Analysis and design of filters for differentiation

John C. Bancroft and Hugh D. Geiger

SUMMARY

Differential equations are an “integral” part of seismic processing. In the discrete computer world they are solved using finite difference equations. Derivatives are also used in the integral forms of DMO and migration algorithms.

This paper looks at some commonly used convolutional and recursive filters that are used to approximate the derivative. Traditionally, the convolutional methods are the easiest to design, but the recursive methods run faster.

The main motivation is to find fast and accurate filters that represent the first derivative, the second derivative and the square root derivative.

INTRODUCTION

Differential equations are used in a number of geophysical applications. We will briefly describe some common applications, look at the theoretical Fourier domain expressions, and then examine some solutions in detail. Although the filters can be used for both time and spatial applications the discussion will focus on the time solutions.

Wave equation modelling and migration

The standard form of the acoustic wave equation is usually expressed as,

$$\frac{\partial^2 P}{\partial x^2}+\frac{\partial^2 P}{\partial y^2}+\frac{\partial^2 P}{\partial z^2}=\frac{1}{V^2}\frac{\partial^2 P}{\partial t^2}. \qquad (1)$$

The second-derivatives may be approximated by a three point finite difference operator (1, -2, 1). This convolution operator is successfully used in full wave equation solution for modelling and migration.

Finite difference downward continuation

Downward continuation methods of migration use the first derivative to propagate energy from one depth layer at $P(z_n)$ to a deeper layer $P(z_{n+1})$ by,

$$P(z_{n+1})=P(z_n)+\delta z\frac{\partial P(z_{n+1/2})}{\partial z}. \qquad (3)$$

Note, in this case, the derivative is defined at the midpoint between the two layers, i.e. at the point $z_{n+1/2}$.

Integral solutions for DMO

Integral forms of DMO and migration require filters that modify the phase and amplitude spectrum. For both 2-D and 3-D data, the correct filter to apply is a square-root differential filter.

An example of a DMO operator with and without the phase filter is shown in Figure 1. Each part of Figure 1 contains an input trace on the right, a DMO suite of traces in the center, and an output trace on the left. The output trace is formed by spatially summing all traces in the DMO suite. Wavelets in this summed trace represents the DMO'd wavelets of a flat event, and should match the input wavelet. The DMO suite in (a) uses Liner's frequency domain method and may be considered to be correct (Liner 1990). The DMO in (b) and (c) was computed using a time domain method in which the input trace is spread along the appropriate ellipse. Both scaling and antialiasing filters have been applied to (b) and (c) however the phase correcting filter has been omitted in (b) but included in (c).

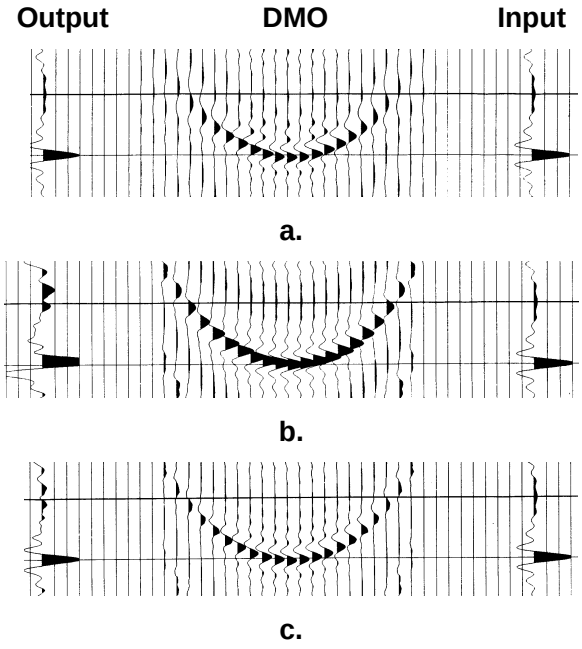


Figure 1. DMO operators showing a) Liners (1999) frequency domain method, b) time domain with no phase filter, and c) time domain with the phase filter.

Note the wavelet in the DMO suite of (b) is zero phase, and the summed trace does not match the input trace. Application of the phase filter in (c) has improved the DMO suite as is evident by the reasonable match between summed trace and the input trace.

Kirchhoff migration

The integral or Kirchhoff solution to the wave equation for 2-D and 3-D data (Gazdag 1979) are written as

$$p_{2\text{-}D}(x_1, z_1, t=0) = \int \frac{\cos\beta}{\sqrt{2\pi\, rc}} \frac{\partial\, p^{1/2}}{\partial\, t}\,(x, z=0, t=r/c)\, dx\,, \tag{4}$$

$$p_{3\text{-}D}(x_1, y_1, z_1, t=0) = \iint \frac{\cos\beta}{2\pi\, rc} \frac{\partial\, p}{\partial\, t}(x, y, z=0, t=r/c)\, dxdy\,. \tag{5}$$

Both integrals contain derivative terms; a square root derivative for the 2-D input data in equation (4) and a derivative of the 3-D input data in equation (5).

Theoretical differential filters

The derivative of an input trace may be accomplished accurately in the frequency domain by applying a ninety-degree phase shift $(\pi/2)$ and multiplying the amplitudes by a ramp function that is proportional to the frequency. Since this filter is the Fourier transform of the derivative, we refer to it as the $j\omega$ (jay-omega) filter where j represents $\sqrt{(-1)}$.

The second-derivative filter may also be applied in the frequency domain by squaring $j\omega$ to get $-\omega^2$. This is accomplished by multiplying the amplitude spectrum by function that is proportional to the frequency squared. The sign change is equivalent to a phase shift of 180 degrees (π). We refer to this filter as the $-\omega^2$ (omega squared) filter.

The square root differential filter is accomplished in the Fourier domain by applying a forty five-degree phase shift $(\pi/4)$, and multiplying the spectrum amplitudes by a function that is proportional to the square root of the frequency. We refer to this as the $rj\omega$ (r jay omega) filter.

These filters can be applied efficiently in the frequency domain along with other operations such as band-pass filters or static shifts when the data is already in the frequency domain. However, the time required to perform Fourier transforms just for differential filters may be excessive and the time domain solutions may run much faster. It is the objective of this paper to design and evaluate appropriate time domain filters.

FIRST DERIVATIVE

The standard form

The method of taking a first derivative is illustrated in Figure 2 and expressed as

$$\frac{dy}{dx}(n) \approx \frac{y(n+1) - y(n)}{\delta x}, \tag{6}$$

where δx is the sample spacing. Note the derivative is defined at *n*, but could also have been defined at *n + 1*. A more correct solution would define the derivative at n + ½. Unfortunately that is usually not possible as the output samples must be aligned with the input sample at *n, n + 1,* etc.

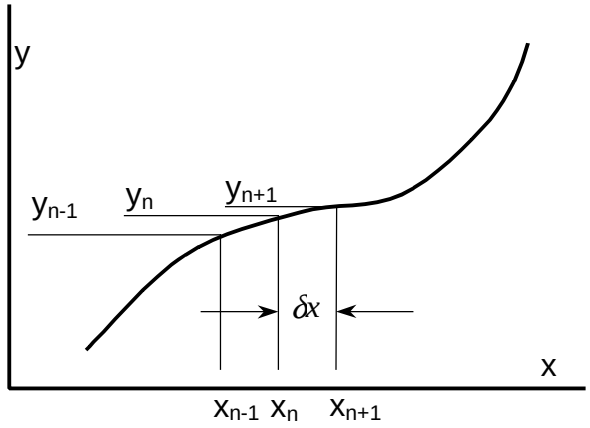


Figure 2. Simple geometric plot defining samples in x and corresponding values for y.

The finite difference approximation to the derivative of equation (6) is used in many areas, but is not satisfactory for geophysical applications. This was recognised as a problem and the following solution proposed and used in finite difference solutions (Brysk 1983).

An improved derivative (???)

A rather "obvious" improvement for estimating the derivative at n + 1/2 may be found by averaging the derivatives at n and n + 1 giving,

$$\frac{dy}{dx}\left(n+\tfrac{1}{2}\right) \approx \frac{1}{2}\left(\frac{dy}{dx}(n) + \frac{dy}{dx}(n+1)\right) \approx \frac{y(n+1) - y(n)}{\delta x}. \tag{7}$$

This equation is realised using the "z" transform where z^n represents a delay of *n* samples. The derivative *p* is expressed as

$$p \approx \frac{\partial}{\partial x} \approx \frac{2(1-z)}{\delta x(1+z)}, \tag{8}$$

where z is used to represent a delay of one sample. Expressed in sample notation, equations (7) and (8) become

$$p_n + p_{n-1} \approx \frac{2(y_n - y_{n-1})}{\delta x} \tag{9}$$

or

$$p_n \approx \frac{2(y_n - y_{n-1})}{\delta x} - p_{n-1}. \tag{10}$$

This form of the equation is written to show that output p_n must have the previous output p_{n-1} added back into the solution. This is a *recursive* filter and typically has an infinite impulse (time) response (IIR). IIR filters may be more compact, accurate, and faster than conventional convolutional filters, however they are better suited for causal responses. Let's check it out.

The impulse response for the convolutional filter defined by equation (6) is

$$p_n = 1, -1, \quad 0, \quad 0, \quad 0, \quad 0, \quad 0, \quad 0, \quad 0, \ldots ,$$

and the recursive IIR filter defined by equation (10) is

$$p_n = 2, -4, +4, -4, +4, -4, +4, -4, +4, \ldots .$$

This certainly does not look like the filter we desire as a large amplitude has been generated at the Nyquist frequency.

Fourier transform of the first derivatives.

We will start by defining the derivative in the Fourier transform domain as illustrated in Figure 3.

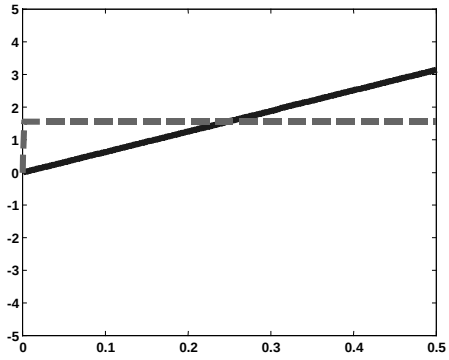


Figure 3. Theoretical magnitude (solid) and phase (dashed) of the derivative.

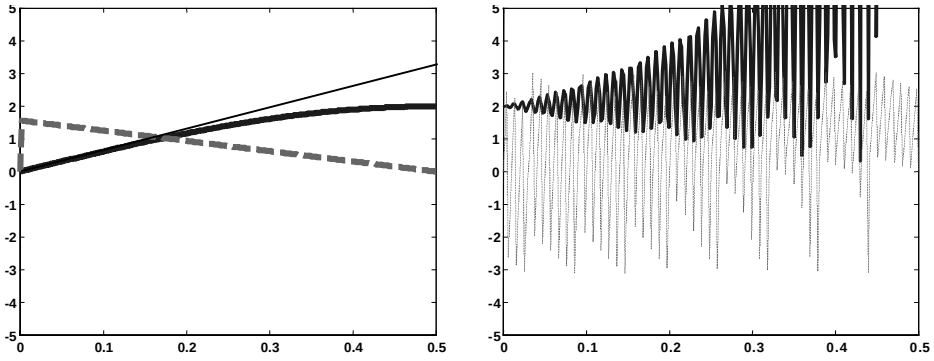
The vertical scales are amplitude, and phase in radians. The frequency scale across the bottom is normalised to the sampling frequency. The amplitude is the solid line (blue in colour) and forms a ramp proportional to the frequency. The phase is represented by the dashed grey line (purple in colour) with a constant value at 1.5708 = $\pi/2$, or a 90 degree phase shift. (Note there is no phase shift at zero frequency; the DC value.)

Fourier domains of the basic time domain filters

The Fourier domain representation of the (1, -1) derivative approximation is shown if Figure 4a, and the "improved" recursive filter shown in 4b. With the (1, -1) filter, note the loss of high frequencies relative to the theoretical shape (thin black) line and the change in phase shift as the frequency increases. With the recursive filter, note the instability of both amplitude and phase, and that the amplitude tends to a high value at the maximum frequency. The instability of the IIR filter can be improved by including a "damping factor" *r* that reduces oscillations in the filter response,

$$p_n \approx \frac{2(y_n - ry_{n-1})}{\delta x} - rp_{n-1} \tag{11}$$

Examples with *r* equal to 0.99 and 0.985 are shown in Figure 5.

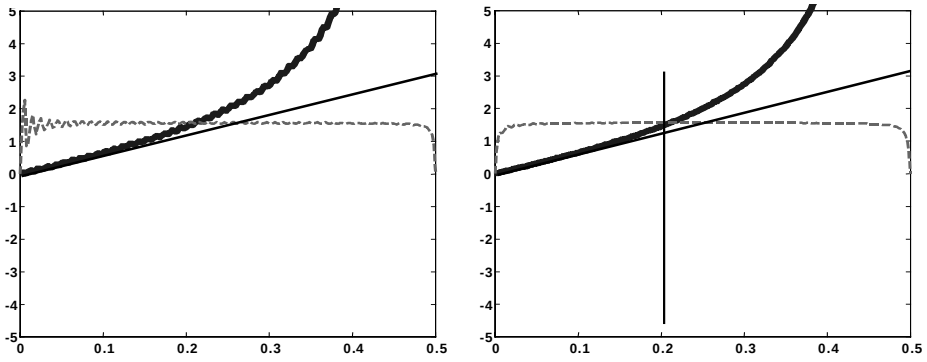


a. b.

Figure 4. Fourier transform of a) the (1, -1) approximation to the derivative and b) the "improved" recursive filter.

Note that damping stabilises the response, but the amplitude still tends to a high value at the higher frequencies. At a larger damping factor (r = 0.085 in Figure 5b), the phase deviates at the lower frequencies.

Typically, seismic data is over-sampled, so the frequency band of interest may not exceed a relative frequency of 0.2, indicated by the vertical line in Figure 5b. Thus, a filter with this response might be adequate.



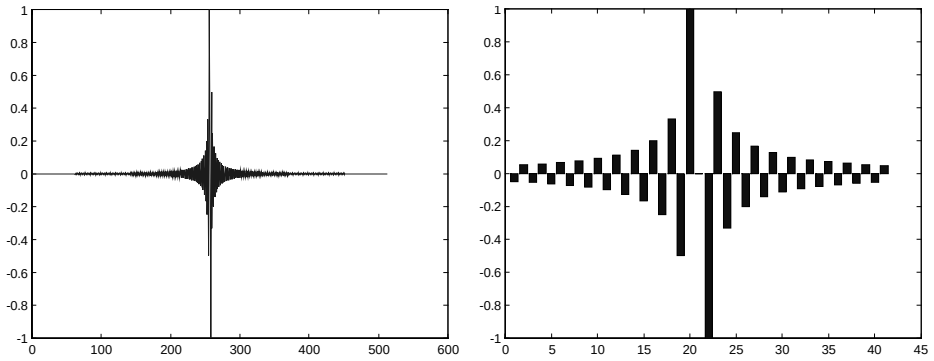
a. **b.**

Figure 5. Examples of the recursive filter with damping factors of a) 0.99, and b) 0.985.

Ideal time domain derivative

The ideal time domain filter may be found by taking the inverse Fourier transform of the filter defined in the frequency domain. We used 257 complex points for the frequency domain definition, which then gave a time domain filter with 512 samples. The filter is anti-symmetric and centered about the first sample. The negative time samples start at the last sample (512) then decrement. For viewing convenience, the array was shifted so that the actual time zero now appears at sample 256 (Figure 6a). A close-up in Figure 6b shows *t = 0* at sample 21. A partial listing of these samples follow, with the zero time sample in bold print and underlined. The filter is anti-symmetric with the amplitude at time zero equal to zero.

```
-0.0497    0.0524   -0.0553    0.0586   -0.0623    0.0665   -0.0713
 0.0768   -0.0832    0.0908   -0.0999    0.1110   -0.1249    0.1428
-0.1666    0.1999   -0.2499    0.3333   -0.5000    1.0000        0
-1.0000    0.5000   -0.3333    0.2499   -0.1999    0.1666   -0.1428
 0.1249   -0.1110    0.0999   -0.0908    0.0832   -0.0768    0.0713
-0.0665    0.0623   -0.0586    0.0553   -0.0524    0.0497
```



a. **b.**

Figure 6. Plot of the ideal differential filter, shifted so that sample 0 is plotted in a) at sample location 256, with a close-up plot in b) with sample 0 plotted at 21.

Note the filter has the form (…, 1, 0, -1, …) and not (1, -1). The Fourier transform of (1, 0, -1) is shown in Figure 7a. The amplitude linearity can be improved over the

seismic band by adding two additional terms giving the five point filter (-0.15, 1, 0, -1, 0.15) with its Fourier domain representation in Figure 7b.

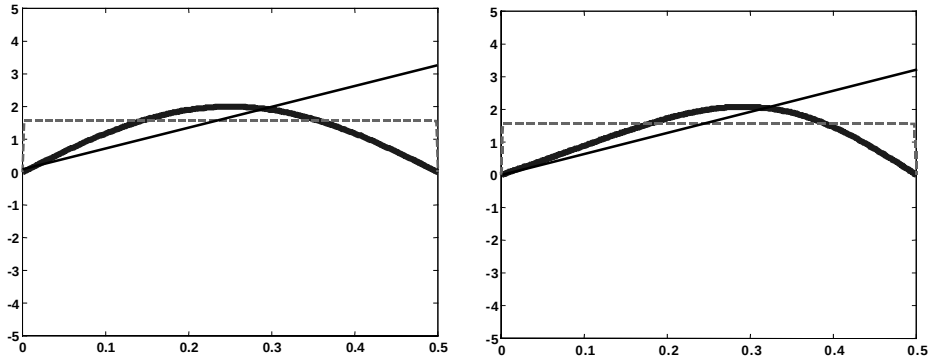


Figure 7. The Fourier transform of convolution filter a) (1, 0, -1), and b) (-0.15, 1,.0, -1, 0.15).

The (1, 0, -1) convolutional filter gives excellent results in the seismic band, especially at the lower frequencies. The five-point filter gives a slightly improved linear amplitude response over the seismic band. However, the three-point filter (1, 0, and -1) is much faster as only two points are used, and no scaling is required.

THE SECOND DERIVATIVE

The ideal second derivative response is shown in Figure 8 where the amplitude spectrum increases with square of the frequency, and the phase is a constant π.

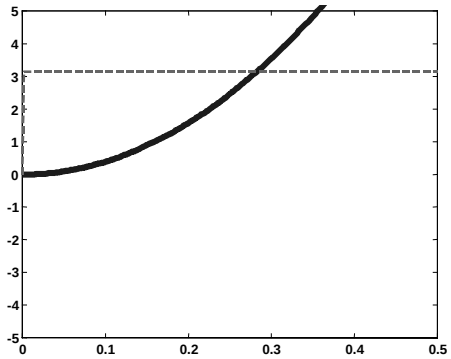
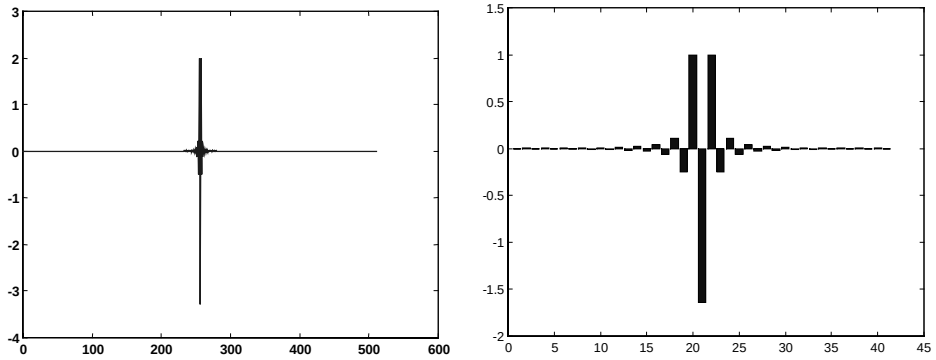


Figure 8. The ideal response of the second order derivative.

The time domain filter may be computed from the spectrum definition using an inverse Fourier transform. The complete response, shifted by 256 samples, is shown in Figure 9a. A close up, shifted by 20 samples is shown in Figure 9b and is listed below with the zero time point shifted to the 21st position..

```
-0.0025    0.0028   -0.0031    0.0035   -0.0039    0.0045   -0.0051
 0.0059   -0.0070    0.0083   -0.0100    0.0124   -0.0156    0.0204
-0.0278    0.0400   -0.0625    0.1111   -0.2500    1.0000   -1.6449
 1.0000   -0.2500    0.1111   -0.0625    0.0400   -0.0278    0.0204
-0.0156    0.0124   -0.0100    0.0083   -0.0070    0.0059   -0.0051
 0.0045   -0.0039    0.0035   -0.0031    0.0028   -0.0025
```



a. b.

Figure 9. The time domain representations of the exact second order differential with a) showing all points and b) a close-up of the shifted samples.

An estimate of the second derivative can be found by starting with the Taylor series for p(x + Δx) and (x – Δx) ;

$$p(x+\Delta x) = p(x) + (\Delta x)\frac{\partial p}{\partial x}(x) + \frac{(\Delta x)^2}{2!}\frac{\partial^2 p}{\partial x^2}(x) + \frac{(\Delta x)^3}{3!}\frac{\partial^3 p}{\partial x^3} + \frac{(\Delta x)^4}{4!}\frac{\partial^4 p}{\partial x^4} + \ldots \quad (12)$$

$$p(x-\Delta x) = p(x) - (\Delta x)\frac{\partial p}{\partial x}(x) + \frac{(\Delta x)^2}{2!}\frac{\partial^2 p}{\partial x^2}(x) - \frac{(\Delta x)^3}{3!}\frac{\partial^3 p}{\partial x^3} + \frac{(\Delta x)^4}{4!}\frac{\partial^4 p}{\partial x^4} - \ldots \; . \quad (13)$$

Summing, truncating, and bringing *p(x)* to the left we get,

$$p(x-\Delta x) - 2p(x) + p(x+\Delta x) \;\; = \;\; (\Delta x)^2 \frac{\partial^2 p}{\partial x^2}(x) \, . \quad (14)$$

Solving for the second derivative gives a first order approximation to the second derivative,

$$\frac{\partial^2 p}{\partial x^2}(x) \approx \frac{T}{(\Delta x)^2} p(x) \, , \quad (15)$$

where T is the familiar (1, -2, 1) operator.

The series could have been truncated after the fourth order derivative to give a recursive solution (Brysk 1983), but that exercise is left to the interested reader.

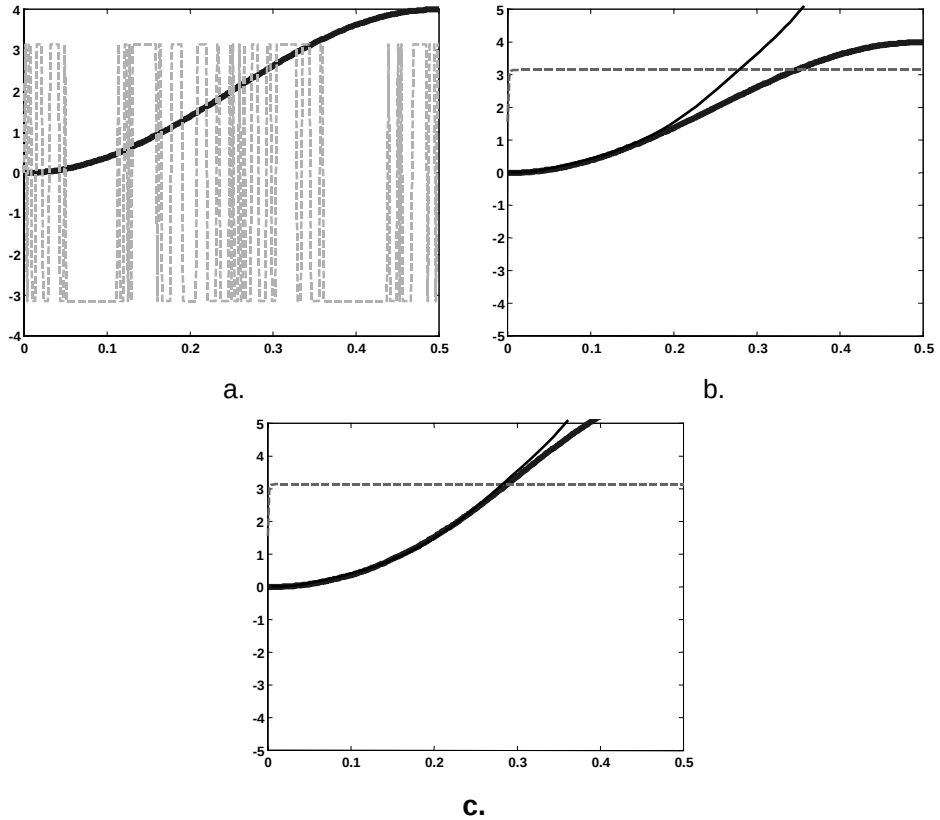


Figure 10. The spectrum of the (1, 2 ,1) approximation to the second order differential with a) showing the exact response b) the response with 0.00001 added to the imaginary value before the phase is estimated, and c) a five point filter defined in the text.

The second derivative may also be estimated by using equation (8) and taking the derivative of the derivative. This method will give the identical result defined in equation (14) or the (1, -2, 1) convolutional operator.

The spectrum of the (1, -2, 1) approximation to the second order differential appears in Figure 10. The exact response (a) has a problem computing the phase at $\pm \pi$ as the imaginary term is almost zero. Adding 0.00001 to the imaginary value stabilises the phase plot. This modification is for display purposes and does not affect the performance of the filter. The theoretical amplitude curve is included as a thin black line in Figures 10b and 10c. Note that the amplitude of the spectrum is approximately parabolic over the seismic frequencies. Figure 10c used a five-point filter (-0.1, 1, -1.8, 1, -0.1) with the amplitude spectrum scaled by an additional 1.5 to illustrate the accurate fit over a larger frequency range.

The (1, -2, 1) filter appears to be best for seismic applications as no multiplications are required (i.e., 2 = 1+1).

SQUARE ROOT DIFFERENTIAL

The square root differential is difficult to conceptualise in the time domain. As with the previous examples it is easily defined in the frequency domain as a 45 degree or $\pi/4$ phase shift, with an amplitude spectrum proportional to the square root of the frequency. Figure 11a shows the amplitude and phase scaled the same as previous plots. Figure 11b is the same spectra plotted at an expanded vertical scale. This new scale will also be used in the following figures.

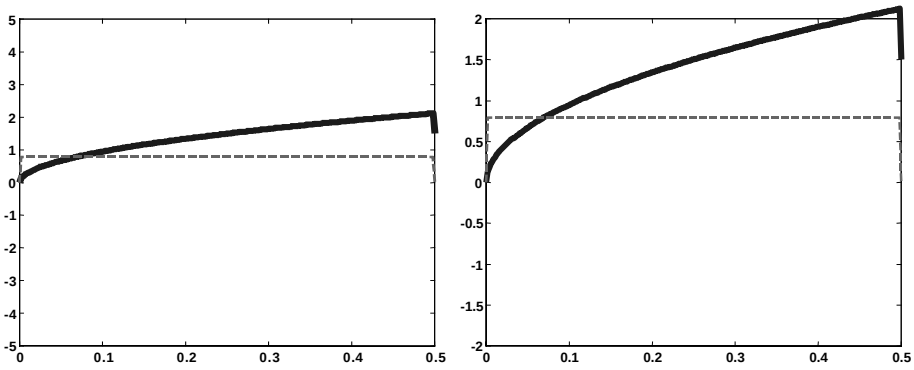


Figure 11. The square root differential filter defined in the frequency domain.

The corresponding time domain responses are shown in Figure 12, with (a) showing all 512 points, (shifted by 256 samples), and (b) an expanded horizontal scale that highlights the details around the central point (shifted by 20 samples). These samples are listed below with the central point in bold and underlined.

```
-0.0198    0.0207   -0.0219    0.0231   -0.0247    0.0262   -0.0282
 0.0302   -0.0328    0.0356   -0.0393    0.0434   -0.0489    0.0556
-0.0648    0.0772   -0.0960    0.1263   -0.1851    0.3467   0.8355
-0.7496    0.1165   -0.1947    0.0685   -0.1077    0.0490   -0.0736
 0.0383   -0.0556    0.0315   -0.0445    0.0268   -0.0371    0.0233
-0.0317    0.0206   -0.0277    0.0185   -0.0245    0.0168
```

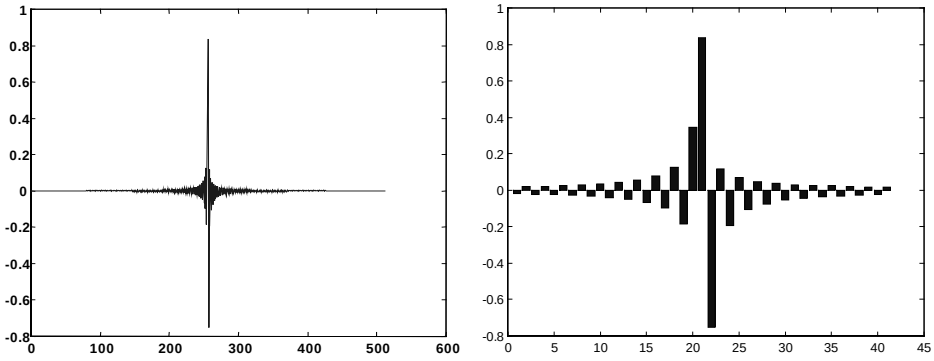
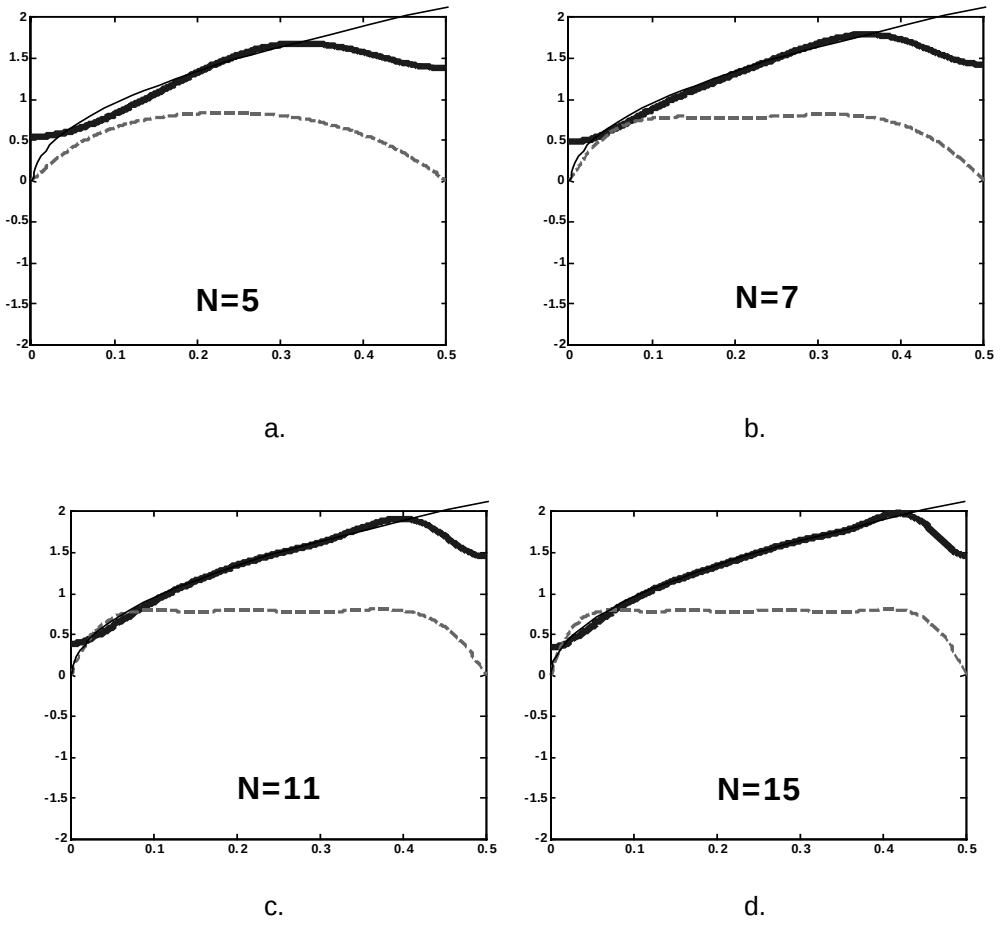


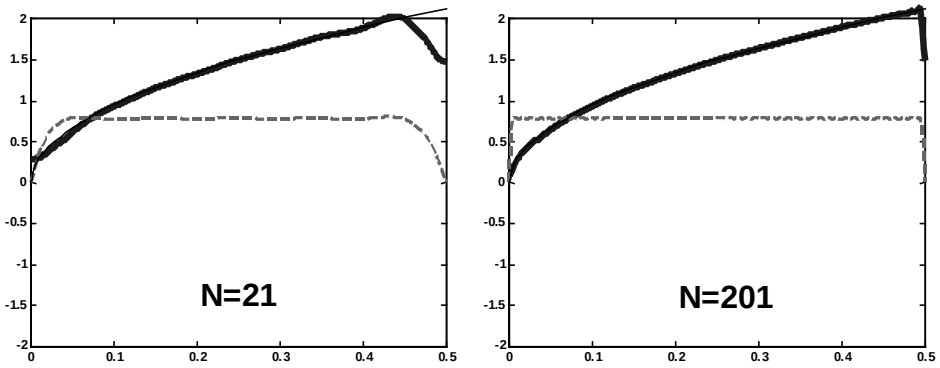
Figure 12. Time domain response of the root differential filter, a) showing the full response for 512 samples (t=0 at sample 256) and b) a closer view (t=0 at sample 21).

The rjw filter is asymmetric and is difficult to design as a short filter. One practical method is to window the long theoretical filter with a short cosine bell. The number of samples in the filter is defined by the size of the cosine window, with the amplitudes smoothly tapered to zero by the cosine weighting. The number of samples in a filter is defined by *N*, which, for a symmetric application of the cosine, is always odd.

The following filters were designed by applying cosine windows that varied from $N = 5$ to N = 201. The results are displayed in Figure 13. For small *N*, the amplitude spectrum has a non-zero DC amplitude $(\omega = 0)$ and does not match the theoretical shape (displayed as the thin black line). The phase is roughly 45 degrees, or $\pi/4$, but deviates to smaller values as the frequency tends to zero. The choice of a filter becomes a compromise between speed and acceptable accuracy over the frequency band of interest. (Note that as the size of the filter approaches 50 samples, it may be faster to use the Fourier transform and get exact results).

Ensuring samples in the filter sum to zero may reduce the frequency error at zero frequency.





e. f.

Figure 11. Approximations to the rjw filter using various cosine windows on the theoretical response with, a) N=5, b) N=7, c) N=11, d) N=15, e) N=21, f) N=201.

CONCLUSIONS

A number of filters designs were presented for approximating a derivative, a second order derivative, and a square-root derivative (rjw) over the seismic band of interest. The convolutional form was the best in all cases. Two point filters best approximated the first and second order derivatives. Applying a cosine window to the ideal impulse response however, best approximated the rjw filter. A number of window sizes were presented.

REFERENCES

Brysk, H., 1983, Numerical analysis of the 45-degree finite difference equation for migration, Geophysics, Vol. 48, p. 532 – 542.

Liner, C. L., 1990 General theory and comparative anatomy of dip moveout, Geophysics, Vol. 55, p. 595-607.

APPENDIX 1

Various rjw filters with different N. Time zero is in bold and underlined.

N = 7,
(0.0483, -0.1309, 0.3203, **0.8355**, -0.6925, 0.0823, -0.0745)

N = 11,
(0.0200, -0.0480, 0.0893, -0.1603, 0.3349,
0.8355, -0.7241, 0.1008, -0.1377, 0.0342, -0.0279)

N = 15,
(0.0109, -0.0248, 0.0429, -0.0679, 0.1050, -0.1710, 0.3400,
0.8355,
-0.7352, 0.1076, -0.1619, 0.0484, -0.0598, 0.0188, -0.0144)

N = 21,
(-0.0056, 0.0122, -0.0203, 0.0301, -0.0424, 0.0584, -0.0807, 0.1149, -0.1776, 0.3431,
0.8355,
-0.7420, 0.1117, -0.1771, 0.0576, -0.0814, 0.0321, -0.0398, 0.0159, -0.0157, 0.0045)