TU Delft

Delft University of Technology

Implementation of the Marchenko multiple elimination algorithm

Thorbecke, Jan; Zhang, Lele; Wapenaar, Kees; Slob, Evert

DOI
10.1190/geo2020-0196.1

Publication date
2021

Document Version
Accepted author manuscript

Published in
Geophysics

Citation (APA)
Thorbecke, J., Zhang, L., Wapenaar, K., & Slob, E. (2021). Implementation of the Marchenko multiple elimination algorithm. *Geophysics*, *86*(2), F9-F23. https://doi.org/10.1190/geo2020-0196.1

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Implementation of the Marchenko Multiple Elimination algorithm

Jan Thorbecke[*], *Lele Zhang*[†], *Kees Wapenaar* [‡], *and Evert Slob*[§]

ABSTRACT

The Marchenko multiple elimination and transmission compensation schemes retrieve primary reflections in the two-way traveltime domain without model information or using adaptive subtraction. Both schemes are derived from projected Marchenko equations and similar to each other, but use different time-domain truncation operators. The Marchenko multiple elimination scheme retrieves a new dataset without internal multiple reflections. The transmission compensated Marchenko multiple elimination scheme does the same and additionally compensates for transmission losses in the primary reflections. Both schemes can be solved with an iterative algorithm based on a Neumann series. At each iteration, a convolution or correlation between the projected focusing function and the measured reflection response are performed and after each convolution or correlation, a truncation in the time domain is applied. After convergence, the resulting projected focusing function is used for retrieving the transmission compensated primary reflections and the projected Green's function is used for the physical primary reflections. We demonstrate that internal multiples are removed by using time-windowed input data that only contain primary reflections. We evaluate both schemes in detail and develop an iterative implementation that reproduces the presented numerical examples. The software is part of our open-source suite of programs and fits into the Seismic Unix software suite of the Colorado School of Mines.

INTRODUCTION

The Marchenko algorithm can eliminate internal multiple reflections in reflection data (Slob et al., 2014; Wapenaar et al., 2014a; Behura et al., 2014). For these schemes up- and downgoing focusing functions, with a focal point in the subsurface, are retrieved by solving the coupled Marchenko equations. Green's functions can be computed once these focusing functions are known. A virtual reflection response, with an acquisition surface placed in the subsurface, can be obtained by deconvolving the solved up- and downgoing Green's functions (Slob et al., 2014; Wapenaar et al., 2014b; Broggini et al., 2014; Van der Neut et al., 2015b; Matias et al., 2018). Based on the constructed virtual reflection response, an artifact-free image at the focal point can be created. The measured single-sided reflection response and a smooth velocity model are required for the implementation of these Marchenko redatuming schemes. The iterative implementation of the Marchenko redatuming scheme has been

[*]E-mail: J.W.Thorbecke@tudelft.nl
[†]E-mail: L.Zhang-1@tudelft.nl
[‡]E-mail: C.P.A.Wapenaar@tudelft.nl
[§]E-mail: E.C.Slob@tudelft.nl

discussed in detail and the software has been published by Thorbecke et al. (2017). Lomas and Curtis (2019) illustrate the concepts of the Marchenko method for redatuming and imaging with reproducible scripts in MATLAB. A Marchenko implementation is also part of PyLops (Ravasi and Vasconcelos, 2020).

A wide range of applications has been developed and based on the solutions of the Marchenko equations. Singh et al. (2015) modify the Marchenko scheme to account for free-surface related multiple reflections. Singh et al. (2017), Ravasi (2017) and Dukalski and de Vos (2017) extend the Marchenko redatuming scheme for marine seismic data and show the performance in numerical and field examples. Meles et al. (2018) propose a different time-focusing condition of the Marchenko redatuming scheme for the retrieval of virtual plane-wave responses. The plane-wave scheme allows multiple-free imaging at a fraction of the computational cost of the regular Marchenko scheme. Wapenaar et al. (2017) derive the homogeneous Green's function retrieval scheme from the Marchenko equations, where the homogeneous Green's function between any two points inside a medium can be retrieved from the measured single-sided reflection response. Ravasi et al. (2016),Jia et al. (2018),Staring et al. (2018),Pereira et al. (2019), and Mildner et al. (2019) apply the Marchenko method successfully on field data. Sripanich et al. (2019) derive a method that can estimate initial focusing functions from data and does not rely on a velocity model for mildly varying media. Mildner et al. (2019) develop a method to estimate a source wavelet from the Marchenko focusing functions that enables more precise Marchenko redatuming. The Marchenko redatuming scheme has also been extended from acoustic media to elastic (da Costa Filho, et al. (2014),Wapenaar (2014)) and dissipative (Slob, 2016) media. Lomas et al. (2020) develop a VSP Marchenko imaging methodology that enables imaging of horizontal and vertical structures.

Marchenko-based methods have also been developed for dealing with internal multiple reflections in two-way traveltime domain. Meles et al. (2015) combine the convolutional interferometry with the Marchenko redatuming scheme to approximately attenuate internal multiple reflections. Meles et al. (2017) and da Costa Filho, et al. (2017) introduce a method that can directly construct primaries without need for adaptive subtraction. Van der Neut and Wapenaar (2016) project the coupled Marchenko equations to the surface by convolving both sides of the equations with the initial Green's function to reduce the requirement of model information. The schemes for Marchenko Multiple Elimination (MME) (Zhang and Staring, 2018) and Transmission-compensated Marchenko Multiple Elimination (T-MME) (Zhang et al., 2019) are derived from the projected equations to eliminate all orders of internal multiple reflections without model information or adaptive subtraction. In this case the projected focusing functions are regarded as regular filters that are defined within an offset independent time window, and hence are truly model independent. The MME scheme retrieves the primaries from the reflection data, while the T-MME retrieves the transmission compensated primaries from the upgoing filter function. The MME scheme has been tested on numerical and field data (Zhang and Slob, 2020a). The examples in Zhang and Slob (2019) show that all orders of internal multiple reflections are successfully eliminated by both schemes. Zhang and Slob (2019) extend the MME scheme to account for free-surface related multiple reflections. Thus, free-surface and internal multiple reflections are removed in one step without adaptive subtraction or model information.

In this paper, we describe the implementation of both MME and T-MME schemes in detail. Both schemes eliminate internal multiple reflections without the need for model

information or adaptive subtraction. Only a reflection response without source wavelet and free-surface related multiple reflections is required as input. The paper is organised as follows: In the theory section, we briefly review the equations of the MME and T-MME schemes. In the implementation section the processing details are explained step by step and this section provides a user's first step with the MME and T-MME schemes. The mechanism of the algorithm is illustrated with a simple three-reflector 1.5-dimensional horizontally layered model. This simple model is chosen to keep the number of events limited and to allow for an explanation that can be followed more easily. The method is not limited to simple models and can successfully be applied to complicated 3D media as well (see for example Zhang and Slob (2020c)).

The software accompanied with this paper contains scripts and source code to reproduce all the numerical examples presented in this paper. The code can also be found at its GitHub repository (Thorbecke et al., 2017; Thorbecke and Brackenhoff, 2020), where the most recent version and latest developments are available. The commands to reproduce all figures in this paper can be found in the directory `marchenko/demo/mme`. The README_PRIMARIES in that directory explains in detail how to run the scripts. A more complicated (lateral varying) model can be found in the directory `marchenko/demo/twoD`. This example will take several hours to compute the reflection data and is not discussed here. To reproduce the figures and to carry out a few pre- and post-processing steps, Seismic Unix (Cohen and Stockwell, 2016) is required.

THEORY

In this section we give a brief overview of the theory of both MME and T-MME schemes. The acquisition surface is located at the surface boundary $\partial\mathbb{D}_0$. The reflection response $R(\mathbf{x}_0, \mathbf{x}_0', t)$ is measured with source and receiver positioned at $\mathbf{x}_0'$ and $\mathbf{x}_0$, which is free from free-surface related multiple reflections and source wavelet. The time is denoted t.

MME

As presented by Zhang et al. (2019), we give the equations of the Marchenko Multiple Elimination (MME) scheme as

$$R_t(\mathbf{x}_0', \mathbf{x}_0'', t = t_2) = R(\mathbf{x}_0', \mathbf{x}_0'', t = t_2) + \sum_{m=1}^{\infty} M_{2m}(\mathbf{x}_0', \mathbf{x}_0'', t = t_2, t_2), \tag{1}$$

with

$$\begin{aligned} M_{2m}(\mathbf{x}_0', \mathbf{x}_0'', t, t_2) = &\int_{t'=0}^{+\infty} \int_{\partial\mathbb{D}_0} R(\mathbf{x}_0''', \mathbf{x}_0', t') H(t - t' - \varepsilon) \mathrm{d}\mathbf{x}_0''' \mathrm{d}t' \times \\ &\int_{t''=0}^{+\infty} \int_{\partial\mathbb{D}_0} R(\mathbf{x}_0, \mathbf{x}_0''', t'') H(t' - t + t_2 - t'' - \varepsilon) \times \\ &M_{2(m-1)}(\mathbf{x}_0, \mathbf{x}_0'', t - t' + t'', t_2) \mathrm{d}\mathbf{x}_0 \mathrm{d}t'', \end{aligned} \tag{2}$$

and initialization

$$M_0(\mathbf{x}_0', \mathbf{x}_0'', t, t_2) = -(H(t + t_2 - \varepsilon) - H(t + \varepsilon))R(\mathbf{x}_0', \mathbf{x}_0'', -t), \tag{3}$$

where R_t denotes the retrieved dataset without internal multiple reflections at time t_2 and H indicates the Heaviside function, which is used to apply the offset independent truncation window $(\varepsilon, t_2 - \varepsilon)$ in the equations. The constant ε indicates a small positive value which can be taken as the half source time-duration in practice. The initialization of the scheme (with M_0) is the time-reversed shot record at shot position $\mathbf{x}_0''$ for times between $(\varepsilon, t_2 - \varepsilon)$. The R_t in the left side of equation 1 is the same shot record, but without internal multiples. We follow Zhang and Staring (2018) and make time t_2 constant and independent of the source and receiver positions in the reflection response. Note that the integration is carried out over the receiver coordinate for both integrals, the same as implemented in the source code. The second term in the right-hand side of equation 1 predicts all internal multiple reflections correctly. Equation 3 indicates that the measured reflection response is the only input of the MME scheme given in equation 1. To retrieve a dataset without internal multiple reflections for all times t this process must be repeated for all times t_2.

Equation 2 contains the terms that correct for the internal multiples that are present in $R(\mathbf{x}_0', \mathbf{x}_0'', t)$. To better explain the right-hand side of equation 2 we divide the expression into two parts:

$$M_{2m}(\mathbf{x}_0', \mathbf{x}_0'', t, t_2) = \int_{t'=0}^{+\infty} \int_{\partial\mathbb{D}_0} R(\mathbf{x}_0''', \mathbf{x}_0', t')H(t - t' - \varepsilon)\times \\ M_{2m-1}(\mathbf{x}_0''', \mathbf{x}_0'', t - t', t_2)\mathrm{d}\mathbf{x}_0'''\mathrm{d}t', \tag{4}$$

$$M_{2m-1}(\mathbf{x}_0''', \mathbf{x}_0'', t - t', t_2) = \int_{t''=0}^{+\infty} \int_{\partial\mathbb{D}_0} R(\mathbf{x}_0, \mathbf{x}_0''', t'')H(t' - t + t_2 - t'' - \varepsilon)\times \\ M_{2(m-1)}(\mathbf{x}_0, \mathbf{x}_0'', t - t' + t'', t_2)\mathrm{d}\mathbf{x}_0\mathrm{d}t''. \tag{5}$$

Equation 4 is a time domain *convolution* of R with M integrated over the spatial coordinate $\mathbf{x}_0'''$, which is the receiver position of the shot at $\mathbf{x}_0'$. Equation 5 is a time domain *correlation* of R with M integrated over the spatial coordinate $\mathbf{x}_0$, which is the receiver position of the shot at $\mathbf{x}_0'''$. The Heaviside function in equation 4 is to exclude negative times for $t - t' < \varepsilon$ and only use the causal time in M_{2m-1}. In equation 5 the function $M_{2(m-1)}$ contains non-zero values for times larger than $t_2 - \varepsilon$. These times should not contribute to the integral and the Heaviside guarantees that $M_{2(m-1)}$ does not have a contribution to the integration result for values of $t'' + t - t' > t_2 - \varepsilon$. Note that equation 4 and 5 perform very similar operations (they differ by a sign change) and are implemented by a single function. To evaluate equation 2 this function is applied two times. The convolution terms are the even numbered operations with this function and the correlation terms are the odd numbered operations.

To better explain the method and for illustration purposes the summation of the even M terms in equation 4 is defined as the field:

$$k_{1,i}^{-}(\mathbf{x}_0', \mathbf{x}_0'', t, t_2) = R(\mathbf{x}_0', \mathbf{x}_0'', t, t_2) - \sum_{m=1}^{i} \int_{t'=0}^{+\infty} \int_{\partial\mathbb{D}_0} R(\mathbf{x}_0''', \mathbf{x}_0', t')H(t - t' - \varepsilon)\times \\ M_{2m-1}(\mathbf{x}_0''', \mathbf{x}_0'', t - t', t_2)\mathrm{d}\mathbf{x}_0'''\mathrm{d}t'. \tag{6}$$

We can evaluate equation 6 also for $t \geq t_2 - \varepsilon$ and the equation can be further split in the time domain as follows

$$k^-_{1,i}(\mathbf{x}'_0, \mathbf{x}''_0, t, t_2) = \begin{cases} v^-_{1,i}(\mathbf{x}'_0, \mathbf{x}''_0, t, t_2) & t < t_2 - \varepsilon \\ u^-_{1,i}(\mathbf{x}'_0, \mathbf{x}''_0, t, t_2) & t \geq t_2 - \varepsilon \end{cases}, \tag{7}$$

where u^-_1 and v^-_1 are similar to the projected Green's function and focusing function in the regular Marchenko scheme as defined in Van der Neut and Wapenaar (2016). The minus superscript in equations 6 and 7 refers to upgoing wavefields at the receiver location $\mathbf{x}'_0$. To solve for M in equations 1-3, k^-_1 is not needed. The time values in M between ε and $t_2 - \varepsilon$ are used to solve the Marchenko equations and compute a value at t_2 in u^-_1. The solution for time t_2 in the MME scheme is collected from u^-_1 at $t = t_2$.

Similarly for the summation of the terms in equation 5, a downgoing function is defined as

$$v^+_{1,i}(\mathbf{x}'''_0, \mathbf{x}''_0, t, t_2) = \sum_{m=1}^{i} \int_{t''=0}^{+\infty} \int_{\partial \mathbb{D}_0} R(\mathbf{x}_0, \mathbf{x}'''_0, t'') H(t' - t + t_2 - t'' - \varepsilon) \times \\ M_{2(m-1)}(\mathbf{x}_0, \mathbf{x}''_0, t + t'', t_2) \mathrm{d}\mathbf{x}_0 \mathrm{d}t'', \tag{8}$$

where v^+_1 is similar to the projected focusing function in the regular Marchenko scheme as defined in Van der Neut and Wapenaar (2016). In v^+_1 the multiple annihilator is created and this is demonstrated in the numerical examples section. Equation 8 only holds for $\varepsilon < t < t_2 - \varepsilon$. The plus superscript in equation 8 refers to downgoing wavefields. To solve the MME Marchenko equations v^+_1 is not needed and only defined for illustration purposes to explain the mechanism of the method.

Time t_2 is the instant two-way travel-time where the solution of the Marchenko equation is computed. The primary reflection is collected from u^-_1 for every time instant t_2. This is a computational expensive way, since only one sample is collected in the output. Nevertheless, this process is fully automated and implemented without any human interaction or model information. It is possible to collect more that one sample around the instant time t_2, and take bigger time steps, but the number of samples to use around t_2 must take into consideration the frequency bandwidth of the data. This statement is supported by examples in the detailed discussion of the implementation and allows the implementation of a faster algorithm.

In this MME scheme the primary is collected from the original reflection data. The Marchenko scheme removes all overlapping internal multiples from earlier reflections, the primary is untouched and keeps the physical reflection amplitude as present in the data.

T-MME

Both internal multiple reflections and transmission losses in primary reflections can be accounted for by the Transmission-compensated Marchenko Multiple Elimination (T-MME)

scheme (Zhang et al., 2019). The equation is given by

$$R_r(\mathbf{x}_0', \mathbf{x}_0'', t = t_2) = R(\mathbf{x}_0', \mathbf{x}_0'', t = t_2) + \sum_{m=1}^{\infty} \bar{M}_{2m}(\mathbf{x}_0', \mathbf{x}_0'', t = t_2, t_2), \tag{9}$$

with

$$\begin{aligned}\bar{M}_{2m}(\mathbf{x}_0', \mathbf{x}_0'', t, t_2) = \int_{t'=0}^{+\infty} \int_{\partial\mathbb{D}_0} R(\mathbf{x}_0''', \mathbf{x}_0', t') H(t - t' + \varepsilon) \mathrm{d}\mathbf{x}_0''' \mathrm{d}t' \times \\ \int_{t''=0}^{+\infty} \int_{\partial\mathbb{D}_0} R(\mathbf{x}_0, \mathbf{x}_0''', t'') H(t' - t + t_2 - t'' + \varepsilon) \\ \bar{M}_{2(m-1)}(\mathbf{x}_0, \mathbf{x}_0'', t - t' + t'', t_2) \mathrm{d}\mathbf{x}_0 \mathrm{d}t''\end{aligned} \tag{10}$$

and

$$\bar{M}_0(\mathbf{x}_0', \mathbf{x}_0'', t, t_2) = -(H(t + t_2 + \varepsilon) - H(t + \varepsilon)) R(\mathbf{x}_0', \mathbf{x}_0'', -t), \tag{11}$$

where R_r denotes the retrieved dataset without internal multiple reflections and transmission losses in primary reflections. The truncation window in equation 10 is different from the window in equation 2, which guarantees that the second term in the right-hand side of equation 9 predicts both internal multiple reflections and transmission losses in primary reflections. In equation 10 the Heaviside guarantees that $\bar{M}_{2(m-1)}$ does not have a contribution for values of $t'' + t - t' > t_2 + \varepsilon$. In contrast to equation 5, time t_2 is now part of the integration and included in the sum of $\bar{M}_{2m}$. Still, as given in equation 11, the measured reflection response is the only input to solve the T-MME scheme given in equation 9.

The primary reflection is, different than in the MME scheme, collected from $\bar{v}_1^-$, which achieves the transmission compensation. There is no need to define a $\bar{k}_1^-$, since $\bar{v}_1^-$ is already part of $\bar{M}_{2m}$. The scheme is applied for every time instant t_2 and has the same advantages and disadvantages as the MME scheme. In the T-MME scheme the amplitude of the primary is automatically transmission compensated, because it is the only way to predict and attenuate internal multiples from earlier primary reflections. We come back to this remark in the explanation of Figure 1.

Both MME and T-MME schemes require **only** the measured reflection response R as input. The reflection response R needs to be deconvolved for the source-wavelet and the free-surface related multiple reflections must be removed. The output of a surface-related multiple elimination (SRME Verschuur et al. (1992)) scheme can meet these requirements. Diffracted and refracted waves are beyond the capability of both schemes and a detailed analysis about these limitations can be found in Zhang et al. (2019).

MARCHENKO ALGORITHM

The basic Marchenko algorithm (MME) is explained in Algorithm 1. The arrays in this algorithm are stored in C-order; the last (most right) addressed dimension is contiguous in memory. The discrete dimensions of these arrays are within square brackets [...], the arguments of function calls are within regular brackets (...). The *only* data input of the algorithm is the measured refection data R. This reflection data must be properly pre-

processed as explained in Brackenhoff et al. (2019). The pre-processing must take care of the following:

- Elimination of free-surface multiples.
  Note that there is also a very similar Marchenko algorithm that takes into account free-surface multiples as well. Ravasi (2017) discusses a redatuming algorithm similar to Singh et al. (2015) and requires a smooth model of the medium, while Zhang and Slob (2019) remove all multiples and does not need any model information.
- Sufficient (i.e. alias free) sampling in the spatial receiver and shot direction.
  Note, there are Marchenko-based methods that can fill in missing shot points or receiver locations, under the assumption that the available data are unaliased (Wapenaar and van IJsseldijk, 2020).
- Compensate for dissipation.
- Shot amplitude regularization.
- Deconvolution for source wavelet.

Following Algorithm 1, the pre-processed reflection data is read from disk, transformed (by FFT operator $\mathcal{F}\{\dots\}$) to the frequency (ω) domain and all shots and receivers are stored into memory. This is the first step in the algorithm and the only significant data read. One single shot record (with shot-number j), where we want to suppress the internal multiples from, is selected from this reflection data in the next step. This shot record is transformed back to time, time reversed (R^*), and stored in array DD. The first loop in the algorithm loops over the selected number of time samples that are processed to attenuate internal multiples. Typically this represents all samples in the shot record, with a possible exclusion of the number of samples to the first reflection event in the selected shot record. For each time sample ii the iterative Marchenko algorithm is executed. The largest difference from the algorithm described in Thorbecke et al. (2017) is that time-truncation along the first arrival-time (from a focal point in the subsurface) is replaced by a constant time-truncation and the computation of a first arrival-time is not needed anymore. The initialization of the algorithm by M_0 is from the same shot record j from which we would like to attenuate the internal multiples (DD). M_0 is a copy of the time reversed shot record, see equation 3, and set to zero from the first sample 0 to sample $n_t - ii + n_\varepsilon$, where n_t is the total number of samples in the shot record. The extra samples of n_ε take into account the time duration of the wavelet to exclude a possible reflection event at time ii. The initialization of $k^-_{1,0}$ is a complete (no time-muting is carried out) copy of the shot record that still contains all internal multiples we would like to remove.

With these two initializations the iterations of the Marchenko algorithm can start. In each iteration an updated field is computed by the integration of M_i with R. This integration process is called `synthesis`, produces the output RM_i, and is explained in more detail below. Depending on the iteration number i, being odd or even, different time muting windows are in use to mute events in RM_i and to compute an updated M_{i+1}. For even iterations the times between $ii - n_\varepsilon$ and n_t are set to zero and for odd iterations the times between 0 and $ii + n_\varepsilon$ are set to zero. Only in the odd iterations $k^-_{1,i}$ is updated with the unmuted M_{i+1}. In this update of $k^-_{1,i}$ internal multiples around time ii are attenuated. This

is the update represented in equation 1, where the update M_{2m} is in fact one even and one odd iteration in the implemented Marchenko algorithm, and hence the notation with $_{2m-1}$ in equation 6.

In the regular redatuming Marchenko algorithm (Thorbecke et al., 2017) the truncation windows follow the first arrival time of a focal point in the subsurface. In the Marchenko multiple elimination algorithm, the focal point is projected on the surface and the time-truncation is conveniently chosen at a constant time. The flat time window has the big advantage that it requires no additional model- or data-information. Meles et al. (2020) demonstrate that in the application of the multiple elimination algorithm to dipping plane waves a time truncation consistent with the dip-angle must be used.

Depending on the position of strong reflectors typically, 10-50 Marchenko iterations are needed for each time sample ii in the selected shot record. The presence of strong reflectors in the shallow part makes the convergence slow at large time instances, see also Figure 8. The reason is that higher-order multiples are attenuated with events that are created in v_1^- and that are removed again later when the first-order multiple is finally removed by a converged multiple attenuator. When the first-order multiple is removed, all multiples are removed and hence all earlier higher-order multiple attenuator artifacts will also vanish. Once the iterations are finished the output of sample ii of the updated Marchenko result u_1^- is stored in sample ii in the multiple free shot record R_t, the final output of the program that represents the selected shot record with attenuated internal multiples. It is a compute intensive task to solve the Marchenko equations for each sample ii in the shot record. Algorithm 2 is a faster (10-20x) implementation of Algorithm 1.

Main **begin**
  Read SU-style input parameters
  Initialization, reading of input parameters and allocate arrays
  READ($R[N_{shots}, i\omega, N_{recv}]$)
  $DD[N_{recv}, it] = \mathcal{F}^{-1}\{R^*[j, i\omega, N_{recv}]\}$
  for $ii \leftarrow istart$ **to** $iend$ **do**
    $M_0[N_{recv}, it] = \begin{cases} 0 & 0 < it < n_t - ii + n_\varepsilon \\ -DD[N_{recv}, it] & n_t - ii + n_\varepsilon \leq it < n_t \end{cases}$
    $k^-_{1,0}[N_{shots}, it] = DD[N_{recv}, n_t - it]$
    $v^+_{1,i}[N_{shots}, it] = 0$
    for $i \leftarrow 0$ **to** n_i **do**
      synthesis(R, M_i, RM_i)
      $M_{i+1}[N_{shots}, it] = RM_i[N_{shots}, n_t - it]$
      if *(i % 2 == 0)* **then**
        $M_{i+1}[N_{shots}, it] = 0; \quad ii - n_\varepsilon < it < n_t$
        $v^+_{1,i+1}[N_{shots}, it] = v^+_{1,i}[N_{shots}, it] + M_{i+1}[N_{shots}, it]$
      else
        $k^-_{1,i+1}[N_{shots}, it] = k^-_{1,i}[N_{shots}, it] - M_{i+1}[N_{shots}, n_t - it]$
        $M_{i+1}[N_{shots}, it] = 0; \quad 0 < it < n_t - ii + n_\varepsilon$
      end
    end
    $R_t[j, N_{shots}, ii] = k^-_{1,n_i}[N_{shots}, ii]$
  end
end

Algorithm 1: Basic Marchenko algorithm, without transmission loss compensation, as implemented in the provided source code. Integer time sample number it, that runs from $istart$ to $iend$, represents time $t = it * \Delta t$. The number of recorded time samples is n_t, the time duration of the source signature $\varepsilon = n_\varepsilon * \Delta t$ and time sample ii represents instant time $t_2 = ii * \Delta t$. The number of receivers in R is N_{recv} and number of shots N_{shots}. The i loop represents the number of Marchenko iterations n_i. Note that the sample expression $n_t - it$ stands for negative time $-t$.

In Algorithm 2, after the Marchenko equations are solved at time sample $ii-1$, the next time sample ii is initialized with the result of time sample $ii-1$ (Zhang and Slob, 2020b). The idea is that to remove the internal multiples at the next time sample there is no need to start from scratch and remove the multiples that were already removed in the previous time-sample. For every next time-sample all earlier attenuated multiples need to be attenuated plus one (or a few) more. In the Marchenko update for the new time sample only the multiples have to be removed that were not removed before. This is a small deviation from the previous results and usually 2 iterations are sufficient to accomplish the update for the next time sample. The initial M_0 in the fast algorithm is the difference between the original data (DD) and $k^{-,(ii-1)}_{1,n_i}$; the already estimated internal multiples from time sample $ii-1$. The initial $k^{-}_{1,0}$ is the previous result $k^{-,(ii-1)}_{1,n_i}$. With these initializations the update term RM_i contains only a small correction, since it is based on a converged previous result that is very close to the actual solution. To get the complete internal multiple in M_{i+1}, DD is added to RM_i (Zhang and Slob, 2020b).

In this fast algorithm only one pair of even-odd iterations is needed to reach convergence. In principle we could solve the equations only one time and use that result to update all other time samples. On simple models of numerically modeled data this works fine indeed. However, on geologically complicated models of numerically modeled data and on field data we have to do a full update every 10 to 20 recursive updates and the speed-up of the faster algorithm is limited to one order of magnitude. On complex data-sets we would advise to begin with the basic algorithm and then verify if the fast algorithm can be used to speed-up the computations. The reason for this limited use of the fast algorithm is that for complex data-sets and a large number of iterations, artifacts, for example introduced by a limited aperture, can get amplified. The primary reflections will still converge, but numerical artifacts are not accounted for in the algorithm and can diverge. In the iterative scheme each update adds two iterations to the already computed result based on for example 30 iterations. With 10 iterative updates, 20 iterations are added and can cause artifacts being amplified to signal level.

In the algorithm we solve the Marchenko equations for each sample ii. From the theory we know that the first event after sample ii is a primary reflector (all multiple reflections generated by the reflectors before sample ii are removed by the scheme). Hence, a number of samples after sample ii will still be free of internal multiples. We could make larger steps with ii and use the Marchenko result for a number of samples (at least n_ε samples, since that is the time resolution we are already working with) after sample ii. This can speed-up the code by n_ε (typically 20) times. This is similar to the fast algorithm, but without making any iterations and directly use the previous computed result.

The synthesis process shown in Algorithm 3 computes the second integrant in the right-hand side of equation 2. The synthesis function is a straightforward matrix-vector multiplication. The reflection data are stored in such a way that the most inner loop, that sums over the receiver positions within a shot, is contiguous in memory. To speed-up the computation a parallel OpenMP region is carried out over the outer N_{shots} loop. An alternative implementation of the synthesis process is to make the frequency loop the outer loop and use a BLAS `dgemv` function to compute the matrix-vector multiplication. This implementation will also be efficient when all shots are computed at the same time and the BLAS matrix-matrix `dgemm` function becomes the kernel of the synthesis process. Note that in the synthesis process the integration is carried out over the number of receivers per shot

Main **begin**
  Read SU-style input parameters
  Initialization, reading of input parameters and allocate arrays
  READ($R[N_{shots}, i\omega, N_{recv}]$)
  $DD[N_{recv}, it] = \mathcal{F}^{-1}\{R^*[j, i\omega, N_{recv}]\}$
  for $ii \leftarrow istart$ **to** $iend$ **do**
    $k^-_{1,0}[N_{shots}, it] = k^{-,(ii-1)}_{1,n_i}[N_{shots}, it]$
    $M_0[N_{shots}, it] =$

$$\begin{cases} 0 & 0 < it < n_t - ii + n_\varepsilon \\ DD[N_{shots}, n_t - it] - k^-_{1,0}[N_{shots}, n_t - it] & n_t - ii + n_\varepsilon \le it < n_t \end{cases}$$

    for $i \leftarrow 0$ **to** n_i **do**
      synthesis(R, M_i, RM_i)
      $M_{i+1}[N_{shots}, it] = RM_i[N_{shots}, n_t - it]$
      if *(i % 2 == 0)* **then**
        $M_{i+1}[N_{shots}, it] = 0; \quad ii - n_\varepsilon < it < n_t$
      else
        $M_{i+1}[N_{shots}, it] = M_{i+1}[N_{shots}, it] - DD[N_{recv}, it]$
        $k^-_{1,i+1}[N_{shots}, it] = -M_{i+1}[N_{shots}, n_t - it]$
        $M_{i+1}[N_{shots}, it] = 0; \quad 0 < it < n_t - ii + n_\varepsilon$
      end
    end
    $R_t[j, N_{shots}, ii] = k^-_{1,n_i}[N_{shots}, ii]$
  end
end

Algorithm 2: Faster Marchenko algorithm that uses previous results from time instant $ii - 1$ ($k^{-,(ii-1)}_{1,n_i}$) as input for the current time instant ii.

and each integration result is stored at the shot position. Thus after the synthesis process N_{shots} output traces are computed.

From a computational point of view the transmission compensated algorithm (T-MME) is the same as the MME algorithm, except for the application of the time-truncation window. The sample length of the wavelet (n_ε) is applied in the opposite time direction for the T-MME algorithm. The extra samples of n_ε in the MME algorithm take into account the length of the wavelet to **exclude** a possible event at instant time ii in the initialization and update of M_i. Suppose that time ii is the two-way traveltime of a reflector (see Figure 1a). The reflection of the reflector is excluded in M_i in the MME algorithm (Figure 1a), but included in $\bar{M}_i$ in the T-MME algorithm (Figure 1b). In the T-MME algorithm the reflected event at instant time ii ends up in the updated $\bar{v}^-_1$, while in the MME algorithm the reflected event from the original shot record ends up in u^-_1. When the instant time ii is chosen between two reflectors then there is no difference between the updates made in the MME (Figure 1c) or T-MME (Figure 1d) scheme.

synthesis($R[N_{shots}, i\omega, N_{recv}), M[N_{shots}, it], RM[N_{shots}, it]$)
begin
  $Fop[i\omega, N_{shots}] = \mathcal{F}\{M[N_{shots}, it]\}$
  $RM[N_{shots}, t] = 0$
  #pragma omp parallel for
  for $k \leftarrow 0$ **to** N_{shots} **do**
    for $i\omega \leftarrow \omega_{min}$ **to** ω_{max} **do**
      for $i \leftarrow 0$ **to** N_{recv} **do**
        $sum[i\omega] = sum[i\omega] + R[k, i\omega, i] * Fop[i\omega, i]$
      end
    end
    $RM[k, it] = \mathcal{F}^{-1}\{sum[i\omega]\}$
  end
end

Algorithm 3: Marchenko synthesis kernel with $i\omega = i * \Delta\omega (= \frac{2\pi}{n_t * \Delta t})$.

To get to the T-MME scheme from Algorithm 1 one has to replace in the defined time windows $+n_\varepsilon$ with $-n_\varepsilon$. Then R_t becomes R_r that contains the transmission compensated primary reflections.

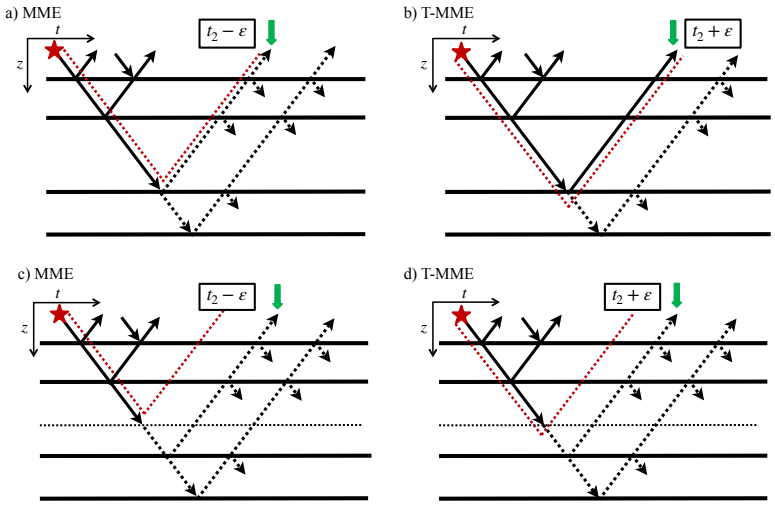


Figure 1: Comparison of the MME and and T-MME schemes. Figures a) and b) show a selected time t_2 equal to the two-way traveltime of the third reflector. The time-truncation window is indicated with a red dotted line. The dotted lines are events that are excluded in M_i, the solid lines are events included in M_i, after application of the time window. Figures c) and d) show a time between two reflectors.

NUMERICAL EXAMPLES

The Marchenko algorithm is illustrated with a 1.5-dimensional horizontally layered model shown in Figure 2. The numerical modeling is carried out with a finite-difference modeling program (Thorbecke and Draganov, 2011) that is also included in the software package. The input source signature, to model the reflection response $R(\mathbf{x}_0', \mathbf{x}_0'', t)$, is approximately a sinc-function with a flat spectrum of amplitude 1 between f_{min} and f_{max} $(5 - 90$ Hz) to represent a deconvolved source wavelet. In the finite-difference program for modeling $R(\mathbf{x}_0', \mathbf{x}_0'', t)$ a source of vertical force is chosen. The receivers are placed at $z = 0$ and measure the pressure field. A fixed spread acquisition is chosen between -2250 to 2250m and the distance between the 901 source/receiver positions is 5m. The receiver traces have a time sampling interval of 4ms and 1024 recorded time samples.

The first iterations

Figure 3 demonstrates the first iteration of equation 5 with $m = 1$ to compute M_1 for time sample-number 276 (t=1.100 s) from M_0. Time-sample 276 corresponds to the zero-offset arrival time of the third reflector. In this first step all shots in the reflection data R are correlated with a time-windowed shot record. In our example we use the middle shot record; $R_0(\mathbf{x}_R, \mathbf{x} = (0, 0), t)$ (shot-number $j = 451$). Before the correlation is carried out the selected shot record is first set to zero beyond time sample $276 - n_\varepsilon$, multiplied by -1 and time reversed, at which moment we have $M_0(t)$ in equation 3. In Figure 3b the shot record is convolved with a Ricker wavelet to reduce the ringing of the flat spectrum of the (deconvolved) wavelet present in R (Figure 3a). The number of n_ε (in this example $n_\varepsilon = 20$)

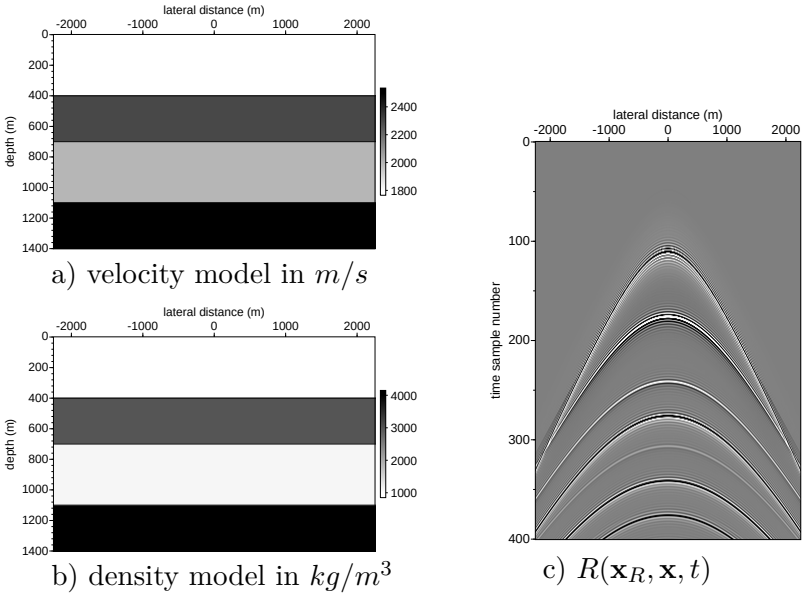


Figure 2: Four layer model with velocity (a) and density (b) parameters. A shot record, with source position $\mathbf{x} = (x = 0, z = 0)$ and receivers at $\mathbf{x}_R = (x = x_r, z = 0)$ (c). The source wavelet in R (c) has a flat frequency spectrum from 5 to 90 Hz.

samples excludes the reflection from the third reflector in M_0. In Figure 3 the middle shot record of R (Figure 3a, where we used source receiver reciprocity) is correlated with the time windowed $M_0(-t)$ (Figure 3b) to give the result in Figure 3c. The events in Figure 3b include the first and second reflection and the first internal multiple between the first and second reflector. In the correlation result (Figure 3c) we see the auto-correlation of the three reflection events around $t = 0$ (with events at negative times appearing at the bottom of the panel). Note that the long train of events starting at the positive time-axis in Figure 3c can interfere with events at the end of the time axis. To overcome this time interference we usually pad the time axis with zeros before the transformation to the frequency domain where the correlation is computed.

The correlation result is time-reversed and shown in Figure 3d for the first 400 samples. There are only three events in Figure 3d and these originate from correlation with the three events in Figure 3b with the first three events $(\mathbf{r}_1, \mathbf{r}_2, \mathbf{m}_1)$ in the shot record. According to the integral in equation 5, to obtain an output trace of M_1 the traces in Figure 3d are summed together. The stationary points of the events in Figure 3d give a contribution in the result of the summation. Besides the stationary points, truncated events (both in time and space) give unwanted contributions that show up as artifacts in the final result.

The integration result is set to zero for samples larger than $276 - n_\varepsilon$ and ends up as a trace at position 0 (the middle trace) of M_1 shown in Figure 3e. In Figure 3e the truncation appears to be at sample 160, but that is the truncation in 3b shifted upward in time with the arrival time of the first reflector. The truncation at sample $276 - n_\varepsilon$ is indicated with a dotted line. There are two hyperbolic events visible and a few linear artifacts. The first hyperbolic event originates from correlation of events $\mathbf{r}_2^*.\mathbf{r}_1$ and events $\mathbf{m}_1^*.\mathbf{r}_2$, and the second hyperbolic event from the first internal multiple and first reflector $\mathbf{c}_2 = \mathbf{m}_1^*.\mathbf{r}_1$. The

linear events are unwanted artifacts due to truncation and can be suppressed by applying a smooth taper at the truncation boundaries in time and space.

Figure 4 demonstrates the computation of the second iteration to compute M_2 from M_1 (Figure 3e) according to equation 4 with $m = 1$. The reflection data (Figure 4a) are convolved with M_1 (Figure 4b) that contains three main events; a linear-artifact ($\mathbf{a}_1$) and two correlation results ($\mathbf{c}_1$ and $\mathbf{c}_2$). Convolving M_1 with the middle shot record of the data R gives Figure 4c. The hyperbolic events in M are now back at the same times as reflection events in the shot-record. The linear artifacts in M_1 also convolves with all events in R and introduce many (mostly linear) artifacts. The convolution result is reversed in time (Figure 4d) and after the integration in equation 4 over the lateral coordinate $\mathbf{x}_R$ it becomes the middle trace in M_2 (Figure 4e). Most of the linear artifacts are reduced in amplitude due to the destructive interference in the integration, only the '$\mathbf{a}_2 = \mathbf{r}_1.\mathbf{a}_1$' artifact is still present in M_2. The first term in the sum in equation 1 is now computed; M_2. The last events in the time reverse of M_2, presented in Figure 4e, will already attenuate the multiple event $\mathbf{m}_1$ in the shot-record.

To compute M_1 (in general odd numbered updates to M_i), events are shifted backward in time (correlation) with the times of the events in M_0. To compute M_2 (even numbered updates to M_i) from M_1, events are shifted forward (convolution) in time. The even and odd iterations are treated differently in the scheme. Each even iteration updates M_i and v_1^+, and each odd iteration updates M_i and k_1^-. The scheme reverts the time-axis for each iteration, hence the time windows, that set time samples to zero, switches also. These time windows, for sample 276, are shown in Figure 5. In these time windows a smooth cosine shaped transition zone is used to reduce the time-truncation artifacts.

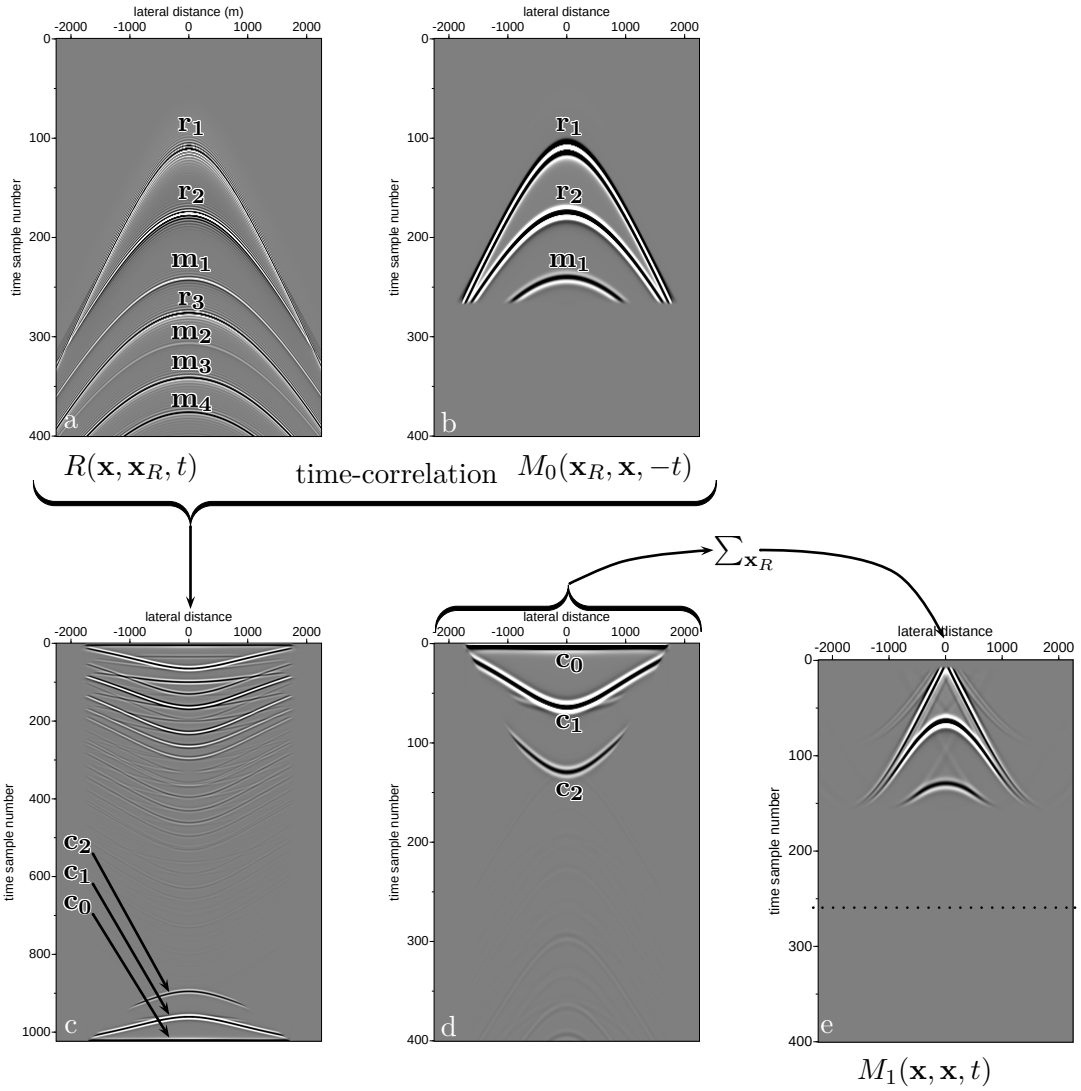


Figure 3: Computational steps to compute M_1 from M_0 at time sample number 276. The middle shot record from R is shown in (a); time truncated after sample $276 - n_\varepsilon$, and convolved with a Ricker wavelet, it gives M_0 in (b). Time-correlation of (a) with (b) gives (c). After time-reversal of (c) and applying the time window again gives (d). The traces in (d) are summed together and only the stationary point of events above sample $276 - n_\varepsilon$ will end-up in the middle trace of M_1 (e). The mute window n_ε samples later than $t = 0$ is needed to mute the autocorrelation of the first-event. The labeled events $\mathbf{r}_i$ indicate the i'th reflector and $\mathbf{m}_l$ the l'th multiple. In (c) the labeled correlated events are $\mathbf{c_0} = \mathbf{r_1^*}.\mathbf{r_1} + \mathbf{r_2^*}.\mathbf{r_2} + \mathbf{m_1^*}.\mathbf{m_1}$, $\mathbf{c_1} = \mathbf{r_2^*}.\mathbf{r_1} + \mathbf{m_1^*}.\mathbf{r_2}$, and $\mathbf{c_2} = \mathbf{m_1^*}.\mathbf{r_1}$.

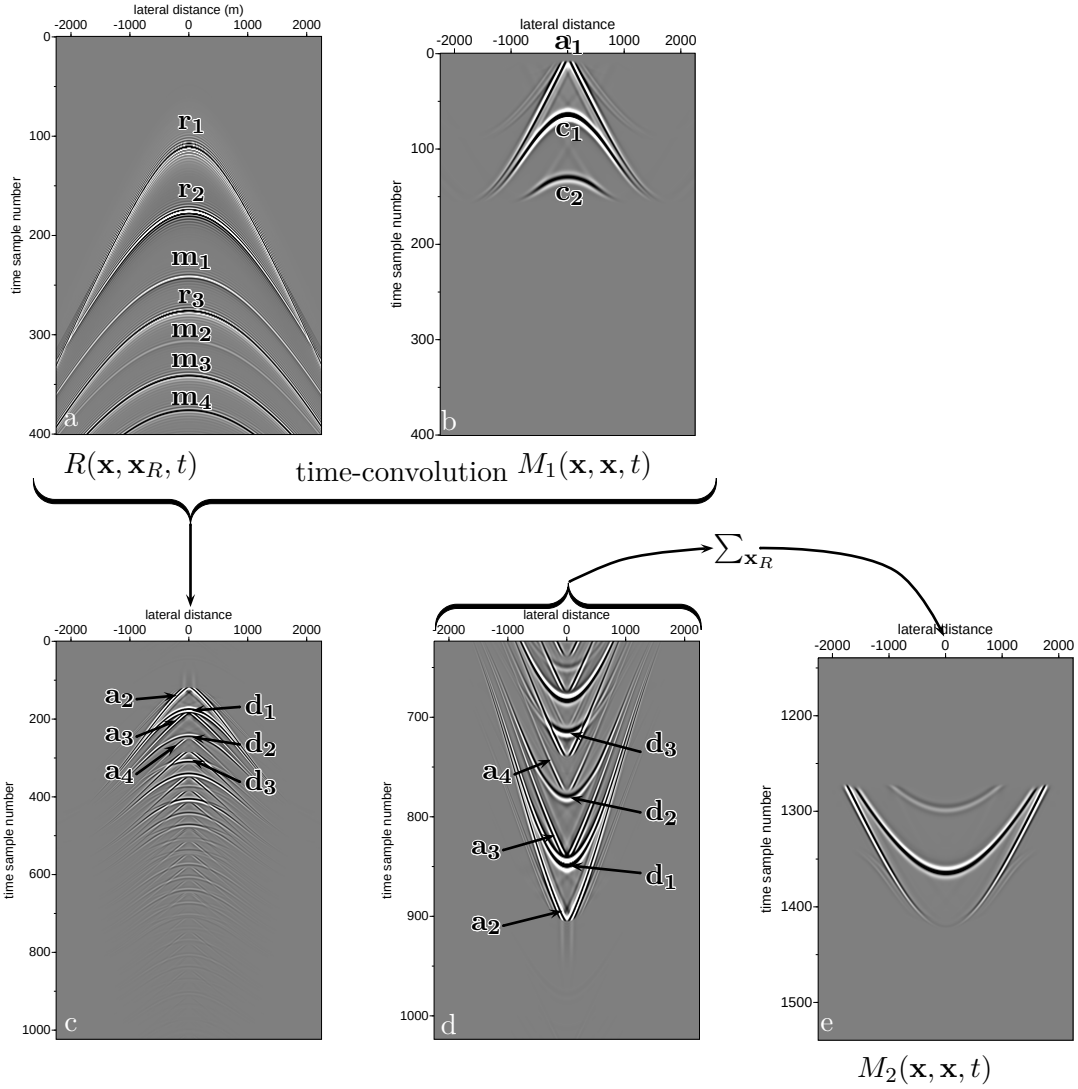
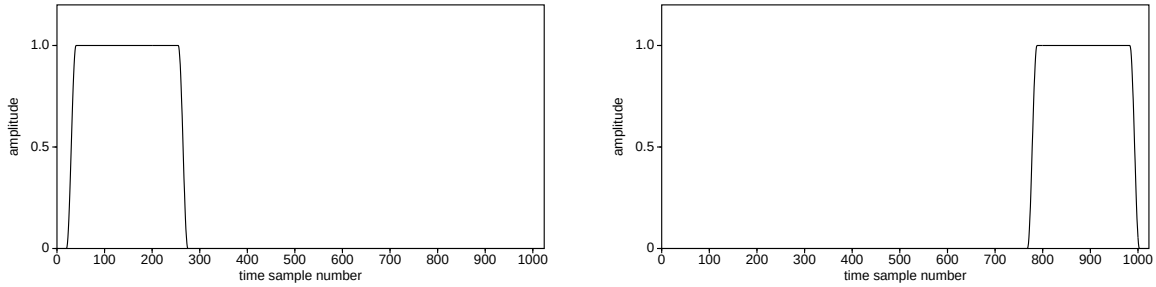


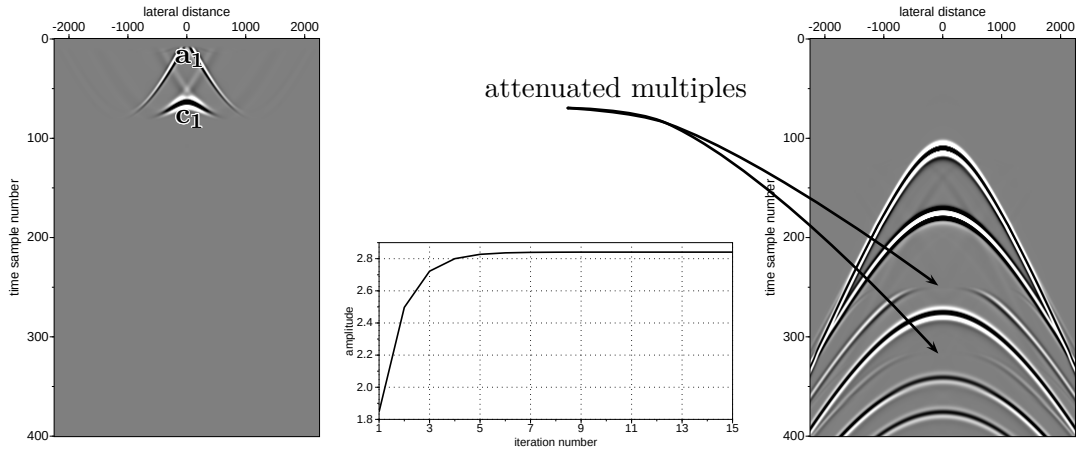
Figure 4: Computational steps to compute $M_2(\mathbf{x}, \mathbf{x}, t)$ from M_1 at time sample number 276. The middle shot record from R is shown in (a) and M_1 truncated after sample 276 (and computed in Figure (3e)) in (b). Time-convolution of (a) with (b) gives (c). After time-reversal of (c) and applying the time window again gives (d). The traces in (d) are summed together and only stationary events later than sample $n_t - 276 + n_\varepsilon$ end-up in the middle trace of M_2 (e). The labeled events $\mathbf{r}_i$ indicate the i'th reflector, $\mathbf{m}_l$ the l'th multiple and $\mathbf{a}_n$ the n'th artifact. The labeled events from the convolution between R(a) and M_1(b) are $\mathbf{a_2} = \mathbf{r_1}.\mathbf{a_1}, \mathbf{a_3} = \mathbf{r_2}.\mathbf{a_1}, \mathbf{a_4} = \mathbf{m_1}.\mathbf{a_1}, \mathbf{d_1} = \mathbf{r_1}.\mathbf{c_1}, \mathbf{d_2} = \mathbf{r_1}.\mathbf{c_2} + \mathbf{r_2}.\mathbf{c_1}$ and $\mathbf{d_3} = \mathbf{r_2}.\mathbf{c_2} + \mathbf{m_1}.\mathbf{c_1}$.



a) Time-window for even iteration numbers b) Time-window for odd iteration numbers

Figure 5: Time-window functions in the Marchenko scheme with a smooth cosine-shaped transition zone. This transition zone has a default setting of $0.5n_\varepsilon$ samples and is within the n_ε samples.

Multiple removal in action



a) $v^+_{1,1}(t)$ b) Convergence of $\mathbf{c}_1$ c) $k^-_{1,30}(t)$

Figure 6: Creation of the event (labeled $\mathbf{c}_1$) that annihilates all the internal multiples between the first and second reflector, the artifact $\mathbf{a}_1$ is ignored in the analysis. Picture a) shows $v^+_{1,1}(t)$ for the first Marchenko iteration at sample number $ii = 200$. In b) the convergence of the maximum amplitude in $\mathbf{c}_1$ is shown as function of the iteration count. Figure c) shows the annihilated multiples in $k^-_{1,30}(t)$ after 30 iterations.

The results in Figure 6 are partial solutions of the Marchenko equations computed for time sample $ii = 200$. After applying the time window, that sets all samples in M_0 to zero beyond $200 - n_\varepsilon$, there are no internal multiple reflections present anymore in M_0. The times between 0 and sample 200 include $\mathbf{r}_1$ and $\mathbf{r}_2$, but not $\mathbf{m}_1$, see Figure 3b. In the first iteration to compute v^+_1, according to equation 8, one extra event in v^+_1 (Figure 6a event $\mathbf{c}_1$) is created to correct for the amplitude of the second reflector in v^-_1. Note that the time windows in Figure 6a appears to be around sample 80, but this is the time window applied in M_0 shifted by the correlation to negative times and time-reversed. The amplitude of this event $\mathbf{c}_1$ converges to the amplitude that can annihilate the amplitude of the first multiple. Applying the converged v^+_1 on the reflection data through equation 4, causes that all multiples arising from bounces between the first and second reflector will vanish from the data in equation 1. The scheme finishes without ever having 'seen' the multiple; from $\mathbf{r}_1$ and $\mathbf{r}_2$ alone it created an event that can attenuate all the internal multiples between these reflectors. The arrows in Figure 6c, that shows $k^-_{1,30}(t)$ and computed according to equation 6, point at the multiples that are already partly gone. The multiples are only partly removed because only a small offset-range of $\mathbf{r}_2$ is used at sample 200. Repeating the scheme for samples larger than 200 will include larger offsets of $\mathbf{r}_1$ and $\mathbf{r}_2$ and attenuate also the higher offsets for all internal multiples between $\mathbf{r}_1$ and $\mathbf{r}_2$.

For the investigation of the amplitudes of the event in v^+_1 we assume, for the sake of argument, that the reflection coefficient is a constant. The primary reflections in Figure 3b have local reflection coefficient a_1, a_2 for respectively the events labeled $\mathbf{r}_1, \mathbf{r}_2$. We consider the situation in Figure 6, for time sample 200 that only creates one extra event in v^+_1. We demonstrate that after sufficient iterations the event in v^+_1 has converged to an amplitude that can cancel the first-order multiple, and hence all higher-order multiples related to that

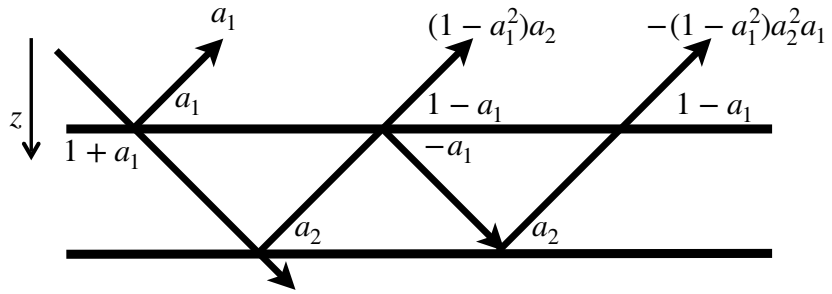


Figure 7: Sketch of the ray-paths and reflection and transmission coefficients in a three layer constant velocity and variable density model. The local reflection coefficients of the first and second reflector are a_1 and a_2 respectively.

event. For sample 200, M_0 contains only the primary reflections $\mathbf{r}_1, \mathbf{r}_2$ that have amplitude:

$$r_1^a = a_1, \tag{12}$$
$$r_2^a = (1-a_1^2)a_2. \tag{13}$$

Figure 7 is a sketch of the reflection paths and reflection and transmission coefficients for this two reflector case. According to equation 5, for the first iteration M_0 is correlated with R and integrated over the receiver coordinate. After applying the time window on M_1 only one event remains; event $\mathbf{c}_1 = \mathbf{r}_2^*.\mathbf{r}_1$ in Figure 6a with amplitude:

$$c_{1,1}^a = a_1(1-a_1^2)a_2. \tag{14}$$

The second subscript in $c_{1,1}^a$ indicates the iteration number. This event is convolved with R in the next iteration (according to equation 4) and after time windowing only one event $\mathbf{r}_1.\mathbf{c}_1$ remains at the time of the reflection of the second reflector with amplitude

$$c_{1,2}^a = a_1^2(1-a_1^2)a_2. \tag{15}$$

In each next iteration, alternating between equations 5 and 4, another multiplication with a_1 is added, in general for iteration i we have:

$$c_{1,i}^a = (a_1)^i(1-a_1^2)a_2. \tag{16}$$

Summation of all odd $c_{1,i}^a$ iterations i gives the final amplitude of the multiple annihilation event $\mathbf{c}_1$ in v_1^+. The initialization of v_1^+ is zero and the summation of the odd terms lead to:

$$\begin{aligned}\sum_{i=0}^{n_i} c_{1,1+2i}^a =& \sum_{i=0}^{n_i}(a_1)^{1+2*i}(1-a_1^2)a_2,\\ =& a_1a_2 - a_1^3a_2 + a_1^3a_2 - a_1^5a_2 + a_1^5a_2 - a_1^7a_2 + \dots \\ \approx& a_1a_2.\end{aligned} \tag{17}$$

Application of v_1^+ to the data creates multiple free data in the resulting u_1^- and is shown in Figure 6c. The first-order internal multiple from the data and the multiple annihilation

event $\mathbf{c}_1$ in Figure 6a, with amplitude $a_1 a_2$, will meet each other in time just below the first reflector. At that point in time the annihilator cancels the first-order downgoing internal multiple and with that all other related multiples. To be able to cancel the first downgoing internal multiple the annihilator must have the same amplitude as that event. The first-order multiple event $\mathbf{m}_1$ in Figure 3b has amplitude

$$m_1^a = -(1 - a_1^2) a_2^2 a_1. \tag{18}$$

After convergence of the scheme the multiple annihilator event $\mathbf{c}_1$ is convolved with the second reflector $\mathbf{r}_2$ of R in the next iteration and arrives at the same time as $\mathbf{m}_1$ and has the same amplitude as m_1^a:

$$\begin{aligned} c_1^a r_2^a = &= a_1 a_2 . r_2^a, \\ &= (1 - a_1^2) a_2^2 a_1. \end{aligned} \tag{19}$$

This result is added to the data to cancel the internal multiple at $\mathbf{m}_1$ as shown in Figure 6c and equation 1. Furthermore, convolution of $\mathbf{c}_1$ with $\mathbf{m}_1$ will create the annihilator of the second-order multiple, hence $\mathbf{c}_1$ will automatically annihilate all higher-order multiples as well.

To complete the amplitude analysis, the amplitude of the second reflector in $\bar{v}_1^-$ (from the equivalent of equation 7 for the T-MME scheme) can be computed according to equation 10 and constructed from the even amplitude terms in equation 16. The initialization of $\bar{v}_1^-$ is the time reversed shot record (DD in Algorithm 1). Summation of all even $c_{1,i}^a$ iterations i at the time of the second reflector creates the final amplitude for the second reflector in $\bar{v}_1^-$

$$\begin{aligned} a_2 =&(1 - a_1^2) a_2 + \sum_{i=1}^{n_i} a_1^{2*i} (1 - a_1^2) a_2 \\ =&a_2 - a_1^2 a_2 + a_1^2 a_2 - a_1^4 a_2 + a_1^4 a_2 - a_1^6 a_2 + \dots \\ \approx&a_2. \end{aligned} \tag{20}$$

This shows that the transmission compensated local reflectivity can be collected from v_1^- as implemented in the T-MME scheme. The approximation sign is due to a limited number of iterations in the numerical implementation.

Figure 8 is obtained in the same way as Figure 6b, but with high contrast layers. The velocity of the layers is the same as used in Figure 6, but the density contrast between the layers has been increased from a factor 3 (1000-3000) to 10 (500-5000). Compared to Figure 6b the convergence is much slower in this high contrast medium. In equation 17 the higher-order terms will have larger values in high contrast media and require more iterations for convergence.

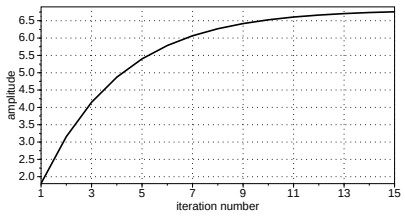


Figure 8: Convergence of the maximum amplitude of the event (labeled $\mathbf{c}_1$ in Figure 6a) that annihilates all the internal multiples between the first and second reflector in a high contrast medium.

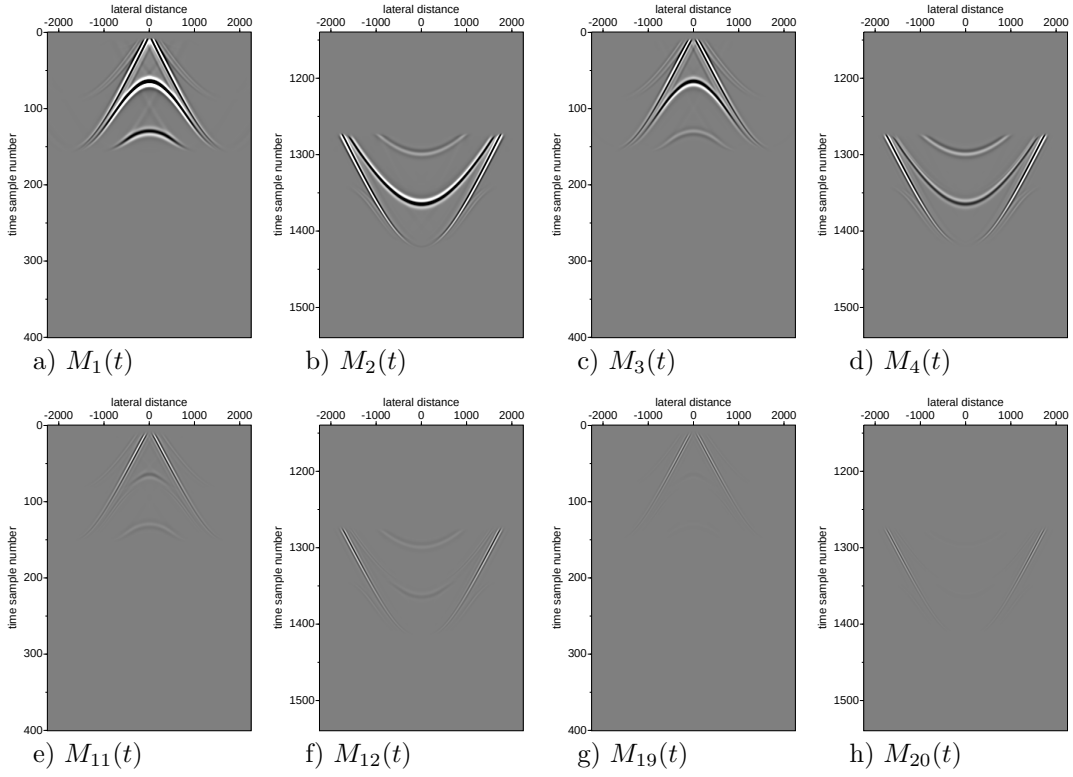


Figure 9: M_i fields for a focal time at sample $ii = 276$; the zero-offset arrival of the third reflector. All figures are plotted with the same clipping factor.

Higher iteration counts

The first few iterations for the update terms M_i are shown in Figure 9. The truncation time is chosen at sample 276 and a first-order multiple of the second layer is present in the initialization shot-record M_0 after time-truncation. For higher numbers of iterations the update terms become smaller in amplitude, indicating that the scheme converges. All the updates show the same number of events and only the amplitude of the events change during the iterations.

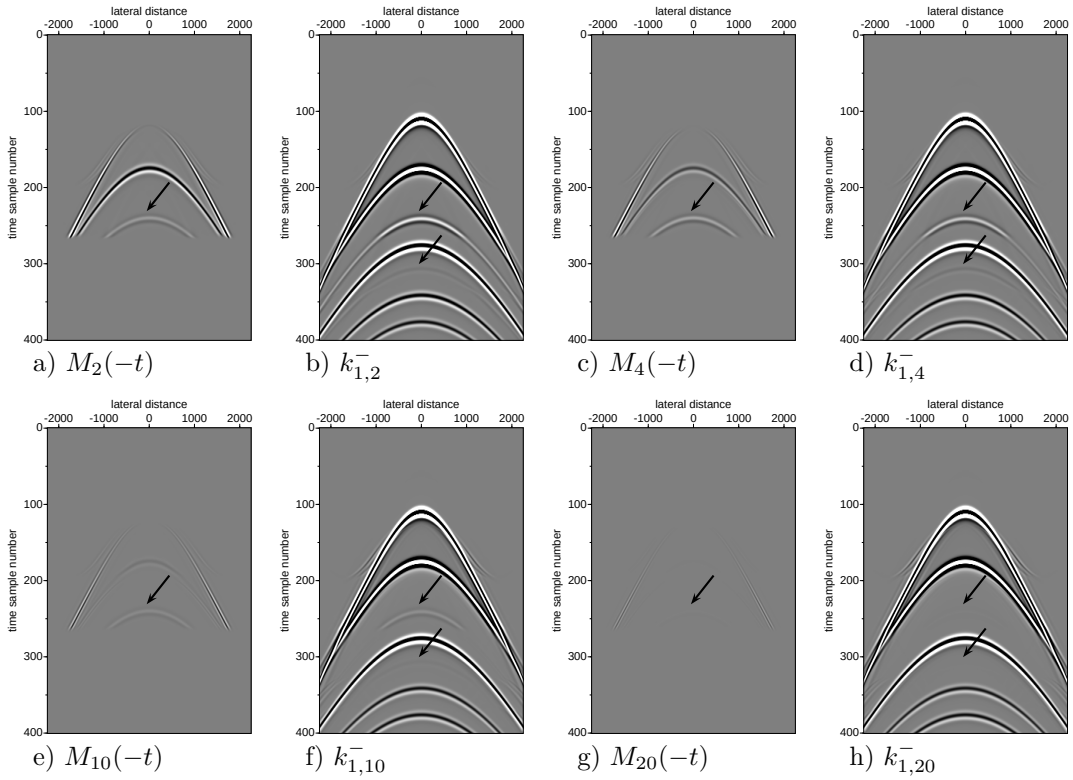


Figure 10: Updates for $k^-_{1,i}$ for a focal time at sample $t_2 = 276$ after i iterations. The arrow indicates the first and second-order internal multiple between the first and second reflector.

In the odd iterations the function $k^-_{1,i}(t)$, see equation 6, is updated with the odd $M_i(-t)$ terms and four selected iterations are shown in Figure 10. After two iterations all order multiples are predicted, but with incorrect amplitudes. In the following iterations the removal of higher-order multiples is improved because the removal of the first-order multiple improves. After 20 iterations the internal multiple events (indicated with arrows) have further attenuated and are not visible anymore; compare Figure 10b with 10h. The higher-order multiples do not have to be removed by extra events in v^+_1, but are removed automatically by removing the first-order multiple.

In Figure 10f one can observe that the first internal multiple (pointed at by the top arrow) is already attenuated beyond sample $276 - n_\varepsilon + 1$, but is not yet completely attenuated before sample $276 - n_\varepsilon$. The first $276 - n_\varepsilon$ samples belong to v^-_1, where the information on attenuation of the internal multiple is constructed, while samples from $276 - n_\varepsilon + 1$ onward belong to u^-_1, where the multiple is already attenuated. The constant-time cross-section (for all lateral positions) in $k^-_{1,i}$ at sample 276 is stored in the final output R_t at sample 276. In the v^-_1 part (between samples 1 and 276) of k^-_1 the second reflector has its local reflection coefficient as amplitude, while in the u^-_1 part (from sample 276 onward) it has its physical amplitude with two-way transmission effects.

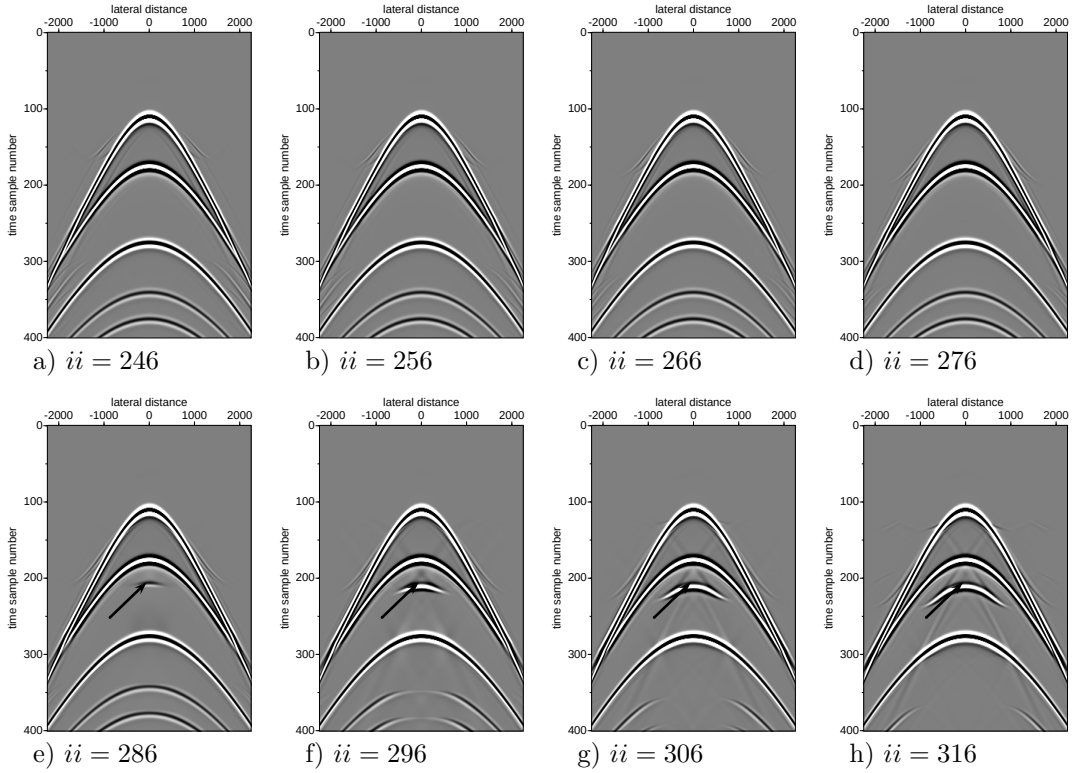


Figure 11: k_1^- after 32 iterations with different time instants $ii = 246$ to $ii = 316$ with steps of 10 samples. From each panel a constant-time cross section is selected at ii and all these cross-sections make up the multiple-free data. The arrows point to an event that compensates all internal multiples created between the second and third reflector.

Different time instances

In Figure 11 the Marchenko equations are solved for different time samples ii and it is possible to investigate how k_1^- changes for larger sample numbers. It is observed that not only at sample ii, but also before and beyond ii the events related to internal multiples are attenuated. Sample point 276 corresponds to the arrival time of the third reflector. The times in Figure 11a-11d are all before sample 276 and we do not observe a change in the number of events. However, going from sample 246 (Figure 11a) to 276 (Figure 11d) one can see that the multiple, arriving in time between the second and third reflector, gets more and more attenuated at larger and larger offsets. This also explains the success of the fast algorithm, to compute the solution in the next time sample there is only a small change needed and a few iterations are sufficient to solve for the multiple attenuation at higher offsets. When the sample time ii passes the arrival time of the third reflector, a non-physical event (pointed at with an arrow in Figure 11e-11h) appears just below the arrival time of the second reflector. This non-physical reflector is the annihilator event in v_1^+ that compensates all internal multiples created between the second and third reflector. The

cancelation of the internal multiples is observed at larger time samples (Figure 11e-11h); all internal multiples related to the third reflector are canceled out.

Figure 12a and 12b sketches of the situation where the time instant ii corresponds to a depth above or below the third reflector, respectively. The event that compensates all internal multiples related to the second reflector (green arrow in Figure 12a) coincides in traveltime with the reflection of the second reflector and also compensates for the transmission loss of the reflection from the second reflector. The internal multiples related to the third reflector are compensated by the red-arrow event (Figure 12b) which coincides with the reflection time of the third reflector. This event is also reflected by the second reflector (upward red arrow) and creates the non-physical reflection event observed in Figure 11e-11h and pointed with an arrow.

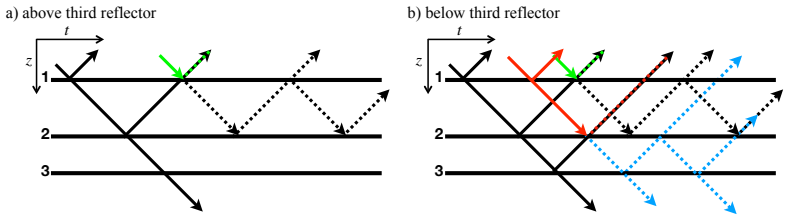


Figure 12: Compensation of internal multiples by events (coloured lines) that are created by the Marchenko method, applied for a point above (a) and below (b) the third reflector. The three reflectors are numbered from top to bottom.

Figures 13a-13d show the same pictures as in Figure 9a-d and Figures 13e-h show the same as Figures 10b,10d,10f,10h, but now with the T-MME scheme (Zhang et al., 2019). The T-MME scheme for time sample 276 includes the reflection of the third reflector, since the time window is now $276 + n_\varepsilon$. This extra reflector introduces new events in $\bar{M}_i$. In the $\bar{k}^-_{1,i}$ terms the non-physical primary, just below the second reflector, is clearly visible. After 20 iterations Figure 13h looks very similar to Figure 11f (time instant 296). The difference is that in the T-MME scheme the truncation starts at $t_2 + \varepsilon$ (sample 276+8) and the value at t_2 (time sample 276) is exactly right in v^- for the local reflection coefficient and stored in the final data output ($R_r[276]$), while in the MME scheme of Figure 11g the truncation starts at $t_2 - \varepsilon$ (sample 296-8) and the value at t_2 (time sample 296) in u^- is the correct value for the physical primary and is stored in the final data output ($R_t[296]$).

In the example for the MME scheme we have shown (in equation 20) that the reflection strength of the second reflector was modified in v^-_1 from its physical amplitude to its local reflection coefficient as amplitude. It is exactly this feature that T-MME exploits. When the Marchenko schemes reaches the arrival time of a reflector there is a decision to be made where to put the reflection of that reflector. Setting the truncation time to $t_2 - \varepsilon$ the time-instant t_2 is correctly obtained in u^-_1. Changing the truncation time from $t_2 - \varepsilon$ to $t_2 + \varepsilon$, the time-instant t_2 is correctly obtained in v^-_1 instead of in u^-_1. It is the time duration of the source wavelet that allows us to make this choice. By taking $t_2 - \varepsilon$ an error is introduced in v^-_1 and hence u^-_1 is correct at t_2, whereas by taking $t_2 + \varepsilon$ the error is in u^-_1 and v^-_1 will be correct at t_2.

The transmission compensated (T-MME) scheme retrieves primary reflections with local reflection coefficients, while in the regular (MME) scheme the primary reflections keep their

two-way reflection coefficients that include transmission losses. The local reflection retrieval of the T-MME scheme is exact for a horizontally layered medium, but in laterally varying media it is approximately true (Zhang et al., 2019). The only computational difference between the T-MME and MME schemes is the position of the time-truncation window.

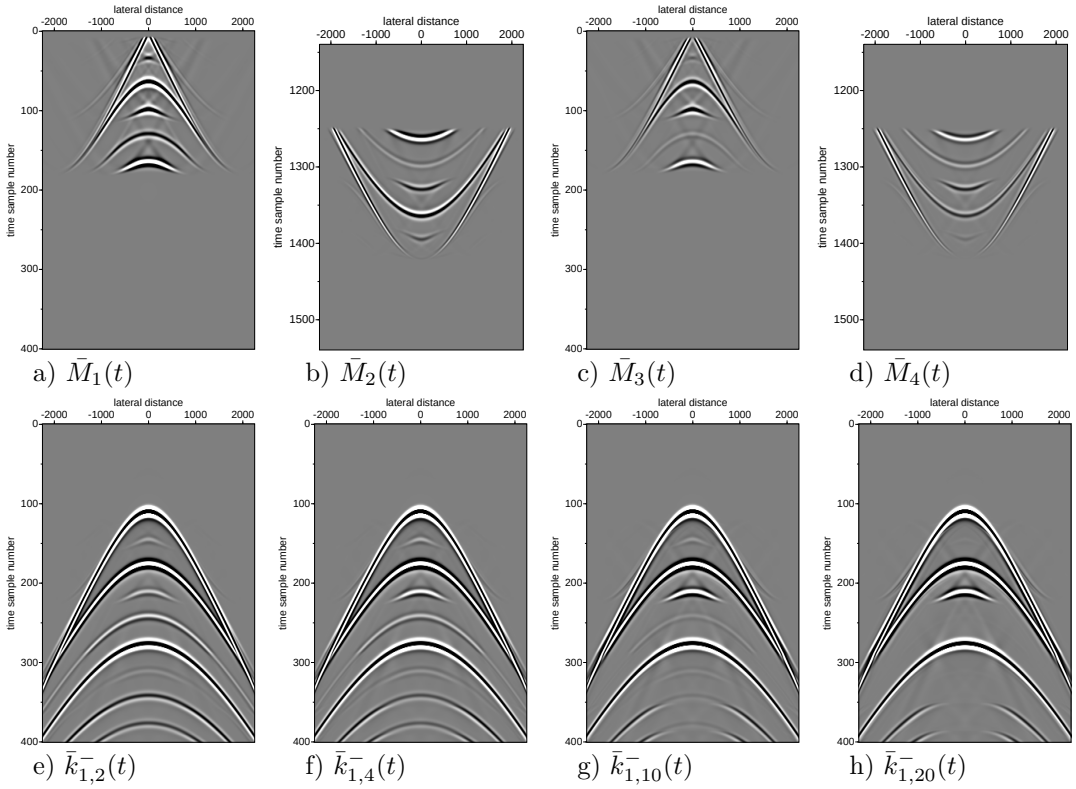


Figure 13: Panels (a)-(d) show the $\bar{M}_i$ fields for a focal time at sample $t_2 = 276$ with the transmission compensated scheme T-MME after $i = 1, 2, 3, 4$ iterations. Panels (e)-(h) show the $\bar{k}^-_i$ fields for a focal time at sample $t_2 = 276$ with the transmission compensated scheme T-MME after $i = 2, 4, 10, 20$ iterations. All figures are plotted with the same clipping factor.

CONCLUSIONS

In this paper we demonstrated step-by-step the correct and effective elimination of internal multiple reflections from the acoustic reflection response using the truncation time instant as a free parameter. With a simple four-layer model we showed that in a 1.5D model MME retrieves the primary reflections with their physical amplitude whereas T-MME retrieves the primary reflections with their local reflection coefficient. The examples also show that when a first-order internal multiple is eliminated, the whole train of multiples associated with that first-order multiple is eliminated as well. This is achieved by a single multiple annihilator that is computed from only primary reflections in the data. The compute kernels of the two schemes are identical and consist of a matrix vector product that alternates between a time-correlation and a time-convolution. Exact and efficient computation of the discrete correlations and convolutions is done with an FFT routine. The only difference between MME and T-MME is the implementation of the time truncation. MME excludes a chosen

time window before the truncation time instant and obtains the result after the scheme has converged by evaluating the equation at the truncation time instant. T-MME includes a chosen time window after that time instant and obtains the result directly from the value at the truncation time instant when the scheme has converged. For each truncation time instant the iterative scheme computes the multiple annihilators and the corresponding reflection output. The current standard implementation takes every data time-sample as a truncation time; the faster implementation can takes larger time steps for the truncation time by exploiting the finite frequency bandwidth of the data.

ACKNOWLEDGMENT

The research of Jan Thorbecke and Kees Wapenaar has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant no. 742703). We would like to thank the associate editor and 3 anonymous reviewers for their constructive comments and efforts to compile the code and reproduce the figures.

REFERENCES

Behura, J., K. Wapenaar and R. Snieder, 2014, Autofocus imaging: Image reconstruction based on inverse scattering theory: Geophysics, **79** (3), A19–A26.

Brackenhoff, J., Thorbecke, J., and Wapenaar, K., 2019, Virtual sources and receivers in the real Earth - considerations for practical applications: Journal of Geophysical Research - Solid Earth, **124** (11), 802–821.

Broggini, F., R. Snieder, and K. Wapenaar, 2014, Data-driven wave field focusing and imaging with multidimensional deconvolution: Numerical examples from reflection data with internal multiples: Geophysics, **79** (3), WA107–WA115.

Cohen, J. K. and J. W. Stockwell, 2016, CWP/SU: Seismic Un*x Release No. 43R4: an open source software package for seismic research and processing: Center for Wave Phenomena, Colorado School of Mines.

Costa Filho, C. A. da, M. Ravasi, A. Curtis, and G.A. Meles, 2014, Elastodynamic Green's function retrieval through single-sided Marchenko inverse scattering: Physical Review E, **90**, 063201.

Costa Filho, C. A. da, G.A. Meles, A. Curtis, M. Ravasi and A. Kritski, 2017, Imaging strategies using Marchenko focusing functions: 79th Annual International Meeting, European Association of Geoscientists and Engineers, Expanded Abstracts, Tu P9 15.

Dukalski, M.S., and K. de Vos, 2017, Marchenko inversion in a strong scattering regime including surface-related multiples: Geophysical Journal International, **212** (2), 760–776.

Jia, X., Guitton, A. and Snieder, R., 2018, A practical implementation of subsalt Marchenko imaging with a Gulf of Mexico data set: Geophysics, **83** (5), S409–S419.

Lomas, A.,and A. Curtis, 2019, An introduction to Marchenko methods for imaging: Geophysics, **84** (2), F35–F45.

Lomas, A., S. Singh, and A. Curtis, 2020, Imaging vertical structures using Marchenko methods with vertical seismic-profile data: Geophysics, **85** (2), S103–S113.

Matias, M. A., R. Pestana, and J. van der Neut, 2018 Marchenko imaging by unidimensional deconvolution: Geophysical Prospecting, **66** (9), 1653–1666.

Meles, G.A., K. Löer, M. Ravasi, A. Curtis and C.A. da Costa Filho, 2015, Internal multiple prediction and removal using Marchenko autofocusing and seismic interferometry: Geophysics, **80** (1), A7–A11.

Meles, G.A., C.A. da Costa Filho and A. Curtis, 2017, Synthesising singly-scattered waves (primaries) from multiply-scattered data: 79th Annual International Meeting, European Association of Geoscientists and Engineers, Expanded Abstracts, Tu P4 11.

Meles, G.A., K. Wapenaar, and J. Thorbecke, 2018, Virtual plane-wave imaging via Marchenko redatuming: Geophysical Journal International, **214** (1), 508–519.

Meles, G.A., L. Zhang, J. Thorbecke, K. Wapenaar and E. Slob, 2020 Data-driven retrieval of primary plane-wave responses: Geophysical Prospecting, **68**, 1834–1846.

Mildner C., F. Broggini, K de Vos, and J.O.A. Robertsson, 2019 Accurate source wavelet estimation using Marchenko focusing functions: Geophysics, **84** (6), Q73–Q88.

Pereira, R., Ramzy, M., Griscenco, P., Huard, B., Huang, H., Cypriano, L. and Khalil, A., 2019, Internal multiple attenuation for OBN data with overburden/target separation: 89nd Annual International Meeting, Society of Exploration Geophysicists, Expanded Abstracts, 4520–4524.

Ravasi, M., I Vasconcelos, A. Kritski, A. Curtis, C.A. da Costa Filho and G.A. Meles, 2016, Target-oriented Marchenko imaging of a North Sea field: Geophysical Journal International, **205** (1), 99–104.

Ravasi, M., 2017, Rayleigh-Marchenko redatuming for target-oriented, true-amplitude imaging: Geophysics, **82** (6), S439–S452.

Ravasi, M., and I. Vasconcelos, 2020, PyLops - A linear-operator Python library for scalable algebra and optimization: SoftwareX, **11**, 100361.

Singh, S., R. Snieder, J. Behura, J. van der Neut, K. Wapenaar and E. Slob, 2015, Marchenko imaging: Imaging with primaries, internal multiples, and free-surface multiples: Geophysics, **80** (5), S165–S174.

Singh, S., R. Snieder, J. van der Neut, J. Thorbecke, E. Slob, and K. Wapenaar, 2017, Accounting for free-surface multiples in Marchenko imaging: Geophysics, **82** (1), R19–R30.

Slob, E., K. Wapenaar, F. Broggini and R. Snieder, 2014, Seismic reflector imaging using internal multiples with Marchenko-type equations: Geophysics, **79** (2),S63–S76.

Slob, E., 2016, Green's function retrieval and Marchenko imaging in a dissipative acoustic medium: Physical Review Letters, **116** (16), 164301.

Sripanich, Y., Vasconcelos, I., and Wapenaar, K., 2019, Velocity-independent Marchenko focusing in time- and depth-imaging domains for media with mild lateral heterogeneity: Geophysics, **84** (6), Q57–Q72.

Staring, M., R. Pereira, H. Douma, J. van der Neut, and K. Wapenaar, 2018, Source-receiver Marchenko redatuming on field data using an adaptive double-focusing method: Geophysics, **83** (6), S570–S590.

Thorbecke, J. and D. Draganov, 2011, Finite-difference modeling experiments for seismic interferometry: Geophysics, **76** (6), H1–H18.

Thorbecke, J., E. Slob, J. Brackenhoff, J. van der Neut, and K. Wapenaar, 2017, Implementation of the Marchenko method: Geophysics, **82** (6), WB29–WB45.

Thorbecke, J., and J. Brackenhoff, 2020, OpenSource code for Finite Difference, Marchenko and utilities: https://github.com/JanThorbecke/OpenSource, doi 10.5281/zenodo.4271598.

Van der Neut, J., I. Vasconcelos and K. Wapenaar, 2015b, On Green's function retrieval by iterative substitution of the coupled Marchenko equations: Geophysical Journal International, **203** (2), 792–813.

Van der Neut, J., and K. Wapenaar, 2016, Adaptive overburden elimination with the multidimensional Marchenko equation: Geophysics, **81** (5), T265–T284.

Verschuur, E., A. Berkhout, and K. Wapenaar, 1992, Adaptive surface-related multiple elimination: Geophysics, **57** (9), 1166–1177.

Wapenaar, K., 2014, Single-sided Marchenko focusing of compressional and shear waves: Physical Review E, **90** (6), 063202.

Wapenaar, K., J. Thorbecke, J. van der Neut, F. Broggini, E. Slob and R. Snieder, 2014a, Green's function retrieval from reflection data, in absence of a receiver at the virtual source position: The Journal of the Acoustical Society of America, **135** (5), 2847–2861.

Wapenaar, K., J. Thorbecke, J. van der Neut, F. Broggini, E. Slob and R. Snieder, 2014b, Marchenko imaging: Geophysics, **79** (3), WA39–WA57.

Wapenaar, K., J. Thorbecke, J. van der Neut, E. Slob and R. Snieder, 2017, Virtual sources and their responses, Part II: Data-driven single-sided focusing: Geophysical Prospecting, **65** (6), 1430–1451.

Wapenaar, K., and van IJsseldijk, J., 2020, Discrete representations for Marchenko imaging of imperfectly sampled data: Geophysics, **85** (2), A1–A5.

Zhang, L., and M. Staring, 2018, Marchenko scheme based internal multiple reflection elimination in acoustic wavefield: Journal of Applied Geophysics, **159** (9), 429–433.

Zhang, L., and E. Slob, 2019, Free-surface and internal multiple elimination in one step without adaptive subtraction: Geophysics, **84** (1), A7–A11.

Zhang, L., J. Thorbecke, K. Wapenaar, and E. Slob, 2019, Transmission compensated primary reflection retrieval in data domain and consequences for imaging: Geophysics, **84** (4), Q27–Q36.

Zhang, L., and E. Slob, 2020a, A field data example of Marchenko multiple elimination: Geophysics, **85** (2), S65–S70.

Zhang, L., and E. Slob, 2020b, A fast algorithm for multiple elimination and transmission compensation in primary reflections: Geophysical Journal International, **221** (1), 371–377.

Zhang, L., and E. Slob, 2020c, Marchenko multiple elimination of a laboratory example: Geophysical Journal International, **221** (2), 1138–1144.