

1.5D internal multiple prediction in MATLAB

Kris Innanen

ABSTRACT

We present a 1.5D MATLAB implementation of the inverse scattering series internal multiple prediction algorithm developed by Weglein and collaborators in the 1990s. We discuss the transformation of the data from the space and time domain to those of wavenumber and pseudo-depth, and the subsequent prediction operation, and illustrate the procedure with a synthetic example. Our plan forward is to apply with the 1.5D algorithm the methods developed by Hernandez for the 1D algorithm (in this report), which involve stepping from synthetic, to laboratory, and finally to land data.

INTRODUCTION

In this paper we review the basic features of the inverse scattering series internal multiple attenuation algorithm as introduced to the geophysics literature in the 1990s (Araújo et al., 1994; Weglein et al., 1997, 2003), and demonstrate its application to 1.5D data using a MATLAB implementation. The purpose is to complement the 1D version of the algorithm studied by Hernandez and Innanen (2012) in a progression from synthetic, physical modelling and land data environments. The plan forward is to explore the field application of the current algorithm using a similarly staged approach. We begin with a review of the algorithm and its inputs, then describe how these are implemented in MATLAB. We demonstrate the use of the algorithm on a simple synthetic data set.

REVIEW: INTERNAL MULTIPLES AND THEIR PREDICTION

Transforming the input data to the pseudo-depth domain $b_1(k_g, k_s, z)$

The procedure for generating the input to the prediction algorithm is much the same as that by which a constant velocity Stolt migration (Stolt and Benson, 1986) is carried out. We begin with a data set measured over intervals in lateral source location x_s , lateral receiver location x_g , and time t . The data are Fourier transformed over all three of these coordinates:

$$d(x_g, x_s, t) \rightarrow D(k_g, k_s, \omega). \quad (1)$$

and a change of variables is made from ω to k_z :

$$D(k_g, k_s, \omega) \rightarrow D(k_g, k_s, k_z), \quad (2)$$

where $k_z = q_g + q_s$ and

$$q_g = \frac{\omega}{c_0} \sqrt{1 - \frac{k_g^2 c_0^2}{\omega^2}}, \quad q_s = \frac{\omega}{c_0} \sqrt{1 - \frac{k_s^2 c_0^2}{\omega^2}}. \quad (3)$$

The data are scaled by $-i2q_s$, forming

$$B_1(k_g, k_s, k_z) = (-i2q_s) D(k_g, k_s, k_z), \quad (4)$$

and B_1 is, finally, inverse Fourier transformed over k_z , appearing in the wavenumber-pseudodepth domain as

$$B_1(k_g, k_s, k_z) \rightarrow b_1(k_g, k_s, z). \quad (5)$$

The quantity $b_1(k_g, k_s, z)$ is the input to the prediction algorithm.

Internal multiple prediction in 2D

The prediction algorithm (Weglein et al., 1997) is then

$$\text{PRED}(k_g, k_s, \omega) = \left(\frac{1}{2\pi}\right)^2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} dk_1 dk_2 \times \Gamma(k_g, k_1, k_2, k_s | \epsilon), \quad (6)$$

where

$$\begin{aligned} \Gamma(k_g, k_1, k_2, k_s | \epsilon) = & \int_{-\infty}^{\infty} dz e^{i(q_g + q_1)z} b_1(k_g, k_1, z) \int_{-\infty}^{z-\epsilon} dz' e^{-i(q_1 + q_2)z'} b_1(k_1, k_2, z') \\ & \times \int_{z'+\epsilon}^{\infty} dz'' e^{i(q_2 + q_s)z''} b_1(k_2, k_s, z''), \end{aligned} \quad (7)$$

and the

$$q_X = \frac{\omega}{c_0} \sqrt{1 - \frac{k_X^2 c_0^2}{\omega^2}} \quad (8)$$

are vertical wave numbers associated with the various lateral wave numbers and the reference velocity c_0 .

Internal multiple prediction in 1D

The multidimensional prediction algorithm reduces to 1D—producing the form studied by Weglein and Matson (1998) and used by Hernandez and Innanen (2012)—via the replacement

$$k_g = k_s = 0, \quad (9)$$

in which case we obtain the frequency domain prediction

$$\text{PRED}(\omega) = \int_{-\infty}^{\infty} dz e^{i2\frac{\omega}{c_0}z} b_1(z) \int_{-\infty}^{z-\epsilon} dz' e^{-i2\frac{\omega}{c_0}z'} b_1(z') \int_{z'+\epsilon}^{\infty} dz'' e^{i2\frac{\omega}{c_0}z''} b_1(z''). \quad (10)$$

The difference in computational complexity between equations (6) and (10) could be fairly called enormous. The heart of the prediction is a set of three nested integrals over pseudodepth. The 1D algorithm requires one such set for each output frequency ω . The 2D algorithm requires many such sets: at each frequency the nested integration is repeated for every value of the integration variables k_1 and k_2 , over which the results are summed. Then, this is repeated for *every* output k_g, k_s pair.

Internal multiple prediction in 1.5D

However, if the data have offset but the Earth is nearly layered, a 1.5D version of the algorithm may be considered, in which

$$k_g = k_s, \quad (11)$$

but they are no longer necessarily equal to nil. Multiplying each instance of $b_1(k', k'', z)$ (and PRED) in equation (6) with $2\pi\delta(k' - k'')$, and comparing coefficients of the delta functions after integration, we obtain the 1.5D algorithm

$$\begin{aligned} \text{PRED}(k_g, \omega) = & \int_{-\infty}^{\infty} dz e^{ik_z z} b_1(k_g, z) \int_{-\infty}^{z-\epsilon} dz' e^{-ik_z z'} b_1(k_g, z') \\ & \times \int_{z'+\epsilon}^{\infty} dz'' e^{ik_z z''} b_1(k_g, z'') \end{aligned} \quad (12)$$

where $k_z = 2q_g$. The computation problem has been dramatically reduced, with the equivalent of a single 1D prediction for every output k_g . We implement this version of the algorithm in MATLAB. We describe the form this implementation takes in the next section.

1.5D INTERNAL MULTIPLE PREDICTION IN MATLAB

Input data

We assume the availability of a single split-spread shot record of input data with direct arrivals muted (halt muting at offsets where the direct wave begins to interfere with reflections). Deconvolution and deghosting are a useful preprocessing step; if the internal multiples are resolvable in the data without these steps, they may be avoided (we will not deconvolve our synthetic data later in this paper), though this will lead to a more involved subtraction problem subsequently. The data are stored in a matrix *data* in MATLAB of size $[N, M] = \text{size}(\text{data})$.

Making the pseudo-depth domain input $b_1(k_g, z)$

Step 1: 2D FFT

Beginning with our input data, we follow the set of procedures described in equations (1)–(5). First, we Fourier transform the data: the 2D FFT algorithm internal to MATLAB maps the data from a regular (x_g, t) grid to a regular (k_g, ω) grid.

Step 2: choosing a regular k_z vector

We define a regular output grid on (k_g, k_z) . The output wavenumber variable k_z is conjugate to pseudo depth $z = c_0 t / 2$. Hence, we can choose an optimum grid for the vector $\{k_{z_0}, k_{z_1}, \dots, k_{z_N}\}$ by starting with our input t vector:

$$t = dt * ((1 : N) - 1); \quad (13)$$

with dt the sampling interval. We can thereafter map to pseudo-depth

$$z = 0.5 * c0 * t, \quad (14)$$

where $c0$ is the reference medium P-wave velocity (for internal multiple prediction any plausible value “reasonably” close to the velocity nearby the sources and receivers can be used), by determining the associated $dz = z(2) - z(1)$, and then defining

$$\begin{aligned} k_z &= -N/2 : N/2 - 1; \\ k_z &= k_z / (N * dz); \end{aligned} \quad (15)$$

as the regularly-sampled output depth wavenumber.

Resampling

The data are available on a regular (k_g, ω) grid after the 2D FFT. Because the relationship between (k_g, ω) and (k_g, k_z) is nonlinear, a direct change of variables from this regular grid would lead to a data set on an irregular (k_g, k_z) grid. To get around this, we compute a regular desired (k_g, k_z) grid (with k_z chosen as described above), determine the irregular ω values needed to accommodate it, and perform a linear interpolation of the data onto these irregularly spaced values.

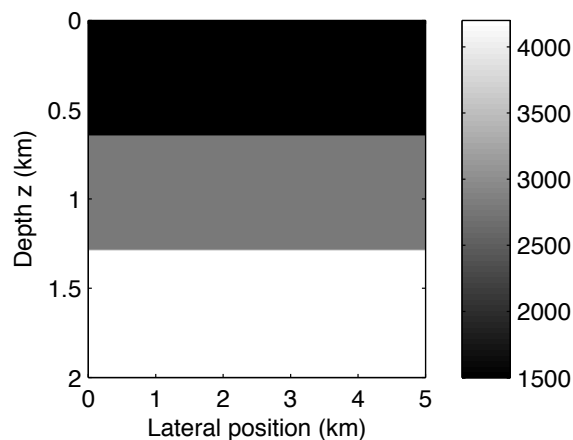


FIG. 1. Layered two interface model used to generate synthetic data and test the 1.5D internal multiple prediction algorithm.

Predicting internal multiples

The 1.5D prediction comprises three nested loops. (Vectorizing one or more of these, which would lead to considerable computational savings, is a matter of ongoing study.) Table 1 outlines the basic form in pseudocode. The loops are over lateral wavenumber k_x (ii), temporal frequency w (jj), and depth z (kk). In all three cases the integration beginning and ending points (i.e., kxB , kxE , wB , wE , zB , and zE) can sometimes be reduced to speed up computation. For instance, the inner integral is searching for even combinations in pseudodepth: hence, if no contributing events fall outside a given interval, zB and zE can be adjusted to reflect this. Also, the evanescent boundary and the bandlimited nature of the original data will provide limits for k_x and ω , allowing kxB , kxE , wB , wE to be adjusted.

```

For  $ii = kxB : kxE$ 
 $F = kx(ii)^2 c_0^2 / w.^2$ 
 $q_g = (w/c_0) .* \text{sqrt}(1 - F)$ 
    For  $jj = wB : wE$ 
 $A1 = i * 2 * qg(jj) * z;$ 
 $A2 = -i * 2 * qg(jj) * z;$ 
 $I1 = b1(:, ii) .* \text{exp}(A1);$ 
 $I2 = b1(:, ii) .* \text{exp}(A2);$ 
        For  $kk = zB : zE$ 
 $S = \text{sum}(I1(kk + \epsilon : zE));$ 
 $P = P + I2(kk) * S^2;$ 
        End
    End
 $P = P * dz;$ 
End

```

Table 1. Prediction algorithm in pseudocode.

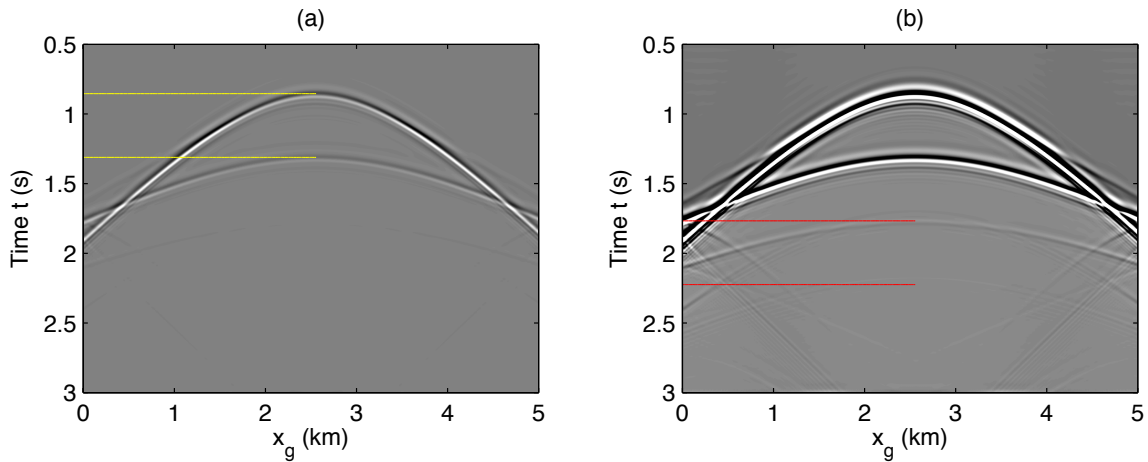


FIG. 2. Synthetic data calculated using the synthetic model in Figure 1. The CREWES code *afd_shotrec.m* was used to create the data. (a) Primary zero offset travel times are indicated in red. (b) Multiple zero offset travel times are indicated in yellow.

SYNTHETIC EXAMPLE

Input data

In Figure 1 we illustrate a simple two interface model which we will use to test the internal multiple predictions. In Figures 2a-b a single shot record of data is illustrated. The data are created using the CREWES acoustic finite difference function *afd_shotrec.m*, with all four boundaries set as “absorbing” to suppress the creation of free-surface multiples.

In Figure 2a we pay particular attention to the two primaries, whose zero offset travel times are indicated in yellow. In Figure 2b we pay attention rather to the two internal multiples that (while dim) are visible in the data. The zero offset travel times of these events are indicated in red. Our objective is to use the primaries as subevents to predict

these two multiples at all offsets.

Making $b_1(k_g, z)$

In Figure 3 we illustrate the construction of the core input to the prediction algorithm, $b_1(k_g, z)$. Note it is constructed for positive k_g values only—later through conjugate symmetry the negative wavenumbers are filled. Although difficult to interpret, at low k_g the arrival times of the two primaries are visible.

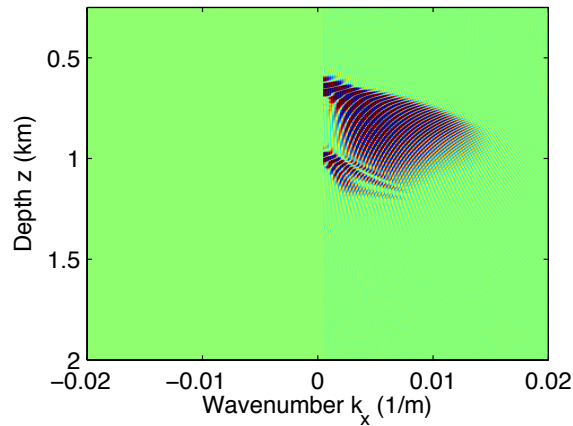


FIG. 3. The algorithm input $b_1(k_g, z)$ is generated using the input data and the single reference velocity c_0 .

Predicting internal multiples

We finally input the constructed b_1 matrix into the prediction algorithm. The results, after a 2D inverse Fourier transform, are displayed in Figure 4. In Figure 4a the prediction is displayed to match with a clipped version of the input data in 4b. The zero offset travel times and moveout patterns of the internal multiples are captured in the prediction. A range of artifacts and edge effects are visible also, which are matters of ongoing consideration.

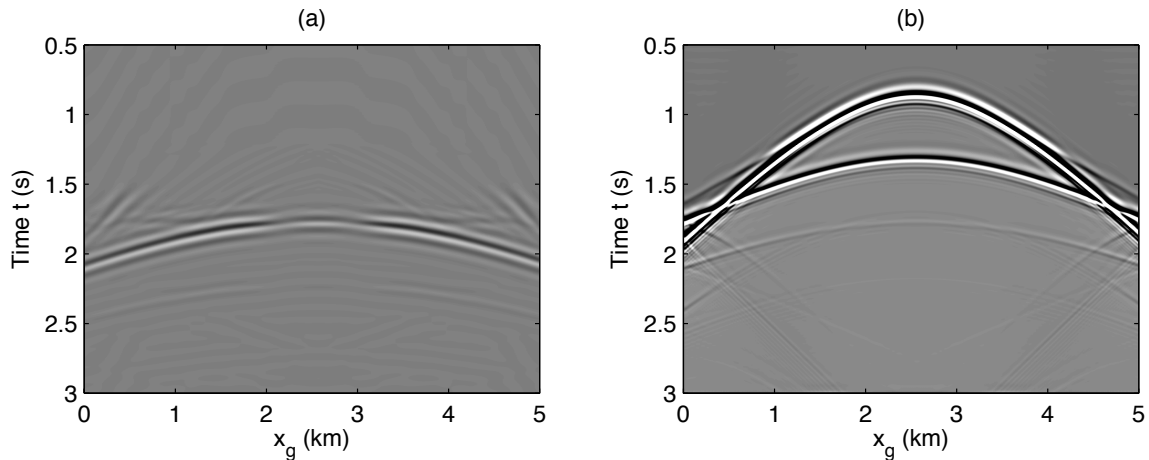


FIG. 4. The output of the 1.5D internal multiple prediction. (a) The prediction, in which two multiples are predicted. (b) The original data with both primaries and internal multiples.

CONCLUSIONS

We implement a 1.5D version of the inverse scattering series internal multiple algorithm developed by Weglein and collaborators in the 1990s in MATLAB. Our plan forward is to apply with the 1.5D algorithm the methods developed by Hernandez for the 1D algorithm (this report), which involve stepping from synthetic, to laboratory, and finally to land data.

Areas that require further study include examining the use of tapering for aperture effects, moving from a linear to a sinc interpolation scheme during the construction of b_1 , and a survey of the response of the algorithm to missing traces and irregular data.

ACKNOWLEDGMENTS

This work was funded by CREWES. CREWES sponsors and research personnel are gratefully acknowledged.

REFERENCES

- Araújo, F. V., Weglein, A. B., Carvalho, P. M., and Stolt, R. H., 1994, Inverse scattering series for multiple attenuation: an example with surface and internal multiples: *SEG Expanded Abstracts*, **13**, 1039–1041.
- Hernandez, M., and Innanen, K. A., 2012, Internal multiple prediction: from synthetic to lab to land data: *CREWES Annual Report-this report*, **24**.
- Stolt, R. H., and Benson, A. K., 1986, *Seismic Migration: Theory and Practice*, vol. 5 of *Seismic Exploration*: Geophysical Press.
- Weglein, A. B., Araújo, F. V., Carvalho, P. M., Stolt, R. H., Matson, K. H., Coates, R. T., Corrigan, D., Foster, D. J., Shaw, S. A., and Zhang, H., 2003, Inverse scattering series and seismic exploration: *Inverse Problems*, , No. 19, R27–R83.
- Weglein, A. B., Gasparotto, F. A., Carvalho, P. M., and Stolt, R. H., 1997, An inverse-scattering series method for attenuating multiples in seismic reflection data: *Geophysics*, **62**, No. 6, 1975–1989.
- Weglein, A. B., and Matson, K. H., 1998, Inverse scattering internal multiple attenuation: An analytic example and subevent interpretation, *in* *SPIE Conference on Mathematical Methods in Geophysical Imaging*, 108–117.