Accounting for attenuation in the downward-generator adaptive subtraction domain

Scott Keating, Andrew Iverson and Kris Innanen

ABSTRACT

Inverse-scattering based internal multiple prediction strategies suffer from inaccuracies in the amplitudes of predicted multiples. A notable source of these errors is the mistreatment of transmission losses by the conventional prediction algorithm. Interface-related transmission losses have been investigated previously, but artifacts from wave attenuation impair the predictions in much the same way, and potentially to a larger extent. Compensating for these errors with a conventional adaptive subtraction is problematic: the attenuation-related errors do not vary consistently as a function of the prediction dimensions. Here, we illustrate that attenuation errors are highly predictable in downward-generator space, and that adaptive subtraction in the downward-generator-prediction space is capable of compensating for attenuation errors.

INTRODUCTION

Seismic processing and inversion approaches often operate under the assumption of primaries-only data. This can be problematic when there are significant multiples in the measured data, and often requires their removal. Inverse scattering series multiple prediction (ISSIMP) is a data driven technique used to predict internal multiples in data (Weglein et al., 1997), which can be a useful step in removing them. This approach does not require subsurface information, and allows for prediction of classes of multiples that cannot be easily removed with other approaches.

Because the multiples predicted through ISSIMP will differ from those measured in both amplitude and wavelet, it is necessary to modify the predicted multiples before removal from the data. This is usually achieved through adaptive subtraction, wherein a matching filter is designed to better fit the predicted multiples to the observed data and these filtered predictions are subtracted from the data. The key challenge in adaptive subtraction is the design of this matching filter. Because the data contains both primaries and multiples, a filter achieving perfect matching is undesirable, as this will necessitate the removal of primaries as well as the multiples. Fortunately, a sufficiently conservative filter will be incapable of perfect matching, as the primaries should bear no consistent relation to the prediction; no one filter will be able to match all the primaries. On the other hand, if the amplitude and wavelet prediction errors are consistent for all multiples, a single filter should be capable of matching all of them. If this is the case, the filter which achieves multiple removal can usually be calculated without knowledge of the true multiples; it is the filter which achieves the best possible matching to the data.

One of the key assumptions making adaptive subtraction possible is the consistency of the amplitude and wavelet prediction errors. If this assumption fails, then the key distinction between multiple energy and primary energy is taken away, making effective subtraction much more difficult. Unfortunately, there are known errors in the prediction process, and some of these are not consistent between different sets of multiples. In particular, transmission through and attenuation en route to the downward reflector associated with the multiple lead to errors that are not consistent between different downward reflectors.

Iverson et al. (2018) proposed an alternate form for internal multiple prediction. Rather than defining the prediction as a function solely of the data coordinates, this strategy defined the prediction as a function also of the data coordinates of the downward-generator, the earliest-time event used in the prediction of a given multiple. While this approach does not result in a different multiple prediction, it does retain information about which predictions are associated with which downward-generators. The crucial importance of this prediction domain is that it greatly improves our ability to anticipate prediction errors. While errors associated with transmission and attenuation losses can be highly variable as a function of prediction time, they are much closer to stationary when defined with respect to both prediction time and downward-generator time. This makes the downward-generator domain highly appealing for adaptive subtraction, especially in data where attenuation or transmission losses are substantial.

In this report, we will outline in detail an adaptive subtraction strategy for the downward-generator (DG) domain, and highlight the improvements it offers when considering data with substantial transmission or attenuation losses.

THEORY

In adaptive subtraction, the corrected prediction to be removed from the data, P, is given by

$$\Theta f = P, \tag{1}$$

where f is a filter, and Θ is a matrix representing convolution with the original prediction. One of the simplest choices for f is the filter which solves the optimization problem

$$\underset{f}{min} ||\Theta f - D||_2^2, \tag{2}$$

where D are the measured data. The minimum is achieved where the derivative with respect to f is equal to zero, or equivalently

$$\Theta^T \Theta f = \Theta^T D. \tag{3}$$

Filters optimal with respect to other metrics can be chosen to emphasize different parts of the data. For instance, a filter minimizing a hybrid L_1/L_2 norm can be calculated by solving the problem

$$\underset{f}{min} ||G\Theta f - GD||_2^2, \tag{4}$$

where G is a weighting matrix, iteratively recalculated to result in an approximate L_1 weighting. The filter minimizing this function is

$$\Theta^T G^T G \Theta f = \Theta^T G^T D. \tag{5}$$

In a conventional multiple-removal adaptive subtraction there are two major decisions to be made in the filter design: which objective function the filter should minimize, and which prediction elements are weighted by the filter. The second question is addressed through the choice of structure in Θ. For instance, in determining the corrected prediction for the n^{th} data point, we might choose a filter which weights terms $n-m$ to $n+m$ in the original prediction. In this case, we would define Θ as

$$\Theta = \begin{bmatrix} \theta_{m+1} & \dots & \theta_2 & \theta_1 & 0 & \dots & 0 \\ \theta_{m+2} & \dots & \theta_3 & \theta_2 & \theta_1 & \dots & 0 \\ \vdots & \dots & \vdots & \vdots & \vdots & \dots & \vdots \\ \theta_{(m+1)+m} & \dots & \theta_{2+m} & \theta_{1+m} & \theta_m & \dots & \theta_1 \\ \vdots & \dots & \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & \dots & \theta_N & \theta_{N-1} & \theta_{N-2} & \dots & \theta_{(N-1)-m} \\ 0 & \dots & 0 & \theta_N & \theta_{N-1} & \dots & \theta_{N-m} \end{bmatrix}, \tag{6}$$

where θ_i corresponds to the i^{th} element in the multiple prediction, and N is the total number of elements in the data. This form is appropriate when the predictions are of the same dimensionality as the data. The number of elements in f (or equivalently the number of columns in Θ) is the key parameter controlling the flexibility of the filter in matching the data. Long filters can better change the prediction to match the data, but this includes undesirable matching of primaries. Ideally, the filter will be long enough to match predictions to multiples, but not so long that matching to primaries can occur as well. As mentioned in the introduction, if the differences between the predictions and multiples are not consistent, it may be impossible to satisfy both these criteria.

To help address this concern, we consider adaptive subtraction with DG domain multiple predictions. This means that predictions are not just functions of the prediction location in data-space, but also of DG location in data-space. To highlight the advantages of this domain in dealing with amplitude errors, we investigate a conceptual example.

Consider a first-order multiple M_{xyz} in a layered medium which experiences up-going reflections at interfaces x and z, and a down-going reflection at interface y. Let us assume that for a given primary reflection, P_n, the amplitude is given by the product of reflectivity, R_n, interface transmission losses to that depth, and attenuation transmission losses to that depth (which may be frequency dependent), $\prod_{j=1}^{n} \alpha_j$:

$$A_n = R_n * \left(\prod_{i=1}^{n-1} (1 - R_i^2) \right) \left(\prod_{j=1}^{n} \alpha_j^2 \right), \tag{7}$$

where $\prod_{i=x}^{y} T_i = 1$ for $x > y$. The amplitude of an internal multiple can be similarly constructed as

$$A_{mnp} = (R_m R_n R_p) * \left(\prod_{i_1=1}^{m-1} (1 - R_{i_1}) \prod_{i_2=m-1}^{n+1} (1 + R_{i_2}) \prod_{i_3=n+1}^{p-1} (1 - R_{i_3}) \prod_{i_4=p-1}^{1} (1 + R_{i_4}) \right) * \left(\prod_{j_1=1}^{m} \alpha_{j_1} \prod_{j_2=m}^{n+1} \alpha_{j_2} \prod_{j_3=n+1}^{p} \alpha_{j_3} \prod_{j_4=p}^{1} \alpha_{j_4} \right). \tag{8}$$

Equivalently,

$$A_{mnp} = (R_m R_n R_p) * \left(\prod_{i_1=1}^{n} (1 - R_{i_1}^2) \prod_{i_2=n+1}^{\min(m,p)-1} (1 - R_{i_2}^2)^2 \prod_{i_3=\min(m,p)}^{\max(m,p)} (1 - R_{i_3}^2) \right) * \left(\prod_{j_1=1}^{n} \alpha_{j_1}^2 \prod_{j_2=n+1}^{\min(m,p)} \alpha_{j_2}^4 \prod_{j_3=\min(m,p)}^{\max(m,p)} \alpha_{j_3}^2 \right) \tag{9}$$

ISSIMP predicts the multiple M_{xyz} through combinations of the primaries P_x, P_y, and P_z. The amplitude of the prediction is simply the product of the amplitude of the individual primaries (Weglein et al., 1997):

$$A_{xyz}^* = P_x P_y P_z. \tag{10}$$

Comparison of equation 10 with equations 8 and 7 shows that there is an error factor associated with transmission losses in the prediction, given by

$$\epsilon = (1 - R_y^2) \prod_{i=1}^{y-1} (1 - R_i^2)^2 \prod_{j=1}^{y} \alpha_j^4. \tag{11}$$

This term has the potential to be quite significant; it involves both the fourth power of α and $(1 - R_i^2)^2$. Even for values of α relatively close to one, and R_i small, this exponent can cause significant amplitude differences between the prediction and the measured signal. Significantly, this error is independent of the multiple's location in data-space, it is a function only of the location of the downward reflection multiple generator P_y in data-space. This means that the correction needed for the prediction's amplitude is not consistent for all multiples, nor does it vary predictably as a function of data-space. For this reason, it may be advantageous to consider multiple predictions from ISSIMP as functions of both the prediction location in data-space and the DG location in data-space. Iverson et al. (2018) formulated such a prediction domain.

Using the approach of Iverson et al. (2018), we can define a prediction in a prediction location - DG location space. In this domain, the errors ϵ associated with transmission effects are highly predictable. This leads us to expect that adaptive subtraction applied to these predictions should be much better able to remove multiple energy without overfitting the data. A naive prediction in this domain will have a very large dimensionality: each possible combination of prediction location and DG location in data-space can be represented, resulting in a prediction with a size equal to the square of the data. This is generally much too large to store in real data applications. We can instead choose to bin the prediction in the DG dimension. The exact extent of the binning will control the trade-off between memory demands and prediction-error consistency, but generally this allows for one of the dimensions to be reduced to a number on the order of the number of large reflectors in the data.

When the binned, DG domain multiple predictions have been made, an adaptive subtraction is needed to remove the predicted multiples. A DG-domain filter can be calculated by solving equation 2 after replacing Θ with matrix Ξ to reflect the dimensionality of the prediction:

$$\underset{f}{min}||G\Xi f - GD||_2^2, . \tag{12}$$

For DG-domain predictions, Ξ is given by

$$\Xi = \begin{bmatrix} \Theta_1 & \Theta_2 & \dots & \Theta_{N_2} \end{bmatrix}, \tag{13}$$

where N_2 is the number of bins in the DG dimension,

$$\Theta_x = \begin{bmatrix} \theta_{m+1,x} & \dots & \theta_{2,x} & \theta_{1,x} & 0 & \dots & 0 \\ \theta_{m+2,x} & \dots & \theta_{3,x} & \theta_{2,x} & \theta_{1,x} & \dots & 0 \\ \vdots & \dots & \vdots & \vdots & \vdots & \dots & \vdots \\ \theta_{(m+1)+m,x} & \dots & \theta_{2+m,x} & \theta_{1+m,x} & \theta_{m,x} & \dots & \theta_{1,x} \\ \vdots & \dots & \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & \dots & \theta_{N,x} & \theta_{N-1,x} & \theta_{N-2,x} & \dots & \theta_{(N-1)-m,x} \\ 0 & \dots & 0 & \theta_{N,x} & \theta_{N-1,x} & \dots & \theta_{N-m,x} \end{bmatrix}, \tag{14}$$

and $\theta_{i,x}$ is the i^{th} element of the multiple prediction in prediction space, and the x^{th} element in DG space.

A simple approach for filter design is to allow each element of f to vary independently, as was the case in the conventional AS filter. In this case, the filter can be calculated by replacing Θ in equation 5 with Ξ:

$$\Xi^T G^T G \Xi f = \Xi^T G^T D. \tag{15}$$

There are several drawbacks to this approach. Firstly, f has dimensionality equal to the number of columns in Ξ, generally the filter length in prediction space, $2m + 1$, by the filter length in DG space, N_2. This is problematic because while this number of filter elements is necessary to apply adaptive subtraction in our chosen domain, it results in $N_2(2m + 1)$ degrees of freedom in the filter design. This can lead to dramatic over-fitting and unwanted subtraction of primaries for even relatively modest values of m and N_2. A second concern is that this approach does not put any constraints on the structure of f. We expect that f should apply a wavelet correction through use of variation in prediction-domain coefficients, independent of generator time, and a transmission effect correction that corrects for the errors expected in equation 11, which increases with DG time, but is independent of prediction time. A solution of equation 15 does not require any of these behaviours, and, when combined with excessive filter dimensionality, may instead tend to unwanted primary matching.

These two problems are, in fact, highly connected. A filter which varies only overall amplitude in DG space, and applies the same wavelet correction at each generator is not a fully $N_2(2m + 1)$-dimensional filter. To illustrate, consider the filter element at the i^{th} location in prediction space, and the j^{th} location in DG space, $f_{i,j}$. If each element of this filter is allowed to vary independently, we have

$$f_{i,j} = w(i, j), \tag{16}$$

where w is an arbitrary function. Because there are no constraints on w, the filter design space is fully $N_2(2m+1)$-dimensional. If, instead, f is composed of g, which corrects for wavelet effects, and h, which corrects for transmission loss prediction errors, we have

$$f_{i,j} = g(i)h(j). \tag{17}$$

In this case, g is a function only of prediction location i because we assume no dependence of the wavelet correction on DG location, and h, correcting for the amplitude errors in equation 11, is a function of only DG location j. This filter has the twin advantages of forcing the type of correction that we have motivated, and reducing the dimensionality of the filter to $N_2 + 2m + 1$. To further enforce the desired features of the filter, we could force h to only increase as DG grow deeper, as the associated error term becomes smaller and smaller in this direction (equation 11). One appropriate choice would be

$$h(j) = \left(1 + e^{k(j)}\right) \prod_{n=1}^{j-1} h(n), \tag{18}$$

where k is allowed to vary freely as a function of j.

Having chosen to adopt a filter design consistent with equations 17 and 18, we must also formulate a way of solving equation 12. The solution to equation 15 will not, in general, be consistent with the filter structure we have selected. In fact, no direct solution for the ideal filter seems available. Instead, we choose to solve equation 12 iteratively, through Newton's method. At each iteration, we update the vector $p = \begin{bmatrix} g \\ k \end{bmatrix}$ by δp, which is calculated through

$$\delta p = -\frac{\partial^2 \phi}{\partial p^2}\frac{\partial \phi}{\partial p} = -\begin{bmatrix} \frac{\partial^2 \phi}{\partial g^2} & \frac{\partial^2 \phi}{\partial g \partial h} \\ \\ \frac{\partial^2 \phi}{\partial h \partial g} & \frac{\partial^2 \phi}{\partial g^2} \end{bmatrix} \begin{bmatrix} \frac{\partial \phi}{\partial g} \\ \\ \frac{\partial \phi}{\partial h}, \end{bmatrix} \tag{19}$$

where

$$\phi(p) = ||G\Xi f(p) - GD||_2^2. \tag{20}$$

These derivatives can be defined in terms of $\frac{d\phi}{df}$ and $\frac{d^2\phi}{df^2}$ by

$$\frac{\partial \phi}{\partial p} = \frac{d\phi}{df}\frac{\partial f}{\partial p} \tag{21}$$

and

$$\frac{\partial^2 \phi}{\partial p^2} = \frac{\partial}{\partial p}\left(\frac{\partial \phi}{\partial p}\right) = \frac{df}{dp}\frac{d^2\phi}{df^2}\frac{df}{dp} + \frac{d\phi}{df}\frac{d^2 f}{dp^2}. \tag{22}$$

The derivative of f with respect to g is

$$\frac{df_{i,j}}{dg_i} = h_j, \tag{23}$$

and with respect to k is,

$$\frac{df_{i,j}}{dk_q} = \begin{cases} g_i \frac{e^{k_q}}{1+e^{k_q}} \prod_{n=1}^{j} h(n) & \text{if } q \leq j \\ 0 & \text{if } q > j. \end{cases} \tag{24}$$

The second derivatives are given by

$$\frac{d^2 f_{i,j}}{dg_i^2} = 0, \tag{25}$$

$$\frac{d^2 f_{i,j}}{dg_i dk_q} = \begin{cases} \frac{e^{k_q}}{1+e^{k_q}} \prod_{n=1}^{j} h(n) & \text{if } q \leq j \\ 0 & \text{if } q > j, \end{cases} \tag{26}$$

and

$$\frac{d^2 f_{i,j}}{dk_q dk_r} = \begin{cases} g_i \frac{e^{k_q+k_r}}{\left(1+e^{k_q}\right)\left(1+e^{k_r}\right)} \prod_{n=1}^{j} h(n) & \text{if } q, r \leq j,\ r \neq q \\ \\ g_i \frac{e^{k_q}}{1+e^{k_q}} \prod_{n=1}^{j} h(n) & \text{if } q = r \leq j \\ \\ 0 & \text{if } q > j \text{ or } r > j. \end{cases} \tag{27}$$

With this approach to filter design, we should be able to design a filter with a large number of elements, allowing for application to the DG domain predictions, but relatively few degrees of freedom, enforcing our assumptions and lessening the chance of primary subtraction. We investigate several 1D synthetic examples to test this approach.

NUMERICAL EXAMPLES

To highlight important features of this approach, we apply it here to a number of 1D examples. For these tests, we generate a velocity model, and use this to calculate travel times and reflection and transmission coefficients associated with each layer under the assumption of a constant density acoustic medium. For the first examples, we consider an extremely simplified model for attenuation. Specifically, we assume that there is a single dominant frequency f^* for the wavelet, and that the data contain negligible energy away from this frequency. In this case, the fraction of energy lost in transmission through a single layer can be approximated as a constant scale factor, and dispersion can be neglected. This simplifies the forward modelling, and avoids complications associated with frequency dependence.

For the numerical tests, we consider the 1-D model shown in figure 2. This model contains several strong contrasts, which should generate significant internal multiples, as well as one layer with substantial attenuation. To generate data from this model, we use an analytical, convolutional approach. Equation 7 is used to calculate the amplitude of the primaries, and first and second order internal multiple amplitudes are calculated using a similar approach. Travel-times for each primary and multiple considered were calculated using the velocity model. Once the travel-time and amplitude of each multiple was determined, each event was convolved with a 40 Hz Ricker wavelet. No surface multiples were included. The resulting trace is shown in figure 1, with the contribution of the multiples shown in orange. Gain is applied to this figure and the others we show in this section in order to show the smaller, later multiples.

Using this trace as input, the DG-domain internal multiple prediction output is shown in figure 3. This prediction is a function of both DG time and prediction time. While this

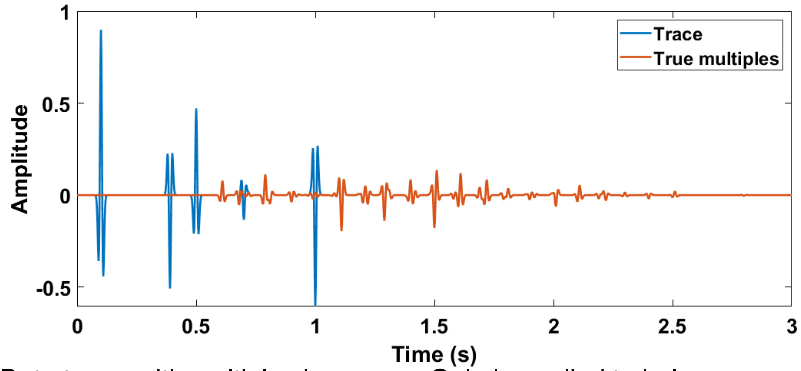

FIG. 1. Data trace, with multiples in orange. Gain is applied to help see smaller events.

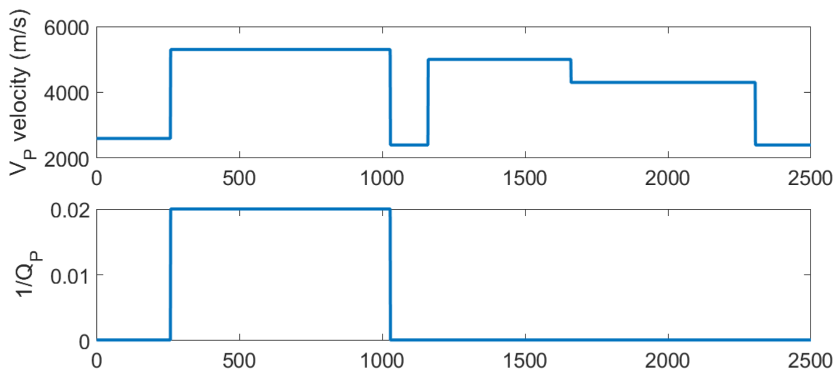

FIG. 2. 1D model used in synthetic test. Strong contrasts are present to create large internal multiples

result represents the most complete representation in the output domain, this prediction has a N^2 elements, where N is the number of elements in the data. A more manageable representation of the prediction is shown in figure 4, where the DG-domain prediction has been summed within a small number of DG time bins. This preserves DG locations to within the bin size, but reduces the size of the prediction to mN, where m is the number of bins. In practise, this summation can be performed during the prediction. In this example, we dvide DG-time into 20 bins and stack within each, discarding bins with negligible energy. The conventional multiple prediction can be obtained by choosing $m = 1$; in this case the prediction loses all DG-location information, but reduces the prediction to the size of the data. A comparison of the scaled conventional prediction and the data trace is shown in figure 5.

A conventional adaptive subtraction designs a filter independent of the DG-time of the prediction to match the prediction to the measured trace. Conceptually, we can interpret this filter in different ways. Usually, the filter is considered to be have F_1 elements, where F_1 is number of time lags considered, and is applied to the conventional prediction (figure 5). We can also consider this filter as applying to the DG-domain prediction however (figure 4). In this case, a conventional filter is of length F_1 in prediction-time, and length m in DG-bins. This filter has many more elements, but has coefficients which vary only as a function of lag time; only F_1 degrees of freedom exist and the m coefficients which exist at each lag will have the same value, simply serving to perform summation in this dimension. These two filters are equivalent, but the second is useful for conceptualizing the connection between the conventional and the DG-domain prediction.

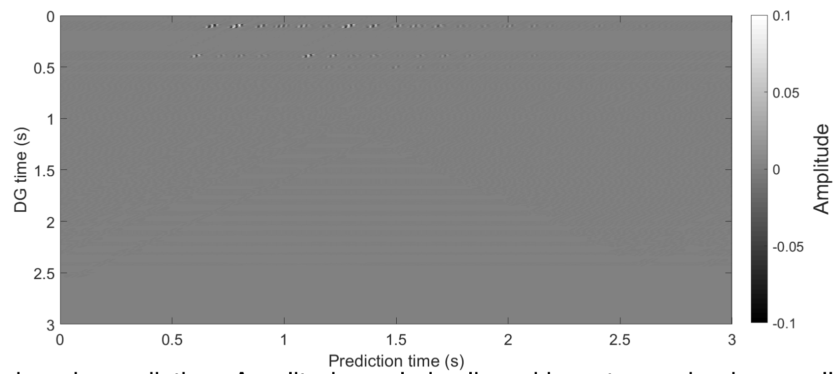

FIG. 3. DG domain prediction. Amplitude scale is clipped here to emphasize smaller events.

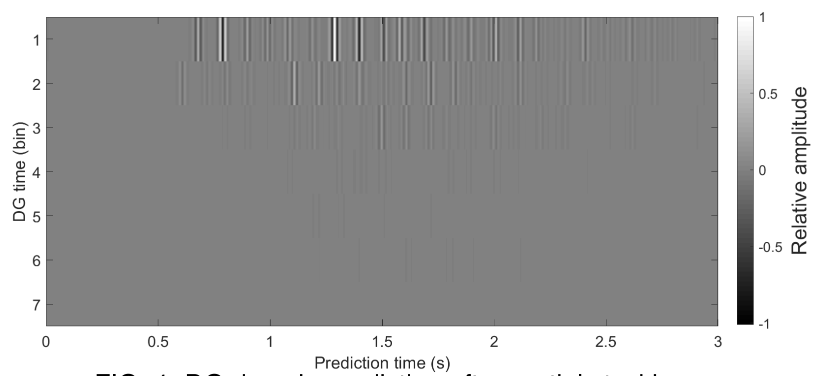

FIG. 4. DG domain prediction after partial stacking.

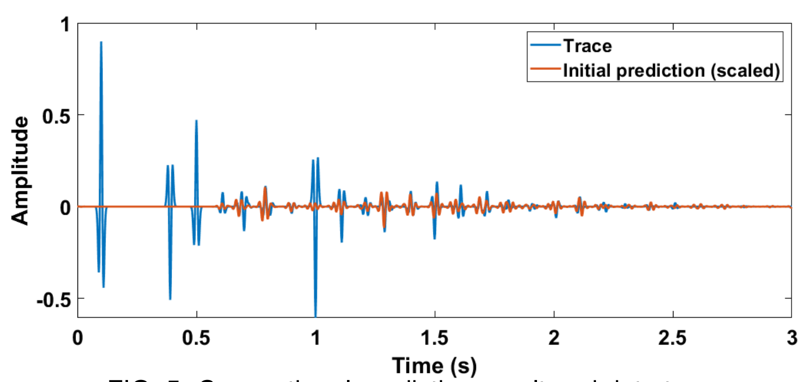

FIG. 5. Conventional prediction result and data trace.

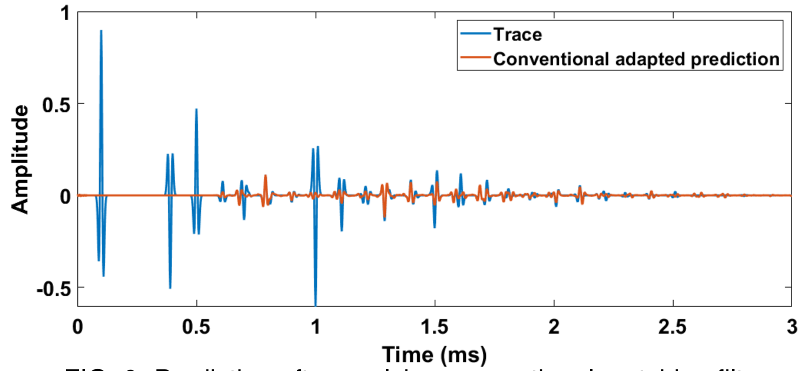

FIG. 6. Prediction after applying conventional matching filter.

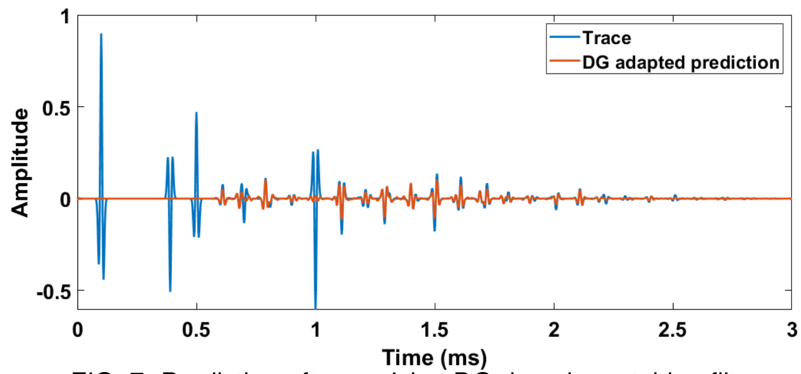

FIG. 7. Prediction after applying DG domain matching filter.

The modified prediction calculated using a 22 time-lag conventional adaptive subtraction filter is shown in figure 6. This prediction tends to align events with real multiples, but in several cases, the amplitude of the predicted events is severely under-estimated, particularly at 0.6 s, 1.1 s, 1.5 s, and 1.6 s. One major cause of these under-predictions is the failure of the prediction algorithm to correctly account for transmission-associated losses; the conventional filter required to match the data is different for multiples with different levels of prediction error. To better approach this problem, we next consider a DG-domain adaptive subtraction.

Using a DG-domain adaptive subtraction filter with 15 time-lags and 7 DG times, we create the modified prediction shown in figure 7. The filter used here has the same number of degrees of freedom as the conventional filter considered, but the resulting prediction is substantially improved, especially where under-predictions were present in the conventional result. As a tool to investigate the effect of the extra prediction dimension, we can sum the DG prediction over the DG-time dimension after multiplying by the filter coefficients for these times. This gives a corrected, data-domain prediction which can be compared to the original data-domain prediction to see the effect of the DG-domain information in correcting the amplitude errors. This corrected prediction is shown in figure 8. This prediction does substantially better at matching the amplitudes of the events at 0.6 s, 1.1 s, and 1.6 s.

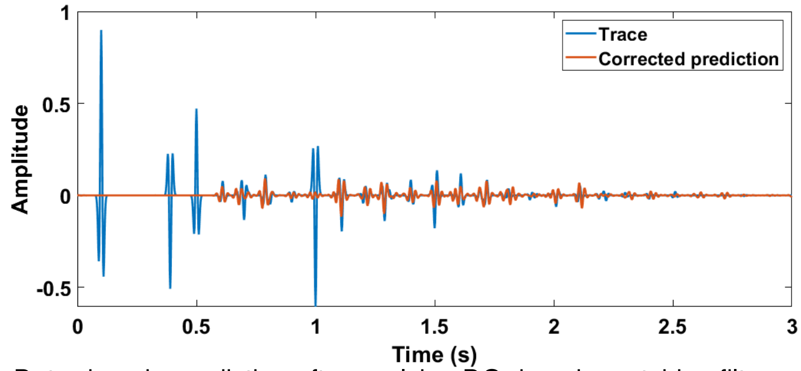

FIG. 8. Data-domain prediction after applying DG domain matching filter corrections.

DISCUSSION

In the numerical results here, we have used a model of attenuation that is frequency independent, and dispersionless. As a result, the attenuation loss errors can be treated in a way very similar to the interface transmission loss errors. In general, however, this simplification, which hinges on considering only a single frequency, will be quite poor; seismic data usually have a substantial frequency bandwidth. The frequency-dependence of attenuation, and especially the dispersion present in real media may have significant effects on prediction errors that aren't represented here. For this reason, the behaviour of this subtraction approach should be investigated when more realistic attenuation errors are present in the data. This topic is a subject for ongoing work.

CONCLUSIONS

In this report, we propose an adaptive subtraction strategy for the removal of internal multiples, using a downward-generator domain ISSIMP. Conventional ISSIMP is prone to prediction errors arising from transmission and attenuation losses in the subsurface. These errors are difficult to handle with conventional adaptive subtraction, but are highly consistent in the DG-domain. The approach we propose here allows for adaptive subtraction to effectively compensate for these prediction errors without requiring additional computation in the prediction. Simple 1-D numerical tests suggest that the uplift associated with this subtraction approach may be substantial.

ACKNOWLEDGEMENTS

The authors thank the sponsors of CREWES for continued support. This work was funded by CREWES industrial sponsors and NSERC (Natural Science and Engineering Research Council of Canada) through the grant CRDPJ 461179-13. Scott Keating was also supported by the Earl D. and Reba C. Griffin Memorial Scholarship.

REFERENCES

Iverson, A., Keating, S., Innanen, K., and Trad, D., 2018, Internal multiple prediction with higher order terms and a new subtraction domain: CREWES Annual Report, **30**.

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.