Butterworth dip filters

Dave Hale and Jon F. Claerbout

Abstract

Dip filters enable a geophysicist to discriminate between various seismic events on the basis of apparent dip. The frequency-wavenumber (ω,k) domain seems an attractive domain to perform dip filtering, because it permits the application of an arbitrary transfer function of dip. Seismic applications of dip filtering, however, seldom require the flexibility offered by the (ω,k) domain; and one may often be willing to sacrifice this flexibility to obtain features not possible with (ω,k) domain filters, but readily available with time-space (t,x) domain filters. Examples are

(1) time and space variability,

(2) flexible treatment of computational grid boundaries,

(3) an efficient, recursive implementation.

We describe a (t,x) domain dip filtering method with these features.

In the derivation of a (t,x) domain filter, we first discuss (t,k) and (ω,x) domain dip filters. While not fully possessing the advantages of a (t,x) domain filter, these filters are an attractive combination of two very efficient and commonly available processes: (1) one-dimensional Butterworth filtering and (2) one-dimensional Fourier transforms. We then derive (t,x) domain approximations to these filters which have the features noted above.

Introduction

Because geophysicists can often distinguish desired seismic signals from unwanted "noise" on the basis of apparent dip (or velocity), dip filters have found widespread use in seismic data processing. A dip filter may, for example, be used to attenuate surface waves which, in a common-midpoint gather, have greater apparent dip (lower apparent velocity) than subsurface reflections. Previous papers pertaining to dip filtering have described applications of this tool and/or algorithms for its implementation. (See the references for general reading.) This paper falls into the algorithmic category. We describe a method of dip filtering which has been used effectively in a variety of applications to seismic data, both by researchers at Stanford and by members of the seismic data processing industry.

A dip filter is best specified by its frequency-wavenumber (ω,k) domain transfer function which, ideally, should have radial contours of constant amplitude and phase extending from the origin. As illustrated in Figure 1, each contour of a dip filter should represent a constant ratio k/ω, corresponding to an apparent dip in the time-space (t,x) domain. Letting $p(t,x)$ denote input data and $q(t,x)$ filtered data, Figure 1 suggests the following computational method of dip filtering:

$$(\omega,k) \textit{ domain dip filter}$$

(1) $P(\omega,k) = FFT[\, p(t,x)\,]$

(2) $Q(\omega,k) = H(\omega,k)\, P(\omega,k) = \tilde{H}(k/\omega)\, P(\omega,k)$

(3) $q(t,x) = IFT[\, Q(\omega,k)\,]$

FFT and IFT denote forward and inverse Fourier transforms, respectively. This commonly used method is probably the most accurate way to apply an arbitrary transfer function of dip $H(\omega,k) = \tilde{H}(k/\omega)$. In practice, however, dip filtering in the (ω,k) domain presents problems. One problem stems from the temporal and spatial non-stationarity of seismic data. For example, the dips of reflections in a common-midpoint-gather change with both time and offset because of normal moveout. Non-stationarity makes a time and space variable filter desirable, and the Fourier transforms in steps (1) and (3) make the above method inherently time and space invariant. Another problem is that, when applied to sampled data, this method suffers from the periodic boundary conditions ("wraparound") implied by the use of discrete Fourier transforms. These problems, together with the fact that two-dimensional Fourier transforms are computationally expensive, motivate the use of a (t,x) domain filter.

(t,x) domain dip filters permit time and space variability and a flexible (non-periodic) treatment of computational grid boundaries. With care taken to ensure stability, recursive

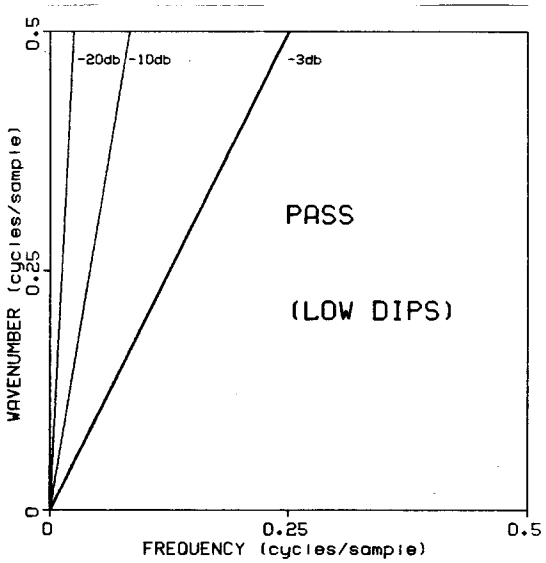


FIG. 1. Contours of constant amplitude and phase for an ideal dip filter are given by radial lines in the frequency-wavenumber (ω,k) domain. The transfer function $H(\omega,k)$ of an ideal dip filter depends only on the ratio k/ω, which corresponds to apparent dip in the time-space (t,x) domain. This particular filter passes low dips and rejects high dips.

implementations of these filters are more efficient than non-recursive algorithms. The two-dimensional dip filters we derive in this paper are recursive in either the time or space dimension, but not both. First, we derive recursive (t,k) or (ω,x) domain dip filters. After establishing the conditions for stability of these filters, the derivation of a stable, recursive (t,x) domain filter follows easily.

(t,k) or (ω,x) domain dip filters

We assume that the desired dip filter can be expressed as a cascade of filters having power spectra of the following forms:

$$\textit{High-dip-pass} \qquad |H(\omega,k)|^2 = \frac{1}{1+\left(\frac{D\omega}{k}\right)^{2n}} \tag{1a}$$

$$\textit{Low-dip-pass} \qquad |H(\omega,k)|^2 = \frac{1}{1+\left(\frac{k}{D\omega}\right)^{2n}} \tag{1b}$$

where D is a user-specified cutoff dip, the half-power point, and n is a positive integer which determines the steepness of the boundary between pass and reject zones. The

contours plotted in Figure 1 are of $20 \log_{10} |H(\omega,k)|$ for the low-dip-pass filter of equation (1b) with $D = 2$ and $n = 1$.

Note that the filters of equations (1) cannot distinguish between positive and negative dips D, because the power spectra are symmetric with respect to the frequency and wavenumber axes; i.e., $|H(\omega,k)| = |H(-\omega,k)|$ and $|H(\omega,k)| = |H(\omega,-k)|$. We chose these power spectra to resemble those of one-dimensional Butterworth filters

$$\textit{Low-frequency-pass} \qquad |H(\omega)|^2 = \frac{1}{1 + \left(\frac{\omega}{\Omega}\right)^{2n}} \tag{2a}$$

$$\textit{High-frequency-pass} \qquad |H(\omega)|^2 = \frac{1}{1 + \left(\frac{\Omega}{\omega}\right)^{2n}} \tag{2b}$$

where Ω is a user-specified cutoff frequency. Methods for implementing these one-dimensional Butterworth filters in the time domain are well known (e.g., Oppenheim and Schafer, 1975), and their use is common in seismic data processing. The similarity between equations (1) and (2) implies that we may perform dip filtering in the (t,k) domain by the following algorithm:

$$(t,k) \textit{ domain dip filter}$$

(1) $P(t,k) = FFT[\, p(t,x)\,]$

(2) $Q(t,k) = Butterworth[\, P(t,k)\,] \quad ; \quad \Omega = k / D$

(3) $q(t,x) = IFT[\, Q(t,k)\,]$

Steps (1) and (3) are Fourier transforms over the space dimension only. Step (2) is the application, for all k, of a conventional, time domain Butterworth filter with the cutoff frequency $\Omega = k / D$. Note that we must use a low-frequency-pass filter to perform high-dip-pass filtering; likewise, a high-frequency-pass filter yields a low-dip-pass filter.

To appreciate the simplicity of the (t,k) domain filter, and to lay the foundation for the derivation of a (t,x) domain dip filter, we should review the key steps in the implementation of discrete, one-dimensional Butterworth filters. Our review will be brief, including only those points relevant to dip filters.

Discrete filters with spectra approximating those of equations (2) may be derived in many ways. The simplest method which is valid for both low and high-pass filters is to use the bilinear transformation

$$-i\omega \approx -2i\tan\left(\frac{\omega}{2}\right) = 2\frac{1-Z}{1+Z}$$

where Z is the unit delay operator defined by $Z \equiv e^{i\omega}$. For notational simplicity, we have assumed a unit sampling interval. The bilinear transformation is particularly attractive because it guarantees a causal and stable discrete filter $H(Z)$, given a causal and stable continuous filter $h(t)$. For $h(t)$ to be causal and stable, the poles of its Fourier transform $H(\omega)$ must lie in the lower half of the complex ω-plane. From equations (2), one may readily verify that the $2n$ poles of $|H(\omega)|^2$ lie equally spaced around a circle of radius $|\Omega|$ at locations

$$\nu_j \equiv |\Omega| e^{-i(2j+1)\pi/(2n)} \quad ; \quad j = 0, 1, \cdots 2n-1 \tag{3}$$

Keeping only those poles in the lower half-plane (i.e., the first n poles), we may express $H(\omega)$ as a cascade of single-pole filters:

$$\textit{Low-frequency-pass} \qquad H(\omega) = \prod_{j=0}^{n-1} \frac{i\nu_j}{-i\omega + i\nu_j} \tag{4a}$$

$$\textit{High-frequency-pass} \qquad H(\omega) = \prod_{j=0}^{n-1} \frac{-i\omega}{-i\omega + i\nu_j} \tag{4b}$$

for which the bilinear transformation yields

$$\textit{Low-frequency-pass} \qquad H(Z) = \prod_{j=0}^{n-1} \frac{i\nu_j(1+Z)}{2 + i\nu_j - (2 - i\nu_j)Z} \tag{5a}$$

$$\textit{High-frequency-pass} \qquad H(Z) = \prod_{j=0}^{n-1} \frac{2(1-Z)}{2 + i\nu_j - (2 - i\nu_j)Z} \tag{5b}$$

Because the bilinear transformation warps the frequency axis, particularly at the higher frequencies, the discrete filters in equations (5) are low-frequency approximations to the continuous filters of equations (4). This non-linear distortion is actually desirable, for it prevents aliasing by mapping the entire continuous frequency axis $|\omega| < \infty$ to the segment $|\omega| < \pi$. However, if we wish the cutoff frequency Ω to be the half-power point of the discrete filters, as it is for the continuous filters, we must "pre-warp" Ω by replacing it with $2\tan(|\Omega|/2)$ in equation (3).

Having reviewed the fundamentals of discrete Butterworth filters, we may now examine (t,k) domain dip filters more closely. As already shown, these dip filters require the application of a time domain Butterworth filter for each wavenumber k, with the cutoff frequency set to $\Omega = k/D$. All goes well until $|k/D| \geq \pi$; note that the simple pre-warping of Ω is not valid for $\Omega \geq \pi$. To solve this problem, we replace $\Omega = k/D$ in equation (3) with $B(k,D)$

defined by

$$B(k,D) \equiv \begin{cases} 2\tan\left|\frac{k}{2D}\right| & ; \quad \left|\frac{k}{D}\right| < \pi \\ \infty & ; \quad \left|\frac{k}{D}\right| \geq \pi \end{cases} \tag{6}$$

so that the poles ν_j become

$$\nu_j \equiv B(k,D)e^{-i(2j+1)\pi/(2n)} \quad ; \quad j = 0, 1, \cdots n-1 \tag{7}$$

In practical computations, a very large number (say 10^{10}) may be used in place of infinity in equation (6). As examples, the single-pole (n = 1) high and low-dip-pass filters derived from equations (5), (6), and (7) are

Single-pole high-dip-pass

$$(2+B_k)Q_{t,k} = (2-B_k)Q_{t-1,k} + B_k(P_{t,k} + P_{t-1,k}) \tag{8a}$$

Single-pole low-dip-pass

$$(2+B_k)Q_{t,k} = (2-B_k)Q_{t-1,k} + 2(P_{t,k} - P_{t-1,k}) \tag{8b}$$

where the recursions begin at t = 0. (To simplify notation, we use subscripts for sampled coordinates and avoid explicitly stating the dependence of B_k on the dip cutoff D.) Boundary conditions are determined by the choice of $P_{-1,k}$ and $Q_{-1,k}$. For example, causal input data implies $P_{-1,k} = Q_{-1,k} = 0$.

The amplitude spectrum of a single-pole, *band*-dip-pass filter is contoured in Figure 2. This spectrum was obtained by cascading the high and low-dip-pass filters of equations (8) with $D = 1/2$ and $D = 2$, respectively. As expected, the half-power (-3 db) contours are straight, because we have placed the poles according to equations (6) and (7). Other contours, however, remain warped by the bilinear transformation. This unavoidable distortion is most severe at the higher frequencies for which the bilinear transformation is a poor approximation. Steeper high and low dip cutoffs are obtained by cascading more of the recursive factors in equations (5). The amplitude spectrum of a six-pole (n = 6), band-dip-pass filter is contoured in Figure 3.

To obtain (ω,x) domain dip filters, one simply swaps the time and space dimensions in the (t,k) domain filters derived above. For seismic data, we note the following advantages of dip filtering in the (ω,x) domain.

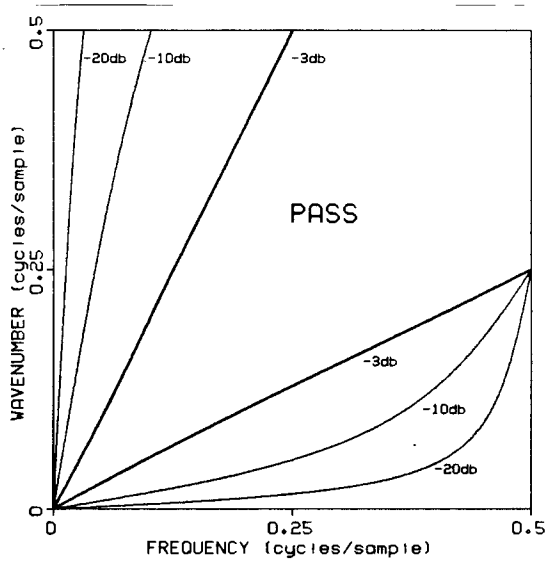


FIG. 2. Contours of constant amplitude for a single-pole (t,k) domain band-dip-pass filter, implemented via equations (8). Note that the (bold) -3 db contour is radial as for the ideal filter, but that other contours are warped by the bilinear transformation, particularly at higher frequencies.

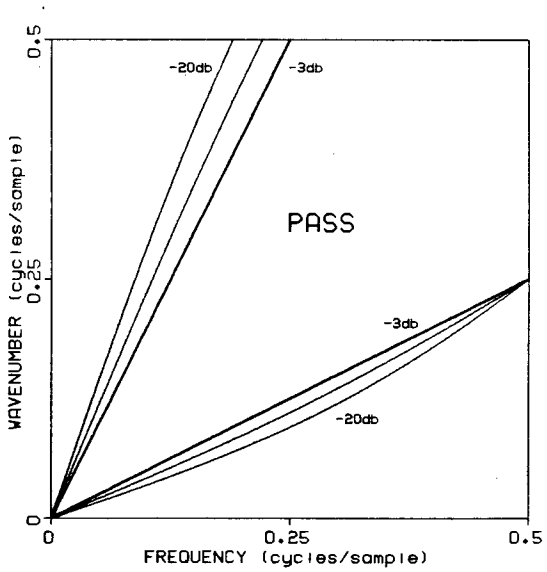


FIG. 3. Contours of constant amplitude for a six-pole (t,k) domain band-dip-pass filter. Note the increased steepness in the boundary between pass and reject zones when compared with Figure 2.

(1) The spatial axis of seismic data is usually more severely truncated than the temporal axis; therefore, the periodic boundary conditions imposed by the use of discrete Fourier transforms are most annoying in the spatial dimension. Padding the spatial dimension with zeros (or whatever seems best) may help, but at the cost of processing more traces. The recursive filters of equations (5) permit the specification of arbitrary boundary conditions for no added cost.

(2) Each recursive factor in equations (5) may be cascaded separately; and each factor requires only three (Fourier transformed) traces, the present and past input traces and the previous output trace, to compute the present output trace. [Recall the examples of equations (8).] Therefore, one may apply dip filtering for about $3n$ times the cost of one-dimensional, frequency-domain filtering which, of course, may be applied concurrently.

(3) Seismic data is typically stored by traces so that temporal Fourier transforms are computationally simpler and cheaper than spatial Fourier transforms.

Note that the (t,k) and (ω,x) domain filters permit non-stationarity in one dimension. The (t,k) domain filter of equation (8), for example, may be made time-varying simply by letting the dip cutoff be a function of time, $D = D(t)$. For some applications, particularly if time *and* space variability is needed, a (t,x) domain dip filter may be preferred. In the next section, we derive a recursive (t,x) domain dip filter based on the (t,k) domain algorithm described above.

(t,x) domain dip filters

The derivation of (t,x) domain dip filters from (t,k) or (ω,x) domain filters is best illustrated by the example of equation (8a). Each multiplication in the wavenumber domain may be replaced by a convolution in the space domain to obtain

$$(2\delta_x + b_x) * q_{t,x} = (2\delta_x - b_x) * q_{t-1,x} + b_x * (p_{t,x} + p_{t-1,x}) \tag{9}$$

where b_x and δ_x are defined by

$$b_x = IFT[\, B_k \,]$$

and

$$\delta_x = \begin{cases} 1 & ; \quad x = 0 \\ 0 & ; \quad |x| \neq 0 \end{cases}$$

Note that solving for the unknown $Q_{t,k}$ in equation (8a) is rather simple, requiring only that

we divide both sides of that equation by the scalar $2+B_k$. The solution of equation (9), however, may be more difficult and costly, for we must *deconvolve* the filter $2\delta_x+b_x$. For the (t,x) domain filter of equation (9) to be computationally practical, we must make this deconvolution inexpensive.

One way to speed up the deconvolution is to replace the filter b_x with a very short approximation which we shall call a_x. As an example, let us choose the length of a_x to be 3 and suppose we have only four traces. Equation (9) may then be expressed in matrix form as

$$\begin{bmatrix} 2+a_0 & a_1 & 0 & 0 \\ a_{-1} & 2+a_0 & a_1 & 0 \\ 0 & a_{-1} & 2+a_0 & a_1 \\ 0 & 0 & a_{-1} & 2+a_0 \end{bmatrix} \begin{bmatrix} q_{t,1} \\ q_{t,2} \\ q_{t,3} \\ q_{t,4} \end{bmatrix} = \begin{bmatrix} r_{t,1} \\ r_{t,2} \\ r_{t,3} \\ r_{t,4} \end{bmatrix} \tag{10}$$

where $r_{t,x}$ denotes the right-hand side of equation (9). To find the unknown $q_{t,x}$, we must solve this tridiagonal system of equations. Fast algorithms for solving tridiagonal systems are well known and are, in fact, used routinely in finite-difference migration programs (Claerbout, 1976). These algorithms make the solution of tridiagonal systems [like equation (10)] quite efficient and practical.

The computational advantage in solving tridiagonal (or any banded) systems becomes particularly apparent for large systems, for the cost increases only linearly with the size of the system. Letting **A** denote an $m\times m$ matrix with the coefficients of a_x on its diagonals as in equation (10), we express the (t,x) domain versions of equations (8) for m traces as

Single-pole high-dip-pass

$$(2\mathbf{I}+\mathbf{A})\mathbf{q}_t = (2\mathbf{I}-\mathbf{A})\mathbf{q}_{t-1} + \mathbf{A}(\mathbf{p}_t + \mathbf{p}_{t-1}) \tag{11a}$$

Single-pole low-dip-pass

$$(2\mathbf{I}+\mathbf{A})\mathbf{q}_t = (2\mathbf{I}-\mathbf{A})\mathbf{q}_{t-1} + 2(\mathbf{p}_t - \mathbf{p}_{t-1}) \tag{11b}$$

where

$$\mathbf{q}_t \equiv (q_{t,1} \quad q_{t,2} \quad \cdots \quad q_{t,m})^T$$

and

$$\mathbf{p}_t \equiv (p_{t,1} \quad p_{t,2} \quad \cdots \quad p_{t,m})^T.$$

The real problem with implementing equations (11) lies not in their solution, but rather in finding a suitable approximation a_x to b_x. This approximation must satisfy several conditions. Because B_k is real and even, b_x and, hence, a_x must be real and even. Recall also that B_k is non-negative for all k. This condition is, in fact, necessary to ensure stability of the (t,k) domain dip filter, for it guarantees that none of the poles ν_j lie in the upper half of the complex ω-plane. [See equation (7).] Therefore, A_k must also be non-negative which implies that **A** must be a positive semi-definite matrix.

A further condition on **A** stems from our stated goal of time *and* space variability. Recall that the coefficients a_x on the diagonals of **A** depend on the dip cutoff D. To make the filters of equations (11) time variable, we let the approximation a_x be a function of time; equivalently, $\mathbf{A} = \mathbf{A}(t)$. Space variability is obtained by letting a_x vary down the diagonals of **A** (making **A** non-Toeplitz), while keeping **A** positive semi-definite. In practice, the approximation a_x should depend on D in some simple way so that the cost of computing the matrix **A** is not unreasonable.

Subject to the conditions of a positive semi-definite **A** and a simple dependence on D, we seek an a_x with Fourier transform approximating $B(k)$ given by equation (6). To this end, we approximate $B(k)$ by the first term of its Maclaurin series:

$$\tilde{B}(k) \equiv \left| \frac{k}{D} \right|$$

Zeroing $\tilde{B}(k)$ for wavenumbers greater than the Nyquist and inverse Fourier transforming, we obtain (Bracewell, 1978)

$$\tilde{b}(x) \equiv \frac{\pi}{D} \left[\text{sinc}(x) - \frac{1}{2} \text{sinc}^2\left(\frac{x}{2}\right) \right]$$

which we may sample without aliasing to obtain

$$\tilde{b}_x = \frac{1}{D} \begin{cases} \frac{\pi}{2} & ; \quad x = 0 \\ -\frac{2}{\pi x^2} & ; \quad |x| = 1, 3, 5, \cdots \\ 0 & ; \quad \textit{otherwise} \end{cases}$$

Let us try the 3-coefficient approximation $a_x = (-2/\pi, \pi/2, -2/\pi)/D$, obtained by truncating $\tilde{b}_x$ for $|x| > 2$. Recall that these three coefficients lie along the three main diagonals of the tridiagonal matrix **A**. Because the sum of off-diagonal elements in any row of **A** is less in magnitude than the diagonal element, **A** is diagonally dominant and, by Gershgorin's theorem, positive-definite (Strang, 1980). And because each element in this

tridiagonal approximation is a simple, linear function of $1/D$, a time and space variable implementation is very efficient.

Substituting **A** into equations (11) and cascading high and low-dip-pass filters with $D = 1/2$ and $D = 2$ as before, we obtain the amplitude response contoured in Figure 4. Note that the half-power (-3 db) contour, which was straight for the (t,k) domain filter, is now distorted due to the two approximations made in deriving the tridiagonal matrix **A**. First, by using $\tilde{B}(k)$ in place of $B(k)$, we effectively assumed that the warping caused by the bilinear transformation is negligible. This assumption is most valid when the temporal axis of the input data is oversampled and, for seismic data, may be quite reasonable. For example, if the input data contains little energy in the band of frequencies above half-Nyquist, then we should not be concerned with distortion in that band. In practice, we should direct our efforts at improving the transfer function in the so called "seismic band".

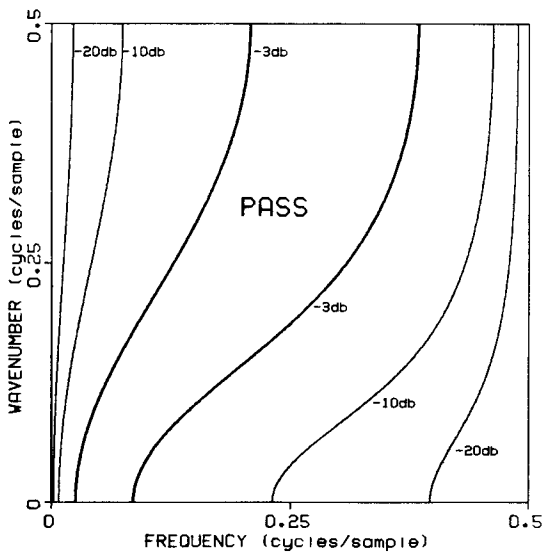


FIG. 4. Contours of constant amplitude for a single-pole tridiagonal (t,x) domain band-dip-pass filter. The coefficients of the tridiagonal approximation are $(-2/\pi, \pi/2, -2/\pi)/D$. Note the distortion at low and high wavenumbers, particularly for higher frequencies. The contours of this filter may only approximate those of the ideal dip filter over a limited frequency band.

The second approximation we made was in truncating $\tilde{b}_x$. The well-known Gibbs phenomenon affects the low and high wavenumber response primarily, because the Fourier transform of $\tilde{b}_x$ has a discontinuous first derivative at zero and Nyquist wavenumbers. We may improve the low wavenumber response by modifying the off-diagonal coefficients of **A**

according to $a_x = (-\pi/4, \pi/2, -\pi/4)/D$, so that $A_{k=0} = 0$. Cascading high and low-dip-pass filters as before, we obtain the amplitude response contoured in Figure 5.

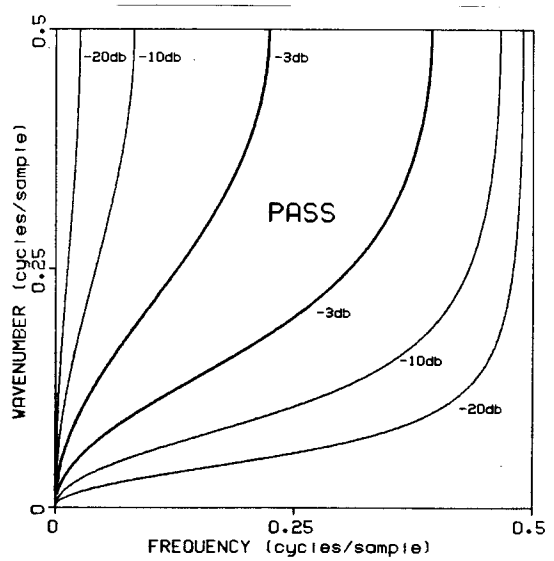


FIG. 5. Contours of constant amplitude for a single-pole tridiagonal (t,x) domain band-dip-pass filter. The coefficients of the tridiagonal approximation $(-\pi/4, \pi/2, -\pi/4)/D$ were chosen to improve the low-wavenumber response. Compare with Figure 4.

One might tune the tridiagonal approximation more finely to yield better contours over some frequency or wavenumber band, but, generally, to obtain a more ideal transfer function, we must increase the length of the approximation a_x. Equivalently, we must increase the bandwidth (the number of non-zero diagonals) of the matrix **A**. Fortunately, even though the resulting system is no longer tridiagonal, we may use the well-known Cholesky decomposition to solve the banded, positive definite system. An efficient algorithm for solving such systems has been published by Martin and Wilkinson (1965). As for tridiagonal systems, the computational cost grows linearly with the size m of the system. The cost, however, grows quadratically with bandwidth, so we should always keep the length of a_x as short as possible.

The amplitude response of the band-dip-pass filter contoured in Figure 6 was again obtained by cascading high and low-dip-pass filters, but here using an 11-diagonal **A** in equations (11). The coefficients a_x were obtained by truncating $\tilde{b}_x$ for $|x| > 6$. The remaining amplitude distortion is due almost entirely to approximating $B(k)$ with only the first term $\tilde{B}(k)$ in its Maclaurin series expansion. Including more terms [ultimately using

$B(k)$ exactly] would improve the amplitude response, but would complicate the dependence of **A** on the dip cutoff D; specifically, this dependence would no longer be linear. Furthermore, a more lengthy a_x would be required to avoid the low and high wavenumber distortion due to truncation. The 11-diagonal filter of Figure 6 represents a compromise between accuracy and computational efficiency.

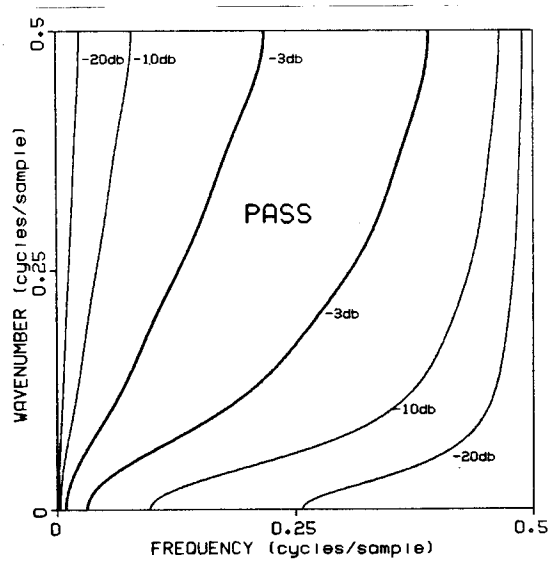


FIG. 6. Contours of constant amplitude for a single-pole 11-diagonal (t,x) domain band-dip-pass filter. Compare this amplitude response with that of the single-pole (t,k) domain filter in Figure 2. Almost all of the contour distortion is at the higher frequencies and is due to neglecting the warping effect of the bilinear transformation.

As Figures 4, 5, and 6 illustrate, the distortion of the amplitude response is most severe for small dip cutoffs D, because the accuracy of the approximation $\widetilde{B}(k)$ depends on the magnitude of $|k/D|$. In other words, the (t,x) domain dip filters described above pass or reject high dips quite accurately, but perform less well when used to pass or reject only low dips. This suggests that, for $D < 1$, we might wish to interchange the temporal and spatial axes in equations (11) and replace D with $1/D$. The resulting filter would be recursive in the spatial rather than the temporal dimension.

Conclusions

The Butterworth dip filters described above are similar in form and implementation to one-dimensional Butterworth filters. They are recursive, with well known and easily satisfied conditions for stability. (t,k) or (ω,x) domain filters combine the computational efficiency of one-dimensional Butterworth filters with that of one-dimensional discrete Fourier transforms. The simple analytical computation of coefficients for these filters makes a time *or* space varying filter practical. For demultiplexed seismic data, usually severely truncated spatially, an (ω,x) domain filter is particularly attractive.

We derived (t,x) domain Butterworth filters by inverse Fourier transforming and approximating the equations for (t,k) or (ω,x) domain filters. These (t,x) domain filters are potentially time *and* space variable; in practice, however, this potential may be realized only when one is willing to accept distortion in the filter's transfer function. While the transfer function of the (t,x) domain dip filter may not be ideal, the advantages of time and space variability and a fast, recursive implementation make this filter appealing for bandlimited data or for applications where fine discrimination between dips is not required. Such an application may be the attenuation of ground roll which typically has a much greater apparent dip than subsurface reflections. Our experience in applying (t,x) domain Butterworth dip filters to seismic data suggests that even the simple tridiagonal approximation is suitable for many applications.

ACKNOWLEDGMENT

This work was supported by the sponsors of the Stanford Exploration Project.

REFERENCES

Bracewell, R.N., 1978, The Fourier transform and its applications: New York, McGraw-Hill.
Claerbout, J.F., 1976, Fundamentals of geophysical data processing: New York, McGraw-Hill.
Martin, R.S., and Wilkinson, J.H., 1965, Symmetric decomposition of positive definite band matrices: Numerische Mathematik, v. 7, p. 355-361.
Oppenheim, A.V., and Schafer, R.W., 1975, Digital signal processing: Englewood Cliffs, Prentice-Hall, Inc.
Strang, G., 1980, Linear algebra and its applications: New York, Academic Press, Inc.

REFERENCES FOR GENERAL READING

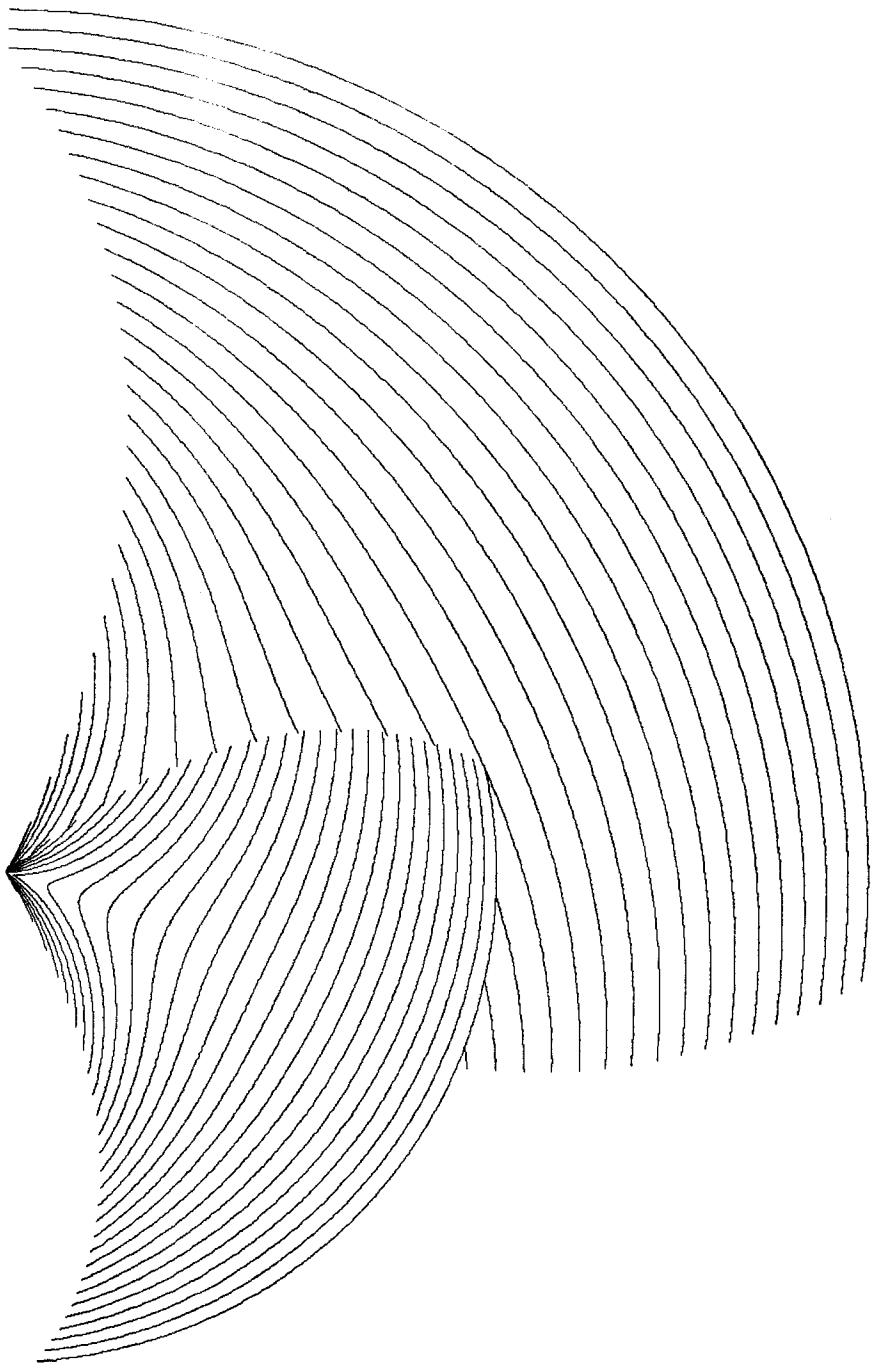
Digicon, 1978, Data processing techniques -- F-K filtering: Digicon Geophysical Corp., Houston.

Embree, P., Burg, J.P., and Backus, M.M., 1963, Wide-band velocity filtering -- the Pie Slice process: Geophysics, v. 28, p. 948-974.

Fail, J.P., and Grau, G., 1963, Les filtres en e'ventail: Geophys. Prosp., v. 11, p. 131-163.

McClellan, J.H., and Parks, T.W., 1972, Equiripple approximation of fan filters: Geophysics, v. 37, p. 573-583.

Ryu, J.V., 1982, Decomposition (DECOM) approach applied to wave field analysis with seismic reflection records: Geophysics, v. 47, p. 869-883.

Treitel, S., Shanks, J.L., and Frasier, C.W., 1967, Some aspects of fan filtering: Geophysics, v. 32, p. 689-800.

Treitel, S., and Shanks, J.L., 1971, The design of multistage separable planar filters: IEEE Trans. Geosci. Electron., v. GE-9, p. 10-27.