were presented. The best solution for a seven point operator was found from the inverse Fourier transform of a windowed $j\omega$ function.

Operators were defined using polynomial approximations and truncated Taylor series. These short operators do assume a low order polynomial and can define the derivative at any location within the locally smoothed area. This aids in defining derivatives at the beginning and end of an array (the boundary conditions). This method is also suitable for data that is not sampled with equal increments.

Spectrally derived operators have a larger bandwidth for an equivalent seven point operator. The low frequency error is larger, however this may not be a problem for seismic data where the energy band of interest may extend from 10 Hz to 100Hz.

ACKNOWLEDGEMENTS

The author wishes to acknowledge the sponsors of CREWES for their continuing support.

REFERENCES

Bancroft, ,J. C., and Geiger, H., 2003, Analysis and design of filters for differentiation, CREWES Research Report, **9**.

APPENDIX

The 3 point operators only differ by a scale factor which may not be too significant, and the performance of either filter, relative to the theoretical values, may be improved with a slight scaling change. Scaling the 3 point spectral operator by 1.17 produces an error of less than 12.5% over the spectral range from 0 to 100 Hz as illustrated in Figure 22.

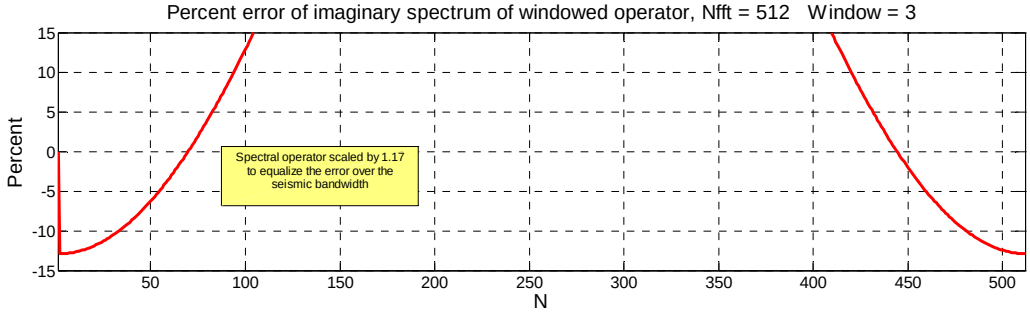
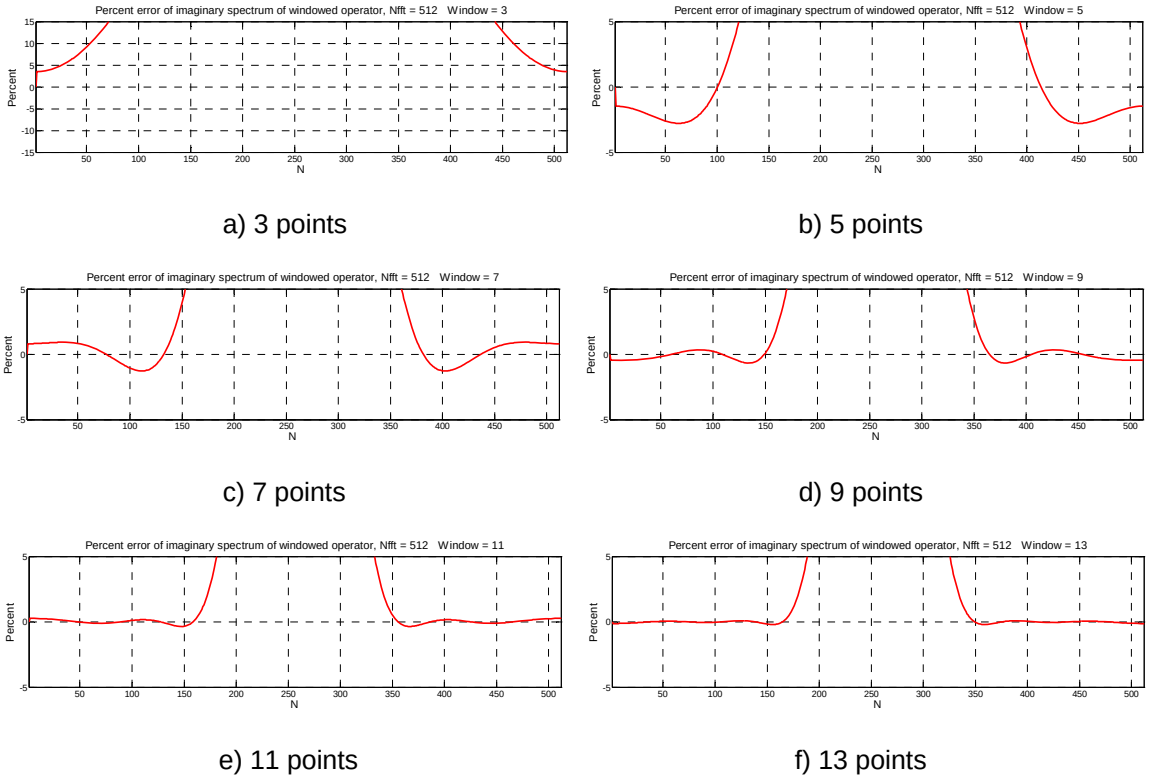


FIG. 25. Three point spectral operator scaled by 1.17 to minimize the error over the seismic bandwidth.

A visual comparison of the percentage errors for spectrally derived operators with sizes 3, 5, 7, 9, 11, and 13 follow.



a) 3 points b) 5 points

c) 7 points d) 9 points

e) 11 points f) 13 points

FIG. 26. A visual comparison of the error of spectrally derived operators for size that range from 3 to 13 points as annotated.