References

Claerbout, J.F., 1976, Fundamentals of geophysical data processing, McGraw-Hill, 274 p.

Dines, K.A. and Lytle, R.J., 1979, Computerized geophysical tomography, Proceedings of the IEEE, v. 67, p. 1065-1073.

Dix, C.H., 1952, Seismic prospecting for oil, Harper, 412 p.

Dix, C.H., 1955, Seismic velocities from surface measurements, Geophysics, v. 20, p. 68-86.

Dobrin, M., 1976, Introduction to geophysical data processing, McGraw-Hill, 630 p.

Doherty, S. and Claerbout, J., 1976, Structure independent velocity estimation, Geophysics, v. 41, p. 850-881.

Judson, D., Lin, J., Schultz, P.S., and Sherwood, J, 1978, Equalizing stacking velocities of dipping events via Devilish, paper presented at the 48'th annual meeting of the Society of Exploration Geophysicists, San Francisco, CA.

Kak, A.C., 1979, Computerized tomography with x-ray, emission, and ultrasound sources, Proceedings of the IEEE, vol. 67, p. 1245-1272.

Levin, F.K., 1971, Apparent velocities from dipping interfaces, Geophysics, v. 36, p. 510-516.

May, B.T. and Straley, D.K., 1979, Higher-order moveout spectra, Geophysics, v. 44, p. 1193-1207.

Ottolini, R., Clayton, R., and Claerbout, J.F., 1978, Migration with common-midpoint stacks, paper presented at the 48'th annual meeting of the Society of Exploration Geophysicists, San Francisco, CA.

Pollet, A., 1974, Simple velocity modelling and the continuous velocity section, paper presented at the 44'th annual meeting of the Society of Exploration Geophysicists, Dallas, TX.

Schneider, W., 1971, Developments in seismic data processing and analysis (1968-1970), Geophysics, v. 36, p. 1043-1073.

Schultz, P.S. and Claerbout, J.F., Velocity estimation and downward continuation by wavefront synthesis, Geophysics, v. 43, p. 691-714.

Stolt, R.H., 1978, Migration by Fourier transform, Geophysics, v. 43, p. 23-48.

Taner, T. and Koehler, F., 1969, Velocity spectra - Digital computer derivation and applications of velocity functions, Geophysics, v. 34, p. 859-881.

Wiggins, R., Larner, K., and Wisecup, D., 1976, Residual statics analysis as a general linear inverse problem, Geophysics, v. 41, p. 922-938.

Yilmaz, O., 1979, Pre-stack partial migration, Ph. D. Dissertation, Dept. of Geophysics, Stanford University, Stanford CA, 94305, 115 p.

Schneider, W., 1971, Developments in seismic data processing and analysis (1968-1970), Geophysics, v. 36, p. 1043-1073.

Schultz, P.S. and Claerbout, J.F., Velocity estimation and downward continuation by wavefront synthesis, Geophysics, v. 43, p. 691-714.

Stolt, R.H., 1978, Migration by Fourier transform, Geophysics, v. 43, p. 23-48.

Taner, T. and Koehler, F., 1969, Velocity spectra - Digital computer derivation and applications of velocity functions, Geophysics, v. 34, p. 859-881.

Wiggins, R., Larner, K., and Wisecup, D., 1976, Residual statics analysis as a general linear inverse problem, Geophysics, v. 41, p. 922-938.

Yilmaz, O., 1979, Pre-stack partial migration, Ph. D. Dissertation, Dept. of Geophysics, Stanford University, Stanford CA, 94305, 115 p.

Appendix A
Subroutine for Solving Pentadiagonal Systems of Equations

```
      c    A pentadiagonal matrix solver.
c
c     solves the pentadiagonal system of equations for the unknown
c     vector t(k):
c
c       a(k)t(k-2) + b(k)t(k-1) + c(k)t(k) + d(k)t(k+1)
c
c          + e(k)t(k+2) = f(k),
c
c     where, k = 1,2,...,n
c
c     using the recursion relations:
c
c       t(k)   = q(k)t(k+2)   + r(k)t(k+1)   + s(k)
c
c       t(k-1) = q(k-1)t(k+1) + r(k-1)t(k)   + s(k-1)
c
c       t(k-2) = q(k-2)t(k)   + r(k-2)t(k-1) + s(k-2)
c
c     set ibc = 0 to solve the n equations exactly (i.e. user
c               specified boundary conditions).
c             = 1 for zero first and second derivative boundary
c               conditions.
c
      subroutine pent(a,b,c,d,e,f,n,t,q,r,s,ibc)
      dimension a(n),b(n),c(n),d(n),e(n),f(n),t(n),q(n),r(n),s(n)
c
c
c     left side boundary conditions
c
      if(ibc.eq.1) goto 3
c
c     solve the set of equations exactly.
c
      q(1) = -e(1)/c(1)
      r(1) = -d(1)/c(1)
      s(1) =  f(1)/c(1)
      den  = c(2) - d(1)*b(2)/c(1)
      q(2) = -e(2)/den
      r(2) = (e(1)*b(2)/c(1) - d(2))/den
      s(2) = (f(2) - f(1)*b(2)/c(1))/den
      goto 5
c
c     zero slope and zero second derivative b.c.'s
c     t(1) = t(2) = t(3)
c
c
    3 q(1) = 0.
```

Appendix A
Subroutine for Solving Pentadiagonal Systems of Equations

```
      c    A pentadiagonal matrix solver.
c
c     solves the pentadiagonal system of equations for the unknown
c     vector t(k):
c
c       a(k)t(k-2) + b(k)t(k-1) + c(k)t(k) + d(k)t(k+1)
c
c          + e(k)t(k+2) = f(k),
c
c     where, k = 1,2,...,n
c
c     using the recursion relations:
c
c       t(k)   = q(k)t(k+2)   + r(k)t(k+1)   + s(k)
c
c       t(k-1) = q(k-1)t(k+1) + r(k-1)t(k)   + s(k-1)
c
c       t(k-2) = q(k-2)t(k)   + r(k-2)t(k-1) + s(k-2)
c
c     set ibc = 0 to solve the n equations exactly (i.e. user
c               specified boundary conditions).
c             = 1 for zero first and second derivative boundary
c               conditions.
c
      subroutine pent(a,b,c,d,e,f,n,t,q,r,s,ibc)
      dimension a(n),b(n),c(n),d(n),e(n),f(n),t(n),q(n),r(n),s(n)
c
c
c     left side boundary conditions
c
      if(ibc.eq.1) goto 3
c
c     solve the set of equations exactly.
c
      q(1) = -e(1)/c(1)
      r(1) = -d(1)/c(1)
      s(1) =  f(1)/c(1)
      den  = c(2) - d(1)*b(2)/c(1)
      q(2) = -e(2)/den
      r(2) = (e(1)*b(2)/c(1) - d(2))/den
      s(2) = (f(2) - f(1)*b(2)/c(1))/den
      goto 5
c
c     zero slope and zero second derivative b.c.'s
c     t(1) = t(2) = t(3)
c
c
    3 q(1) = 0.
```

```
      r(1) = 1.
      s(1) = 0.
      q(2) = 0.
      r(2) = 1.
      s(2) = 0.
c
  5   n1 = n-1
      n2 = n-2
      n3 = n-3
      do 10 k=3,n
        den = a(k)*q(k-2) + r(k-1)*( a(k)*r(k-2) + b(k)) + c(k)
        q(k) = -e(k)/den
        r(k) = -( q(k-1)*(b(k) + a(k)*r(k-2)) + d(k) )/den
        s(k) = ( f(k) - ( s(k-1)*(b(k) + a(k)*r(k-2)) + a(k)*s(k-2)))/den
  10  continue
c
c     right side boundary conditions
c
      if(ibc.eq.1) goto 13
c
c     solve the equations exactly
c
      g1 = a(n)*q(n2) + c(n)
      g2 = a(n)*r(n2) + b(n)
      g3 = f(n) - a(n)*s(n2)
      h1 = a(n1)*r(n3)*q(n2) + b(n1)*q(n2) + d(n1)
      h2 = a(n1)*q(n3) + a(n1)*r(n3)*r(n2) + b(n1)*r(n2) + c(n1)
      h3 = f(n1) - a(n1)*r(n3)*s(n2) - a(n1)*s(n3) - b(n1)*s(n2)
      t(n) =  (g3 - g2*h3/h2)/(g1 - g2*h1/h2)
      t(n1) = (g3 - g1*h3/h1)/(g2 - g1*h2/h1)
      goto 15
c
c     zero slope and zero second derivative.
c     t(n2) = t(n1) = t(n)
c
  13  t(n) = s(n2)/(1.-(q(n2)+r(n2)))
      t(n1) = t(n)
c
  15  continue
      do 20 j=1,n2
        k = n1-j
  20  t(k) = q(k)*t(k+2) + r(k)*t(k+1) + s(k)
      return
      end
```