https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

# Effectiveness and computational efficiency of absorbing boundary conditions for full-waveform inversion

Daiane Iglesia Dolci[1], Felipe A. G. Silva [2], Pedro S. Peixoto[2], and Ernani V. Volpe[1]

[1]Department of Mechanical Engineering, Escola Politécnica, University of São Paulo. Av. Prof. Mello Moraes, 2231, São Paulo, SP, 05508-030, Brazil.
[2]Department of Applied Mathematics, Institute of Mathematics and Statistics, University of São Paulo. Rua do Matão, 1010, São Paulo, SP, 05508-090, Brazil.

**Correspondence:** Daiane Iglesia Dolci (dolci@usp.br)

**Abstract.** Full-Waveform Inversion (FWI) is a high-resolution numerical technique for seismic waves used to estimate the physical characteristics of a subsurface region. The continuous problem involves solving an inverse problem on an infinite domain, which is impractical from a computational perspective. In limited area models, absorbing boundaries conditions (ABCs) are usually imposed, to avoid wave reflections. Several relevant ABCs have been proposed, with extensive literature on their effectiveness on the direct wave problem. Here, we investigate and compare the theoretical and computational characteristics of several ABCs in the full inverse problem. After a brief review of the most widely used ABCs, we derive their formulations in their respective adjoint problems. The different ABCs are implemented in a highly optimized domain specific language (DLS) computational framework, Devito, which targets seismic modeling problems. We evaluate the effectiveness, computational efficiency and memory requirements of the ABC methods, considering from simple models to realistic ones. Our findings reveal that, even though the popular Perfectly Matching Layers (PMLs) are effective on avoiding wave reflections on the boundaries, they can be computationally more demanding than less used Hybrid ABCs. We show here that a proposed Hybrid ABC formulation, with nested Higdon's boundary conditions, is the most cost-effective method among the methods considered here, being as effective, or more, as PML and other schemes, but being computationally more efficient.

*Keywords:* full-waveform inversion, seismic waves, absorbing boundary conditions, domain specific language

## 1 Introduction

Firstly presented for acoustic waves (Tarantola, 1984), and later extended for the elastic (Tarantola, 1986; Mora, 1987) and viscoelastic cases (Tarantola, 1988), Full Waveform Inversion (FWI) is a high-resolution seismic technique used to estimate the psychical parameters in a subsurface region. It is a wave-equation based technique that searches for an optimal match between real and computed data. The former is recorded by receivers in the field, whereas the latter consists of computed estimates of propagated waves emitted by a specified wave source. The observed data at the receivers is subject to influences of the subsurface medium while waves propagate from the source. Synthetic data can be generated by propagating the source waves in an estimated medium, and, therefore, the minimization of the differences between the observed and synthetic data at the receivers is a methodology to seek the medium properties of a region. The data difference is traditionally measured by a

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

least-square misfit function (Tarantola, 1984), also referred to as objective functional. The search for a minimum of the misfit function can be performed by a gradient based optimization technique (Mora, 1987). An efficient means of computing the gradient is the adjoint method (Tarantola, 1984; A. Fichtner, 2006; Fichtner, 2010). This approach is characterized by being reverse in time, where the difference between the observed and synthetic data is back propagated in time from the receivers to the source of the waves. The back propagation requires saving the wave equation solution in every computational time step, thus meaning a high memory usage to solve a FWI problem. In addition, FWI has a high computational cost, due to the size of the systems to be solved, also due to the misfit minimization process, which may demand a substantial number of iterations to achieve satisfactory results (A. Fichtner, 2006; Virieux and Operto, 2009).

In computational procedures, the forward/adjoint waves are propagated in a limited region, which is different from the real case where wave propagation occurs in an unlimited medium. On limited domains, the computational boundaries can allow spurious wave reflections to appear, which means that nonphysical information will eventually reach receivers and influence the misfit function (Gao et al., 2017). To tackle this problem, the so-called Absorbing Boundary Conditions (ABC) have been a usual practice in FWI, as a means of reducing spurious boundary reflections.

In essence, ABCs entail adding terms to both the forward and adjoint operators, and/or a set of additional equations, to be solved together with the original ones. It may also require extending the computational domain to accommodate for an absorbing layer. The performance of ABC methods is generally assessed for the forward problem (Gao et al., 2017; Liu and Sen, 2010, 2012; Grote and Sim, 2010). Such analysis is certainly relevant for FWI, since the forward problem constitutes an expressive part of it, and it is essential to guarantee a good approximation of medium properties. However, the overall impact of ABCs on the full-waveform inversion problem, from the perspective of computational cost-effectiveness and efficiency, is still widely debated in the literature (Gao et al., 2017).

This work proposes to evaluate several relevant ABCs, as applied in the context of FWI problems, while also investigating the ABCs effects on the adjoint wave equation. The analyses are carried out in a highly optimized software, namely Devito (Louboutin et al., 2019; Luporini et al., 2018), which provides a domain-specific Language (DSL) and an optimized code generation framework, for the design of finite difference kernels. In Devito, Sochaki's type of Damping Boundary Layer (Sochacki et al., 1987) is the default method to reduce the spurious reflections. The advantage of such damping method is the ease of implementation, since it only requires one to add a single term to the acoustic wave equation, and an extension of the computational domain. However, it can be less effective than other ABCs, sometimes requiring larger domain extensions. More popular, the so-called Perfectly Matching Layers (PML) have been widely used in FWI (Abubakar et al., 2009; Asnaashari et al., 2012; Aghamiry et al., 2019; Ben-Hadj-Ali et al., 2011). The PML require the introduction of auxiliary variables and equations into the problem, as well as extension of the computational domain. Those features make it more computationally demanding, but they are usually more effective in avoiding wave reflection at boundaries. An interesting solution to avoid the added cost of auxiliary variables, while also preserving method effectiveness, is the use of Hybrid schemes (HABC) (Liu and Sen, 2010, 2012). In such hybrid methods, domain extensions are still required, but no additional variables, nor equations, are needed.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

From the perspective of computational development, this work contributes by implementing further options of ABCs in Devito. Furthermore, we present a new HABC approach based on the Higdon method (Higdon, 1986, 1987), and the analyses of several ABCs, as applied to adjoint equations. The analyses are carried out for two types of ABCs, namely, *Sponge Layers* and *Hybrid Absorbing Boundary Conditions* (HABC). In the former group, we highlight the Sochaki's type of Damping Boundary Layer (Sochacki et al., 1987), the Perfectly Matched Layer (PML) (Grote and Sim, 2010) and Convolutional Perfectly Matched Layer (CPML) (Pasalic and McGarry, 2010). Whereas for the latter, the approach HABC-A1 (Liu and Sen, 2010) and the HABC-Higdon, first presented in this current work, are analyzed.

The ABC analyses are performed with heterogeneous acoustic velocity models, including realist models such as Marmousi (Martin et al., 2006) and a cut of 2D SEG/EAGE salt (Aminzadeh and Brac, 1997). Finally, this work has the objective of proposing an ABC method that combines the effectiveness in decreasing spurious boundary reflections with reduced computational cost and memory usage.

This work is organized as follows. Section 2 describes the mathematical framework of an FWI problem: misfit function, forward wave equation, adjoint wave equation and gradient of misfit function. Section 3 makes a conceptual review of the ABCs methods in the forward wave equation. Next, section 4 shows the algebraic development to obtain adjoint wave equation with ABCs methods. The computational framework adopted in this work is presented in section 5, including the main aspects of the Devito software, machine configurations and libraries tools used in the computational simulations. Results of the ABCs' performance in the forward and adjoint problem are presented in section 6. Section 7 presents FWI results with the employment of PML, HABC-Higdon and Damping methods. Finally, section 8 presents the main conclusions of the current work.

## 2 The FWI Problem

In essence, FWI consists in a local optimization, where the goal is to minimize the misfit between observed and predicted seismograms' data. On following Tarantola (1984), the misfit function can be measured by the $L^2$ norm, which maybe written as, in a continuous space,

$$I(m) \equiv \frac{1}{2} \int_{\tau} \int_{\Omega} \left( u(m,\mathbf{x},t) - u^{obs}(m,\mathbf{x},t) \right)^2 \boldsymbol{\delta}(\mathbf{x}-\check{\mathbf{x}}) \, \mathrm{d}\mathcal{V}\mathrm{d}t. \tag{1}$$

The data functions, $u = u(\mathbf{x},t)$ and $u^{obs} = u^{obs}(m,\mathbf{x},t)$, are respectively the predicted and observed data, both recorded at a finite set of receivers, located at the point positions $\check{\mathbf{x}} \in \Omega_0$, in a time interval $\tau \equiv [t_0, t_f] \subset \mathbb{R}$, where $t_0$ is the initial time and $t_f$ is the final time. The term $\boldsymbol{\delta}(\mathbf{x}-\check{\mathbf{x}})$ is the delta Dirac function to model the receiver point positions. The spatial domain of interest (usually two- or three-dimensional) is set as $\Omega_0$ and referred here as physical domain.

### 2.1 Wave equation

The predicted data, $u(\mathbf{m},\mathbf{x},t)$, is modeled here by an acoustic wave equation,

$$m(\mathbf{x})u_{tt}(\mathbf{x},t) - \nabla^2 u(\mathbf{x},t) = f(\mathbf{x},t), \tag{2}$$

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

where $u_{tt}$ represents the second partial derivative with respect to time $t$ and $\nabla^2(\cdot)$ represents the Laplacian operator with respect to $\mathbf{x} \in \Omega_0$. The variable coefficient $m(\mathbf{x}) : \Omega_0 \to \mathbb{R}$ is such that $m(\mathbf{x}) = \dfrac{1}{c^2(\mathbf{x})}$, where $c(\mathbf{x}) : \Omega_0 \to \mathbb{R}$ is the pressure wave ($P$-wave) velocity, which is assumed to be piecewise-constant and positive. The external force term $f(\mathbf{x}, t) : \Omega_0 \to \mathbb{R}$, models the source of waves and is usually described by a Ricker Wavelet (Wiki, 2019).

The acoustic wave equation should satisfy the homogeneous initial conditions, given by $u(\mathbf{x}, 0) = 0 = u_t(\mathbf{x}, 0) = 0$. Furthermore, for computational simulations, it is necessary to bound the domain $\Omega_0$. A limited area domain is illustrated in Figure 1 (a), where the limitation of type $\Omega_0 = [x_I, x_F] \times [z_I, z_F]$ is considered. The boundaries $\partial\Omega_i$ with $i = 1, 2, 3$ are here referred to as truncated boundaries, and satisfy a null-Dirichlet boundary condition $u(\mathbf{x}, t) = 0$. Finally, the boundary $\Omega_4$ satisfies the null-Neumann $\nabla u(\mathbf{x}, t) \cdot \mathbf{n} = 0$ (free surface) boundary condition, where $\mathbf{n}$ represents the outward normal (with respect to $\partial\Omega_4$) unit vector.

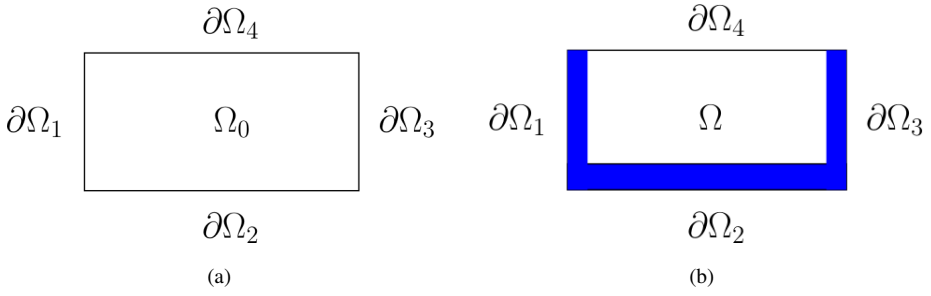



**Figure 1.** (a) Limited Domain Representation, with $\Omega_0 = [x_I, x_F] \times [z_I, z_F]$. (b) Extended domain representation, $\Omega = [x_I - L_x, x_F + L_x] \times [z_I, z_F + L_z]$, with absorption or sponge regions (of lengths $L_x$, $L_z$) highlighted in blue. $\partial\Omega_i$, $i = 1, 2, 3, 4$, indicates the outmost boundaries of the full domain.

### 2.2 Gradient of misfit function

As mentioned in the first part of this section, in FWI the goal is to minimize the misfit function, which can be measured by eq. (1). Typically, this minimization is carried out by employing a local optimization method. Thus, it is necessary to obtain the gradient, $\nabla_m I(m)$, which may be computed efficiently by the adjoint method (Plessix, 2006). The adjoint-based gradient is achieved by using an augmented functional, also referred as Lagrangian functional. In the current case, it is given by,

$$\mathcal{L}(u, u^\dagger, m) = I(m) - \int_\tau \int_{\Omega_0} u^\dagger \cdot \left(m u_{tt} - \nabla^2 u - f\right) \mathrm{d}\mathcal{V}\mathrm{d}t, \tag{3}$$

where $u = u(\mathbf{x}, t)$, $m = m(\mathbf{x})$, and $u^\dagger = u^\dagger(\mathbf{x}, t)$ is the Lagrange multiplier.

On a local minimum, the gradient of $\mathcal{L}$ with respect to $u$, $u^\dagger$ and $m$ should vanish. The gradient of $\mathcal{L}(u, u^\dagger, m)$, with respect to $m$, can be computed by

$$\lim_{\epsilon \to 0} \frac{\mathcal{L}(u, u^\dagger, m + \epsilon m') - \mathcal{L}(u, u^\dagger, m)}{\epsilon} = \nabla_m [\mathcal{L}(u, u^\dagger, m)] m' == \nabla_m [I(m)] m' = \int_\tau \int_{\Omega_0} m'\, u \cdot u^\dagger_{tt}\, \mathrm{d}\mathcal{V}\mathrm{d}t, \tag{4}$$

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

where $m'$ is a perturbation of the parameter $m$.

### 2.3 Adjoint Equation

In eq. (4) we observe that the gradient $\nabla_m I(m)$ depends on the adjoint variable $u^\dagger$, that is computed by solving the adjoint wave equation:

$$mu^\dagger_{tt} - \nabla^2 u^\dagger = (u - u^{obs})\boldsymbol{\delta}(\mathbf{x} - \check{\mathbf{x}}). \tag{5}$$

For the domain $\Omega_0$ illustrated in Figure 1 (a), the adjoint wave equation must satisfy the boundary conditions: $u(\mathbf{x},t) = 0$ for $\mathbf{x} \in \partial\Omega_i$ with $i = 1,2,3$; and $\nabla u(\mathbf{x},t) \cdot \mathbf{n} = 0$ for $\mathbf{x} \in \partial\Omega_4$. The adjoint wave equation is reverse in time. This way, the initial condition is given by $u^\dagger(\mathbf{x},t_f) = 0$.

The adjoint wave equation is obtained by carrying out the gradient of $\mathcal{L}(u,u^\dagger,m)$ with respect to the state variable $u$. Details as to the method to obtain it can be found in the works of Plessix (2006) and A. Fichtner (2006).

## 3 ABC's in Forward Problem

### 3.1 Domain Extension

For all the methods that we are described here, we consider an extension of the spatial domain given by $\Omega = [x_I - L_x, x_F + L_x] \times [z_I, z_F + L_z]$, in which an *absorption region* or *sponge layer* is added to the original spatial domain, $\Omega_0 = [x_I, x_F] \times [z_I, z_F]$. The *absorption region* is composed by two bands of length $L_x$ at the beginning and end of the domain in the direction $x$ and of a band of length $L_z$ at the end of the domain in the $z$ direction. Again, $\partial\Omega$ denotes the boundary of $\Omega$. Figure 1b shows the extended domain $\Omega$, with the absorption region highlighted in blue. This kind of extension represent the typical configuration for seismic problems.

### 3.2 Damping

The method called *Damping* has been proposed initially by Sochaki Sochacki et al. (1987) and it is a very simple way to reduce the spurious reflections of wave propagation in limited domains. The basic idea is to extend the original domain, by adding a sponge layer to it, like the one in Figure 1b, and then to introduce a damping term into the original wave equation (2), such that it only affects the added layer. The resulting damped acoustic equation is given by,

$$m(\mathbf{x})u_{tt}(\mathbf{x},t) + \zeta(\mathbf{x})u_t(\mathbf{x},t) - \nabla^2 u(\mathbf{x},t) = f(\mathbf{x},t), \tag{6}$$

where the acoustic wave equation (2) has been modified by the introduction of the damping term $\zeta(\mathbf{x})u_t(\mathbf{x},t)$, with $\zeta(\mathbf{x})$ being nonzero only within the absorption region. That is, it should grow smoothly across the absorption bands from zero to its maximum at the outermost boundary. One may still impose the same initial and boundary conditions defined in the previous section.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

Sochacki et al. (1987) proposed various alternatives for the damping function, $\zeta(\mathbf{x})$, including linear, cubic or exponential forms. In general, all of them share a similar characteristic: they vanish identically throughout the interior domain $\Omega_0$, while growing within the added bands from zero toward the outer boundary $\partial\Omega$. We define the pair of functions $\zeta_1(\mathbf{x})$ and $\zeta_2(\mathbf{x})$ as, respectively:

$$\zeta_1(\mathbf{x}) = \begin{cases} 0, \\ \qquad \text{if } x \in (x_I, x_F), \\ \bar{\zeta}_1(\mathbf{x}) \left( \dfrac{|x - x_I|}{L_x} - \dfrac{1}{2\pi} \sin\left( \dfrac{2\pi |x - x_I|}{L_x} \right) \right), \\ \qquad \text{if } x_I - L_x \leq x \leq x_I, \\ \bar{\zeta}_1(\mathbf{x}) \left( \dfrac{|x - x_F|}{L_x} - \dfrac{1}{2\pi} \sin\left( \dfrac{2\pi |x - x_F|}{L_x} \right) \right), \\ \qquad \text{if } x_F \leq x \leq x_F + L_x, \end{cases} \tag{7}$$

$$\zeta_2(\mathbf{x}) = \begin{cases} 0, \\ \qquad \text{if } z \in (z_I, z_F), \\ \bar{\zeta}_2(\mathbf{x}) \left( \dfrac{|z - z_F|}{L_z} - \dfrac{1}{2\pi} \sin\left( \dfrac{2\pi |z - z_F|}{L_z} \right) \right), \\ \qquad \text{if } z_F \leq z \leq z_F + L_z, \end{cases} \tag{8}$$

so that the actual damping function $\zeta(\mathbf{x})$ is given by:

$$\zeta(\mathbf{x}) = \frac{1}{c_{max}} \left( \frac{\zeta_1(\mathbf{x})}{\Delta x} + \frac{\zeta_2(\mathbf{x})}{\Delta z} \right), \tag{9}$$

where $c_{max}$ denotes the maximum velocity of propagation of $c(\mathbf{x})$, $\Delta x$ and $\Delta z$ are the discrete cell sizes of the spatial domain, respectively in the $x$ and $z$ directions.

### 3.3 Perfectly Matched Layer

The method called *Perfectly Matched Layer* (PML) has several formulations in the literature, considering the acoustic (second order equation or first order system formulations) and elastic cases. Like the Damping method, the PML is widely used in seismic problems, particularly due to its efficacy in reducing spurious reflections in limited domains, being more effective than the Damping method. The formulation we present here has been proposed by Grote and Sim (2010) for the second order form of wave equation m second order equation.

The reasoning is similar to the Damping method, in that sponge layers extend the original domain, like those in Figure 1b. Additional terms are also introduced into the original wave equation (2), which only affect the sponge layers, but now there are two of them, and they have their own evolution equations.

The two auxiliary functions provide adequate damping of wave reflections by using similar terms to those of the Damping method. The design of the method is such that it would ideally suppress all reflections in a continuous setting. However, some reflections may remain for a finite difference discretization, although strongly attenuated.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

The full set of equations for the acoustic wave propagation with PML, along with the auxiliary functions, is given by:

$$m(\mathbf{x})u_{tt}(\mathbf{x},t) + (\zeta_1(\mathbf{x}) + \zeta_2(\mathbf{x}))u_t(\mathbf{x},t)) + \zeta_1(\mathbf{x})\zeta_2(\mathbf{x})u(\mathbf{x},t) = \nabla^2 u(\mathbf{x},t) + \phi_{1,x}(\mathbf{x},t) + \phi_{2,z}(\mathbf{x},t) + f(\mathbf{x},t), \tag{10}$$

$$\phi_{1,t}(\mathbf{x},t) = -\zeta_1(\mathbf{x})\phi_1(\mathbf{x},t) + c^2(\mathbf{x})(\zeta_2(\mathbf{x}) - \zeta_1(\mathbf{x}))u_x(\mathbf{x},t), \tag{11}$$

$$\phi_{2,t}(\mathbf{x},t) = -\zeta_2(\mathbf{x})\phi_2(\mathbf{x},t) + c^2(\mathbf{x})(\zeta_1(\mathbf{x}) - \zeta_2(\mathbf{x}))u_z(\mathbf{x},t). \tag{12}$$

Here, $\phi_1(\mathbf{x})$ and $\phi_2(\mathbf{x})$ represent the auxiliary variables, which are nonzero only in the absorption region. The notation $\phi_{i,t}$ indicates partial derivative of $\phi_i$, $i = 1, 2$, with respect to the variable $t$, and similarly for the variables $x$ and $z$, respectively, with $\phi_{i,x}$ and $\phi_{i,z}$. The damping functions $\zeta_1(\mathbf{x})$ and $\zeta_2(\mathbf{x})$ are defined as in the Damping method. The auxiliary functions will also be kept at zero over all the outer boundary of $\Omega$.

On discretizing the PML equations, we stagger[1] spatial variables for the auxiliary functions $\phi_1$ in the $x$ direction, and stagger $\phi_2$ in the $z$ direction, as in Grote and Sim (2010). This staggering is convenient, considering the centered discretization adopted here for the partial derivatives of those functions. As a result of this, $u$ must be staggered in the directions of the partial derivatives in the evolution equations of $\phi_1$ and $\phi_2$. Conversely, $\phi_1$ and $\phi_2$ should be staggered in the directions of the partial derivatives in the evolution equation of $u$.

Moreover, some variables are also staggered in time, thus being defined at intermediary time instants. As a final result, the variable $u(\mathbf{x},t)$ is taken as *non-staggered* (co-located) in space, whereas $\phi_1(\mathbf{x},t)$ and $\phi_2(\mathbf{x},t)$ are *staggered*, the functions $\zeta_1(\mathbf{x})$, $\zeta_2(\mathbf{x})$, $c(\mathbf{x})$ and $f(\mathbf{x},t)$ are *staggered* in equation (10) and *non-staggered* in the equations (11) and (12), when they appear in those equations. Therefore, when updating $u(\mathbf{x},t)$, we employ averages of neighboring values of $\phi_1(\mathbf{x},t)$ and $\phi_2(\mathbf{x},t)$, so that we have them on the non-staggered grid. On the other hand, when updating $\phi_1(\mathbf{x},t)$ and $\phi_2(\mathbf{x},t)$ we average the neighboring values of $u(\mathbf{x},t)$, to define it on the staggered grid.

### 3.4 Convolutional Perfectly Matched Layer

Although the PML method is usually very efficient for reducing boundary reflections, there are situations, such as in the presence of grazing waves, in which it is less effective. The *Convolutional Perfectly Matched Layer* (CPML) has been proposed as an improvement over PML, which should reduce late time low-frequency wave reflections and provide better absorption of grazing waves. In the case of the acoustic wave equation, the CPML is generally derived for the first order set of PDEs, but here we adopt the formulation proposed by Pasalic and McGarry (2010), which was designed for the second order form of the wave equation.

The rationale is similar to the PML, in that one extends the original domain, by adding a sponge layer to it (Fig. 1b). However, now one introduces four auxiliary functions into the original wave equation (2). These functions also have their own evolution equations, and can only affect the absorbing layer, just as in the previous approach.

[1] Staggering here means we allow the placement of variables to be dislocated by half grid-size, or half temporal-step, in such a way that different variables maybe placed in slightly different grid positions or time instants.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

The four auxiliary functions should provide adequate damping of wave reflections, by using similar terms to those of the PML. They consist of weighed-combinations of the displacement and the auxiliary functions, themselves. By design, the method should ideally suppress all reflections in a continuous setting, including those situations where the PML method fails.

The main equation reads

$$m(\mathbf{x})u_{tt}(\mathbf{x},t) = \nabla^2 u(\mathbf{x},t) + \psi_{1,x}(\mathbf{x},t) + \psi_{2,z}(\mathbf{x},t) + \phi_1(\mathbf{x},t) + \phi_2(\mathbf{x},t) + f(\mathbf{x},t), \tag{13}$$

and the auxiliary functions are updated by discrete in time relations (from time $t_n$ advancing a time-step size of $\Delta t$ leading to $t_{n+1}$):

$$\psi_1(\mathbf{x},t_{n+1}) = a_1(\mathbf{x})\psi_1(\mathbf{x},t_n) + b_1(x,z)u_x(\mathbf{x},t_{n+1}), \tag{14}$$

$$\psi_2(\mathbf{x},t_{n+1}) = a_2(\mathbf{x})\psi_2(\mathbf{x},t_n) + b_2(\mathbf{x})u_z(\mathbf{x},t_{n+1}), \tag{15}$$

$$\phi_1(\mathbf{x},t_{n+1}) = a_1(\mathbf{x})\phi_1(\mathbf{x},t_n) + b_1(\mathbf{x})\left[u_{xx}(\mathbf{x},t_{n+1}) + \psi_{1,x}(\mathbf{x},t_{n+1})\right], \tag{16}$$

$$\phi_2(\mathbf{x},t_{n+1}) = a_2(\mathbf{x})\phi_2(\mathbf{x},t_n) + b_2(\mathbf{x})\left[u_{zz}(\mathbf{x},t_{n+1}) + \psi_{2,z}(\mathbf{x},t_{n+1})\right], \tag{17}$$

where we have again used the notation for partial derivatives with double sub-index, as in $\psi_{2,z}$ meaning that the second component of $\psi$ is differentiated with respect to the $z$ variable. The weighting factors for the auxiliary functions are given by

$$a_1(\mathbf{x}) = e^{-[\zeta_1(\mathbf{x})+\alpha_1]\Delta t}, \quad b_1(\mathbf{x}) = \frac{\zeta_1(\mathbf{x})\left[a_1(\mathbf{x})-1\right]}{\left[\zeta_1(\mathbf{x})+\alpha_1\right]}, \tag{18}$$

$$a_2(\mathbf{x}) = e^{-[\zeta_2(\mathbf{x})+\alpha_2]\Delta t}, \quad b_2(\mathbf{x}) = \frac{\zeta_2(\mathbf{x})\left[a_2(\mathbf{x})-1\right]}{\left[\zeta_2(\mathbf{x})+\alpha_2\right]}. \tag{19}$$

The four auxiliary functions are only nonzero within the absorption region, while the functions $\zeta_1(\mathbf{x})$ and $\zeta_2(\mathbf{x})$ are defined as in the Damping method. The constants $\alpha_1, \alpha_2 \in \mathbb{R}$ can be chosen according to the problem.

### 3.5 Hybrid Absorbing Boundary Condition

The last class of ABCs to be discussed here is termed *Hybrid Absorbing Boundary Condition* (HABC). They can be interpreted as a combination of Sponge Layer and ABC.

It is possible to use ABCs that do not require domain extension, enforced as boundary conditions, as suggested by the A1 Clayton's condition (Clayton and Engquist, 1977), and the schemes by Higdon (1986, 1987). While these demand very little in terms of computational cost, they can still be prone to spurious reflections, if used on their own. However, they can be effective if used in a hybrid way, in combination with sponge layers, as we illustrate below.

Clayton's A1 boundary condition Clayton and Engquist (1977) is based on a One-Way Wave Equation (OWWE). This simple condition is such that outgoing waves normal to the border would leave without reflection. At the $\partial\Omega_1$ part of the boundary, the condition is,

$$u_t(x,z,t) - c(x,z)u_x(x,z,t) = 0, \quad (x,z) \in \partial\Omega_1, \tag{20}$$

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

while at $\partial\Omega_3$ the condition is

$$u_t(x,z,t) + c(x,z)u_x(x,z,t) = 0, \quad (x,z) \in \partial\Omega_3, \tag{21}$$

and at $\partial\Omega_2$ the condition is

$$u_t(x,z,t) - c(x,z)u_z(x,z,t) = 0, \quad (x,z) \in \partial\Omega_2, \tag{22}$$

where we have explicitly expanded the spatial domain variable in its components ($\mathbf{x} = (x,z)$).

The Higdon Boundary condition (Higdon, 1986, 1987), can take into account additional incidence directions, not only the normal direction as in Clayton's A1 condition. The scheme, termed to be of order $p \in \mathbb{N}$, is given at $\partial\Omega_1$ and $\partial\Omega_3$ by:

$$\prod_{j=1}^{p} \left[ \cos(\alpha_j) \left( \frac{\partial}{\partial t} - c(\mathbf{x}) \frac{\partial}{\partial x} \right) u(\mathbf{x},t) \right] = 0, \tag{23}$$

and at $\partial\Omega_2$

$$\prod_{j=1}^{p} \left[ \cos(\alpha_j) \left( \frac{\partial}{\partial t} - c(\mathbf{x}) \frac{\partial}{\partial z} \right) u(\mathbf{x},t) \right] = 0. \tag{24}$$

This method ensure that outgoing waves with angle of incidence at the boundary equal to $\alpha_j$ present no reflection. The method we use in this work employs order 2 ($p = 2$) and angles 0 and $\pi/4$.

To combine these schemes with sponge layers, thus leading to hybrid schemes (HABC), we also extend the spatial domain as in Figure 1b. The difference with respect to previous schemes is that this extended region will now be considered as the union of several nested gradual extensions. As represented in Figure 2, we define a region $A_M = \Omega_0$, and the regions $A_k, k = M-1, \cdots, 1$ will be defined as the previous region $A_{k+1}$ to which we add one extra grid line to the left, right and bottom sides of it, such that the final region $A_1 = \Omega$.

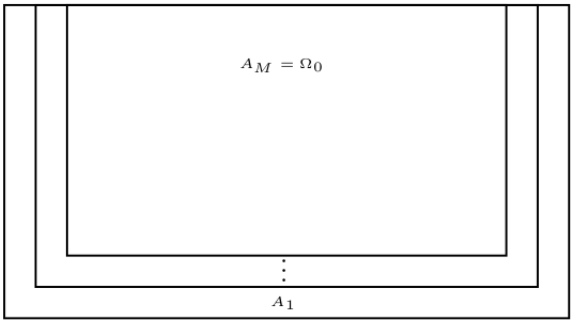



**Figure 2.** Nesting of domains for the Hydrid ABC method. The full region $A_1$ is equivalent to $\Omega$.

To illustrate how the HABC is used, we will describe the process of how we obtain a solution using the usual solution of Acoustic Wave Equation together with the absorbing conditions showed in A1 and Higdon schemes. First, assume $u(\mathbf{x}, t - \Delta t)$

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

is known at instant $t-\Delta t$ in all the extended $\Omega$ domain. We then update one time step from the solution $u(\mathbf{x},t-\Delta t)$ to $u(\mathbf{x},t)$ using the usual Acoustic Wave Equation over $\Omega$, with the null Dirichlet or Neumann boundary conditions defined for $\partial\Omega$.

Now, for each region $A_k$, with $k$ going from the innermost domain $A_M$ to the outermost domain $A_1$, we construct an auxiliary functions, $u_k(\mathbf{x},t)$, based on the current solution, $u(\mathbf{x},t)$, by applying the absorbing condition A1 or Higdon for the domain $(A_k)$. For finite difference schemes, this implies in altering only the values of $u(\mathbf{x},t)$ at the border of $A_k$, that is, on $\partial A_k$, to obtain $u_k(\mathbf{x},t)$. The final solution for each region $A_k$, which will be the input solution for region $A_{k-1}$, will be given by a convex combination between $u_k(\mathbf{x},t)$ and $u(\mathbf{x},t)$, as

$$\tilde{u}(\mathbf{x},t) = (1-\omega_k)u(\mathbf{x},t) + \omega_k u_k(\mathbf{x},t), \tag{25}$$

where $w_k$ is a weight function that grows from zero at $A_M = \partial\Omega_0$ to one at $A_1 = \partial\Omega$, and $\tilde{u}(\mathbf{x},t)$ will be used as the new $u(\mathbf{x},t)$ for the next region $(A_{k-1})$. Summarizing, we loop over the nested regions, from innermost to outermost, subsequently applying the pointwise A1 or Higdon boundary conditions, weighting with respect to the a distance metric of each boarder from the innermost domain (defined by weights $w_k$).

The particular weight function to be used could vary linearly or non-linearly (Liu and Sen, 2018). We can choose a *linear* weight function as

$$\omega_k = \frac{M-k}{M}; \tag{26}$$

or, preferably, a *non linear* function,

$$\omega_k = \begin{cases} 1, & \text{if } 1 \le k \le P+1, \\ \left(\dfrac{M-k}{M-P}\right)^{\alpha}, & \text{if } P+2 \le k \le M-1. \\ 0, & \text{if } k = M. \end{cases} \tag{27}$$

We take $P = 2$ and we choose $\alpha$ following Liu and Sen (2018):

- $\alpha = 1.5 + 0.07(\text{npt} - P)$, in the case of A1;
- $\alpha = 1.0 + 0.15(\text{npt} - P)$, in the case of Higdon.

where the value of $npt$ designates the number of discrete points that define the length of the extended region in the direction $x$ or $z$. In our experiments, we observed that HABC produces better results with a non-linear weight function, but the choice of the type of weights can be adapted according to the application. Moreover, the values for $\alpha$ follow Liu and Sen (2018) and these values are for our applications, but the parameter can be adjusted for specific cases.

## 4 ABCs in Adjoint Problem

After introducing the different approaches of ABCs, the goal now is to obtain the adjoint equations for each of them. To do so, the augmented functional is considered, in which the constraints are given by the wave equation and by the equations used in ABCs.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

As shown before, to apply any of the ABCs of interest of this study, the domain considered is built as the union of the physical domain $\Omega_0$ and the extended domain $\Omega_e = \Omega/\Omega_0$. In FWI, the goal is to minimize the objective functional $I(m)$ on the physical domain $\Omega_0$. Therefore, the objective functional remains being defined by the expression (1), but now defined over $\Omega_0$.

### 4.1 Damping

The acoustic wave equation with dampening mechanism is given by (6). The corresponding adjoint equations are obtained on pursuing the same sequence presented by Plessix (2006). So, the first step is to write the augmented functional considering the equations defined in the physical and in the extended domains:

$$\mathcal{L}(u,m,u^\dagger) = I(m) - \int_\tau \int_\Omega u^\dagger \cdot \left(m u_{tt} - \nabla^2 u - \zeta u_t - f\right) \mathrm{d}\mathcal{V}\mathrm{d}t,$$

where $u = u(\mathbf{x},t)$ and $\zeta = \zeta(\mathbf{x})$.

In the current case, the wave equation with dampening mechanism is defined in the domain $\Omega$ illustrated by the blue region in Fig. 1(b). To obtain the adjoint equation, their initial and boundary conditions, the gradient $\frac{\partial}{\partial u}[\mathcal{L}(u,u^\dagger,m)]u'$ is written as follows:

$$\lim_{\epsilon\to 0} \frac{\mathcal{L}(u+\epsilon u', u^\dagger, m) - \mathcal{L}(u,u^\dagger,m)}{\epsilon} = \nabla_u[\mathcal{L}(u,u^\dagger,m)]u' = \nabla_u[I(m)]u' - \int_\tau \int_\Omega u^\dagger \cdot \left(m u'_{tt} - \nabla^2 u' + \zeta u'_t\right) \mathrm{d}\mathcal{V}\mathrm{d}t,$$

where $u'$ is a perturbation of the variable $u$.

Integration by parts is applied as shown below:

$$\nabla_u[\mathcal{L}(u,u^\dagger,m)]u' = \nabla_u[I(m)]u' - \int_\tau \int_\Omega \left(m u^\dagger_{tt} - \nabla^2 u^\dagger - \zeta u^\dagger_t\right) \cdot u' \, \mathrm{d}\mathcal{V}\mathrm{d}t + \mathcal{B},$$

where the gradient $\nabla_u[I(m)]u'$ is given by:

$$\nabla_u[I(m)]u' = \int_\tau \int_\Omega (u - u^o) u' \boldsymbol{\delta}(\mathbf{x} - \check{\mathbf{x}}) \mathrm{d}\mathcal{V}\mathrm{d}t. \tag{28}$$

The term $\mathcal{B}$ is the bilinear concomitant, which the integration by parts entails. After applying the Divergence Theorem, $\mathcal{B}$ reads:

$$\mathcal{B} = \mathcal{B}(u', u^\dagger, m) = \int_\tau \int_{\partial(\Omega)} \left[(-u^\dagger \nabla u' + \nabla u^\dagger u')\right] \cdot \mathbf{n} \, \mathrm{d}\mathcal{S}\mathrm{d}t + \int_\Omega \left[m\left(u^\dagger \cdot u'_t - u^\dagger_t u'\right) - \zeta u^\dagger u'\right] \mathrm{d}\mathcal{V} \Bigg|_{t_0}^{t_f}.$$

The adjoint wave equation, their initial and boundary conditions, are then defined by imposing $\nabla_u[\mathcal{L}(u,u^\dagger,m)]u' = 0$. That means defining the adjoint equation as follows:

$$m u^\dagger_{tt} - \nabla^2 u^\dagger + \zeta u^\dagger_t = (u - u^o)\boldsymbol{\delta}(\mathbf{x} - \check{\mathbf{x}}) \; \forall \mathbf{x} \in \Omega. \tag{29}$$

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

Also, on considering the boundary and initial conditions of the forward wave equation, and by imposing the adjoint boundary conditions given by:

$$u^{\dagger} = 0, \quad \forall \mathbf{x} \in \partial\Omega_i, \quad i = 1,2,3, \tag{30}$$

$$\frac{\partial u^{\dagger}}{\partial z} = 0, \quad \forall \mathbf{x} \in \partial\Omega_4, \tag{31}$$

The bilinear concomitant is reduced to:

$$\mathcal{B} = \mathcal{B}(u', u^{\dagger}, m) = \int_{\Omega} \left[ m \left( u^{\dagger} \cdot u'_t - u^{\dagger}_t u' \right) - \zeta u^{\dagger} u' \right] \mathrm{d}\mathcal{V} \Bigg|_{t_0}^{t_f}.$$

The homogeneous initial conditions of the forward wave equation drive the domain integral to zero, that is evaluated at $t_0 = 0$. In order to eliminate the corresponding domain integral for $t = t_f$, one could impose the following homogeneous final conditions on the adjoint variable: $u^{\dagger}$:

$$u^{\dagger}(\mathbf{x}, t_f) = u^{\dagger}_t(\mathbf{x}, t_f) = 0 \tag{32}$$

To make the algebra simpler with respect to these conditions, one could define an adjoint time variable in the form

$$t^{\dagger} \equiv t_f - t \quad \Rightarrow \quad \mathrm{d}t^{\dagger} = -\mathrm{d}t \quad \Rightarrow$$

$$\Rightarrow \begin{cases} t = 0 \Leftrightarrow t^{\dagger} = t_f \\ t = t_f \Leftrightarrow t^{\dagger} = 0 \end{cases} \quad \Rightarrow \quad u^{\dagger}(\mathbf{x}, t_f) = u^{\dagger}_t(\mathbf{x}, t_f) = 0\,.$$

As a result of this change of variables, the adjoint wave equation (29) becomes, in its final form:

$$m u^{\dagger}_{tt} - \nabla^2 u^{\dagger} - \zeta u^{\dagger}_t = (u - u^o)\boldsymbol{\delta}(\mathbf{x} - \check{\mathbf{x}}) \; \forall \mathbf{x} \in \Omega, \tag{33}$$

which means it is a self-adjoint wave equation.

### 4.2 PML and CPML

The work of Xie et al. (2014) presents a mathematical development to obtain the adjoint wave system with the refereed complex-frequency-shifted unsplit-field perfectly matched layer (CFS-UPML). In essence, the Fourier transform $\hat{u} = \int_{-\infty}^{\infty} e^{st} u(\mathbf{x}, t)\, \mathrm{d}t$ of the displacement vector, $u$, satisfies the Helmholtz equation:

$$s^2 \hat{u} = \nabla(c^2 \nabla \hat{u}) \tag{34}$$

where $s \in \mathcal{C}$. Also, one considers the transform of spatial coordinates $\mathbf{x}$:

$$\tilde{\mathbf{x}} := \int_{\Omega} \boldsymbol{\gamma}(\mathbf{x})\, \mathrm{d}\mathcal{V}.$$

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

where, for the 2–D case, $\boldsymbol{\gamma}(\mathbf{x}) = [\gamma_1, \gamma_2]^T = [\gamma_1(\mathbf{x}), \gamma_2(\mathbf{x})]^T$ is the complex stretching function,

$$\gamma_j = \kappa_j + \frac{\zeta_j}{\alpha_j + is}, \quad j = 1,2 \quad \text{for 2–D case}$$

and $i = \sqrt{-1}$.

The next step consists in reformulating eq. (34) in time domain. In the 2–D case, it leads to:

$$mL(t) * u = \\ = \frac{\partial}{\partial x}\left(\mathcal{F}^{-1}\left(\frac{\gamma_2}{\gamma_1}\right) * u_x\right) + \frac{\partial}{\partial z}\left(\mathcal{F}^{-1}\left(\frac{\gamma_1}{\gamma_2}\right) * u_z\right) + f, \tag{35}$$

where $L(t) = \mathcal{F}^{-1}(s^2\gamma_1\gamma_2)$, $\mathcal{F}^{-1}$ is the inverse Fourier transform, and $*$ represents a convolution.

On taking the wave equation (35) into account, the adjoint system is then defined as (Xie et al., 2014):

$$mL(t) * u^\dagger = \frac{\partial}{\partial x}\left(\mathcal{F}^{-1}\left(\frac{\gamma_2}{\gamma_1}\right) * u_x^\dagger\right) + \\ + \frac{\partial}{\partial z}\left(\mathcal{F}^{-1}\left(\frac{\gamma_1}{\gamma_2}\right) * u_z^\dagger\right) + (u - u^o)\boldsymbol{\delta}(\mathbf{x} - \check{\mathbf{x}}), \tag{36}$$

which satisfies the conditions (30), (31) and (32).

In the standard PML formulation, $\alpha_j = 0$, $\kappa_j = 1$ (Berenger et al., 1994). Therefore, the adjoint wave equation with PML is a particular case of the methodology presented by Xie et al. (2014). Hence, for the 2–D case the convolutions of the adjoint wave equation (36) are given by:

$$L(t) * u^\dagger = u_{tt}^\dagger + (\zeta_1 + \zeta_2)u_t^\dagger + \zeta_1\zeta_2 u^\dagger, \tag{37}$$

$$\mathcal{F}^{-1}\left(\frac{\gamma_2}{\gamma_1}\right) * u_x^\dagger = u_x^\dagger - (\zeta_1 - \zeta_2)\left[\exp^{-\zeta_1 t} H(t)\right] * u_x^\dagger, \tag{38}$$

$$\mathcal{F}^{-1}\left(\frac{\gamma_1}{\gamma_2}\right) * u_z^\dagger = u_z^\dagger - (\zeta_2 - \zeta_1)\left[\exp^{-\zeta_2 t} H(t)\right] * u_z^\dagger, \tag{39}$$

where $H(t)$ is the Heaviside distributions.

The convolution terms $(\zeta_1 - \zeta_2)\left[\exp^{-\zeta_1 t} H(t)\right] * u_x^\dagger$ and $(\zeta_2 - \zeta_1)\left[\exp^{-\zeta_2 t} H(t)\right] * u_z^\dagger$ may be solved by considering auxiliary differential equations (Grote and Sim, 2010; Xie et al., 2014). On defining the auxiliary functions as

$$\phi_1^\dagger = (\zeta_1 - \zeta_2)\left[\exp^{-\zeta_1 t} H(t)\right] * u_x^\dagger,$$
$$\phi_2^\dagger = (\zeta_2 - \zeta_1)\left[\exp^{-\zeta_2 t} H(t)\right] * u_z^\dagger,$$

the terms (38) and (39) are rewritten as:

$$\mathcal{F}^{-1}\left(\frac{\gamma_2}{\gamma_1}\right) * u_x^\dagger = u_x^\dagger - \phi_1^\dagger, \quad \mathcal{F}^{-1}\left(\frac{\gamma_1}{\gamma_2}\right) * u_z^\dagger = u_z^\dagger - \phi_2^\dagger.$$

Therefore, the adjoint wave equation (36) with the employment of PML method reads:

$$mu_{tt}^\dagger + (\zeta_1 + \zeta_2)u_t^\dagger + \zeta_1\zeta_2 u^\dagger = \nabla^2 u^\dagger + (\phi_1^\dagger)_x + (\phi_2^\dagger)_z + (u - u^o)\boldsymbol{\delta}(\mathbf{x} - \check{\mathbf{x}}), \tag{40}$$

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

where the auxiliary functions $\phi_1^\dagger$ and $\phi_2^\dagger$ satisfy the respective auxiliary differential equations:

$$(\phi_1^\dagger)_t = -\zeta_1\phi_1^\dagger + (\zeta_2 - \zeta_1)u_x^\dagger, \tag{41}$$

$$(\phi_2^\dagger)_t = -\zeta_2\phi_2^\dagger + (\zeta_1 - \zeta_2)u_z^\dagger, \tag{42}$$

The adjoint wave equation (36) may be also written according to the formulation presented in Pasalic and McGarry (2010), i.e., CPML formulation. In this case, $\alpha_j$ is a positive value, and $\kappa_j = 1$ (Pasalic and McGarry, 2010). Therefore, to write an adjoint system with CPML method, eq. (36) is rewritten as:

$$m\mathcal{F}^{-1}(s^2) * u^\dagger = \mathcal{F}^{-1}\left(\frac{1}{\gamma_1}\right) * \frac{\partial}{\partial x}\left(\mathcal{F}^{-1}\left(\frac{1}{\gamma_1}\right) * u_x^\dagger\right) +$$

$$+ \mathcal{F}^{-1}\left(\frac{1}{\gamma_2}\right) * \frac{\partial}{\partial z}\left(\mathcal{F}^{-1}\left(\frac{1}{\gamma_2}\right) * u_z^\dagger\right) + (u - u^o)\boldsymbol{\delta}(\mathbf{x} - \check{\mathbf{x}}), \tag{43}$$

where $\mathcal{F}^{-1}(s^2) * u^\dagger = u_{tt}$ and, on following Pasalic and McGarry (2010), we have:

$$\mathcal{F}^{-1}\left(\frac{1}{\gamma_1}\right) * \frac{\partial}{\partial x}\left(\mathcal{F}^{-1}\left(\frac{1}{\gamma_1}\right) * u_x^\dagger\right) = u_{xx}^\dagger + (\phi_1^\dagger)_x + (\psi_1^\dagger)_x$$

$$\mathcal{F}^{-1}\left(\frac{1}{\gamma_1}\right) * \frac{\partial}{\partial z}\left(\mathcal{F}^{-1}\left(\frac{1}{\gamma_1}\right) * u_z^\dagger\right) = u_{zz}^\dagger + (\phi_2^\dagger)_z + (\psi_2^\dagger)_z.$$

Therefore, the adjoint wave equation (43) is cast in the form:

$$mu_{tt}^\dagger = \nabla^2 u^\dagger + (\psi_1^\dagger)_x + (\psi_2^\dagger)_z + \phi_1^\dagger + \phi_2^\dagger + (u - u^o)\boldsymbol{\delta}(\mathbf{x} - \check{\mathbf{x}}), \tag{44}$$

where the auxiliary functions ($\psi_1^\dagger$, $\psi_2^\dagger$, $\phi_1^\dagger$, $\phi_2^\dagger$) are obtained by using the auxiliary equations given by:

$$\psi_1^\dagger(\mathbf{x}, t_{n-1}) = a_1(\mathbf{x})\psi_1^\dagger(\mathbf{x}, t_n) + b_1(x, z)u_x^\dagger(\mathbf{x}, t_{n-1}), \tag{45}$$

$$\psi_2^\dagger(\mathbf{x}, t_{n-1}) = a_2(\mathbf{x})\psi_2^\dagger(\mathbf{x}, t_n) + b_2(\mathbf{x})u_z^\dagger(\mathbf{x}, t_{n-1}), \tag{46}$$

$$\phi_1^\dagger(\mathbf{x}, t_{n-1}) = a_1(\mathbf{x})\phi_1^\dagger(\mathbf{x}, t_n) + b_1(\mathbf{x})\left[u_{xx}^\dagger(\mathbf{x}, t_{n-1}) + \psi_{1,x}^\dagger(\mathbf{x}, t_{n-1})\right], \tag{47}$$

$$\phi_2^\dagger(\mathbf{x}, t_{n-1}) = a_2(\mathbf{x})\phi_2^\dagger(\mathbf{x}, t_n) + +b_2(\mathbf{x})\left[u_{zz}^\dagger(\mathbf{x}, t_{n-1}) + \psi_{2,z}^\dagger(\mathbf{x}, t_{n-1})\right].$$

The adjoint auxiliary equations (45)–(**??**) are solved recursively in adjoint/backward time $t^\dagger$, which is reverse with respect to the forward time $t$. Again, it is possible to note that the resulting adjoint equations are self-adjoint with respect to the forward problem with PML.

### 4.3 Hybrid Absorbing Boundary Condition (HABC)

The HABC methods apply the discrete convex combination (25) to a discrete transitional area of $\Omega_e$. As explained in section 3.5, this approach combines the solution of the wave equation with the boundary conditions A1 Clayton's condition for HABC-A1, and Higdon for HABC-Higdon. So, to derive the adjoint equations, let us start by considering boundary conditions on the truncated boundaries ($\partial\Omega_{i_0}$, $i_0 = 1, 2, 3$) that satisfy the A1 Clayton's. In this case, the augmented functional is given by (3).

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

365 On integrating it by parts, one arrives at:

$$\nabla_u[\mathcal{L}(u,u^\dagger,m)]u' = \nabla_u[I(m)]u' - \int_\tau \int_\Omega \left(m u^\dagger_{tt} - \nabla^2 u^\dagger\right)\cdot u'\,\mathrm{d}\mathcal{V}\mathrm{d}t + \mathcal{B},$$

where the adjoint wave equation is defined by (5), reaching $\nabla_u[\mathcal{L}(u,u^\dagger,m)]u' = \mathcal{B}$. On adopting the boundary condition at the free surface (31), and zero initial conditions of the forward and adjoint variables, it yields:

$$\nabla_u[\mathcal{L}(u,u^\dagger,m)]u' = -\int_\tau \int_{\partial\Omega_{i_0}} \left[-u^\dagger(\nabla u') + \nabla u^\dagger u'\right]\cdot\mathbf{n}\,\mathrm{d}\mathcal{S}\mathrm{d}t. \tag{48}$$

370 In the truncated boundaries, $\partial\Omega_{i_0}$, A1 Clayton's boundary condition reads

$$\frac{1}{c}\frac{\partial u'}{\partial t} + \nabla u'\cdot\mathbf{n} = 0,$$

which implies $\nabla u'\cdot\mathbf{n} = -\frac{1}{c}\frac{\partial u'}{\partial t}$. Hence, the right side term of eq. (48) is rewritten as:

$$-\int_\tau \int_{\partial\Omega_{i_0}} -u^\dagger\left(-\frac{1}{c}\frac{\partial u'}{\partial t}\right) + \left(\nabla u^\dagger u'\right)\cdot\mathbf{n}\,\mathrm{d}\mathcal{S}\mathrm{d}t.$$

Next, the integration by parts with respect to time can be applied and eq. (48) becomes:

375 $$\frac{\partial}{\partial u}[\mathcal{L}(\mathbf{u},\mathbf{u}^\dagger,m)]u' = -\int_\tau \int_{\partial\Omega_{i_0}} \frac{1}{c}\frac{\partial u^\dagger}{\partial t}u' + \left(\nabla u^\dagger u'\right)\cdot\mathbf{n}\,\mathrm{d}\mathcal{S}\mathrm{d}t,$$

since $u(\mathbf{x},0) = u^\dagger(\mathbf{x},t_f) = 0.0\ \forall\mathbf{x}\in\Omega$ is satisfied. Lastly, on imposing $\frac{1}{c}\frac{\partial u^\dagger}{\partial t} + \nabla u^\dagger\cdot\mathbf{n} = 0$, the extremum $\frac{\partial}{\partial u}[\mathcal{L}(\mathbf{u},\mathbf{u}^\dagger,m)]u' = 0$ is then realized.

The same approach may be employed to obtain the adjoint wave equation in the case where the Higdon boundary condition is imposed on the truncated boundaries $\partial\Omega_{i_0}$. Therefore, the adjoint wave equation is defined by eq. (5). Also, on imposing
380 the boundary condition at the free surface (31), zero initial conditions for the forward ($u$) and adjoint ($u^\dagger$) variables, the gradient is reduced to eq. (48). So, based on Higdon (1986), Higdon's boundary condition was proposed by considering the wave propagating outward at an angle of incidence $\alpha$. In a two-dimensional domain, the wave solution was described by $u = f(x\cos\alpha + y\sin\alpha + ct)$. Hence, the generalized boundary condition is

$$\Pi_{j=1}^{p}\left(\cos(\alpha_j)\left(\frac{1}{c}\frac{\partial}{\partial t} - (\mathbf{n}\cdot\nabla)\right)u(\mathbf{x},t)\right) = 0, \tag{49}$$

385 such that $|\alpha| \leq \frac{\pi}{2}$ for all $j$. That allows to write $\nabla u'\cdot\mathbf{n} = -\cos\alpha\frac{1}{c}\frac{\partial u'}{\partial t}$, and the eq. (48) as follows:

$$\frac{\partial}{\partial u}[\mathcal{L}(\mathbf{u},\mathbf{u}^\dagger,m)]u' = -\int_\tau \int_{\partial\Omega_{i_0}} u^\dagger\left(\cos\alpha\frac{1}{c}\frac{\partial u'}{\partial t}\right) + \left(\nabla u^\dagger u'\right)\cdot\mathbf{n}\,\mathrm{d}\mathcal{S}\mathrm{d}t.$$

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

On integrating it by parts in time, the above equation becomes :

$$\frac{\partial}{\partial u}[\mathcal{L}(\mathbf{u},\mathbf{u}^\dagger,m)u' = -\int_\tau \int_{\partial\Omega_{i_0}} \cos\alpha \frac{1}{c}\frac{\partial u^\dagger}{\partial t}u' - \left(\nabla u^\dagger u'\right)\cdot\mathbf{n}\,\mathrm{d}\mathcal{S}\mathrm{d}t.$$

As a result of it, on imposing $\cos\alpha\frac{1}{c}\frac{\partial u^\dagger}{\partial t} - \nabla u^\dagger = 0$, the extreme $\frac{\partial}{\partial u}[\mathcal{L}(\mathbf{u},\mathbf{u}^\dagger,m)]u' = 0$ is attained and the generalization

$$\Pi_{j=1}^{p}\left(\cos(\alpha_j)\left(\frac{1}{c}\frac{\partial}{\partial t} - (\mathbf{n}\cdot\nabla)\right)u^\dagger(\mathbf{x},t)\right) = 0, \tag{50}$$

may be imposed as boundary conditions on the adjoint wave problem.

After setting the adjoint counterparts to A1 Clayton's and Higdon boundary conditions, the adjoint HABC approach is completed using a similar process of forward problem. In summary, once the adjoint is solved inverse in time, assume $u^\dagger(\mathbf{x},t+\Delta t)$ is known at instant $t+\Delta t$ in all the extended $\Omega$ domain, update it to $u^\dagger(\mathbf{x},t)$ using the usual Adjoint Acoustic Wave Equation over $\Omega$, then construct the auxiliary functions $u_k^\dagger(\mathbf{x},t)$ for each region, from $A_M$ to $A_1$, applying the A1 or Higdon conditions for each region. As in the forward problem, we construct each update of the solution to the adjoint problem, at each region $A_k$, with a convex combination using a weight $\omega_k$,

$$\tilde{u}^\dagger(\mathbf{x},t) = (1-\omega_k)u^\dagger(\mathbf{x},t) + \omega_k u_k^\dagger(\mathbf{x},t), \tag{51}$$

where $\tilde{u}^\dagger(\mathbf{x},t)$ will be used as the new $u^\dagger(\mathbf{x},t)$ for the next region $(A_{k-1})$.

This derivation shows, once more, the self-adjoint nature of the adjoint problem with HABC-A1 or HABC-Higdon.

## 5 Computational Framework

The numerical simulations were carried out using the Devito software (Louboutin et al., 2019; Luporini et al., 2018; Kukreja et al., 2016). According to its own website, Devito is a Python package that combines a domain-specific Language (DSL) and a full code generation framework. It is especially geared towards the design of highly optimized finite difference kernels, for its use in inverse problems. It makes use of SymPy to allow symbolic definition of operators at a high-level notation, and then it generates optimized code that is automatically tuned to specified computer architectures.

### 5.1 Coding framework

In general, symbolic computation is a powerful tool, as it allows users to build complex solvers in only a few lines of high-level code. However, symbolic computation is usually impractical, from a computational performance point of view, for most complex applications. On the other hand, considering the compilation of a high-level symbolic solver into a highly optimized low-level code, with adjustable stencil discretization at run-time, one should be able to develop computationally efficient methods, reducing the coding development time. This is the underling goal of Devito. Here we highlight the main aspects that concern this work with respect to software development in Devito.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

In this work, we use the main Devito backend as driver, but implement all methods using only high-level symbolic methods available from Devito. This allows methods to be easily modified by interested users. Our implementation is described in details in the ABC Devito tutorials, available in the master branch of Devito [2].

Due to the simplicity of working with symbolic equations, the code development can be accomplished with minor modifications of existing codes in Devito for typical acoustic wave propagation. We highlight in Figure 5.2 the main implementation characteristics. The differential operators have a syntax related to their original structures, such as *dt, dt*$^2$*, dx, dz, Laplace*, for instance. Their corresponding Finite Difference approximations (Fornberg, 1988) can then be picked by the user among many available, or custom designed, schemes for each operator. An important resource of Devito for ABCs is the possibility of partitioning the domain of interest into subdomains, to which distinct attributes can be assigned.

The creation of space-dependent ($Function$), space-time dependent ($Timefunction$) and other types of fields is done as a pre-processing step. It amounts to setting the properties linked to that field, such as, for instance, spatial order, temporal order, staggering type, floating number type, among other specific properties of each field.

The term $op$ described in Figure 5.2 represents the time evolution operator for a given set of symbolic equations, and is where all the backend compilation, optimization, and running of Devito takes place. At this point, a user defines a number of time-steps and its size ($dt$). In that operator, one places the elements called *stencils*, which represent the differential equations that are applied to each particular subdomain. Furthermore, the $op$ carries the natural boundary conditions ($bc$), forcing terms ($src\_term$) and information about the receivers ($rec\_term$).

### 5.2 Computational performance

Devito uses Python as frontend, to allow ease of code development, and C/C++ as backend language for optimized computational runtime. Architecture dependent optimizations are possible, and parallel distribution of task also. To date, Devito allows parallelization with OpenMP (*DEVITO_LANGUAGE=openmp*), MPI (*DEVITO_MPI=n*, where $n$ is the number of nodes) and some GPGPU compilation support.

In this work, simulations have been executed on the Mintrop HPC cluster, at the University of São Paulo. The computational executions were carried out on an AMD node, which has dual sockets AMD EPYC 7601 clocked at 2.2 GHz with 64 cores and 512GB of memory DDR4.

In FWI executions, the sources of shot waves ran in parallel with computing library Dask (docs.dask.org/en/latest/), where the number of tasks was equal to the number of sources. OpenMP-only was activated to solve the partial differential equations (PDEs) in parallel in the Devito framework.

Computational performance (wall-clock run time and memory usage) was measured using the Dask diagnostic performance tool.

[2]PML Jupyter notebook example: nbviewer.org/github/devitocodes/devito/blob/master/examples/seismic/abc_methods/03_pml.ipynb

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions
Open Access
EGU

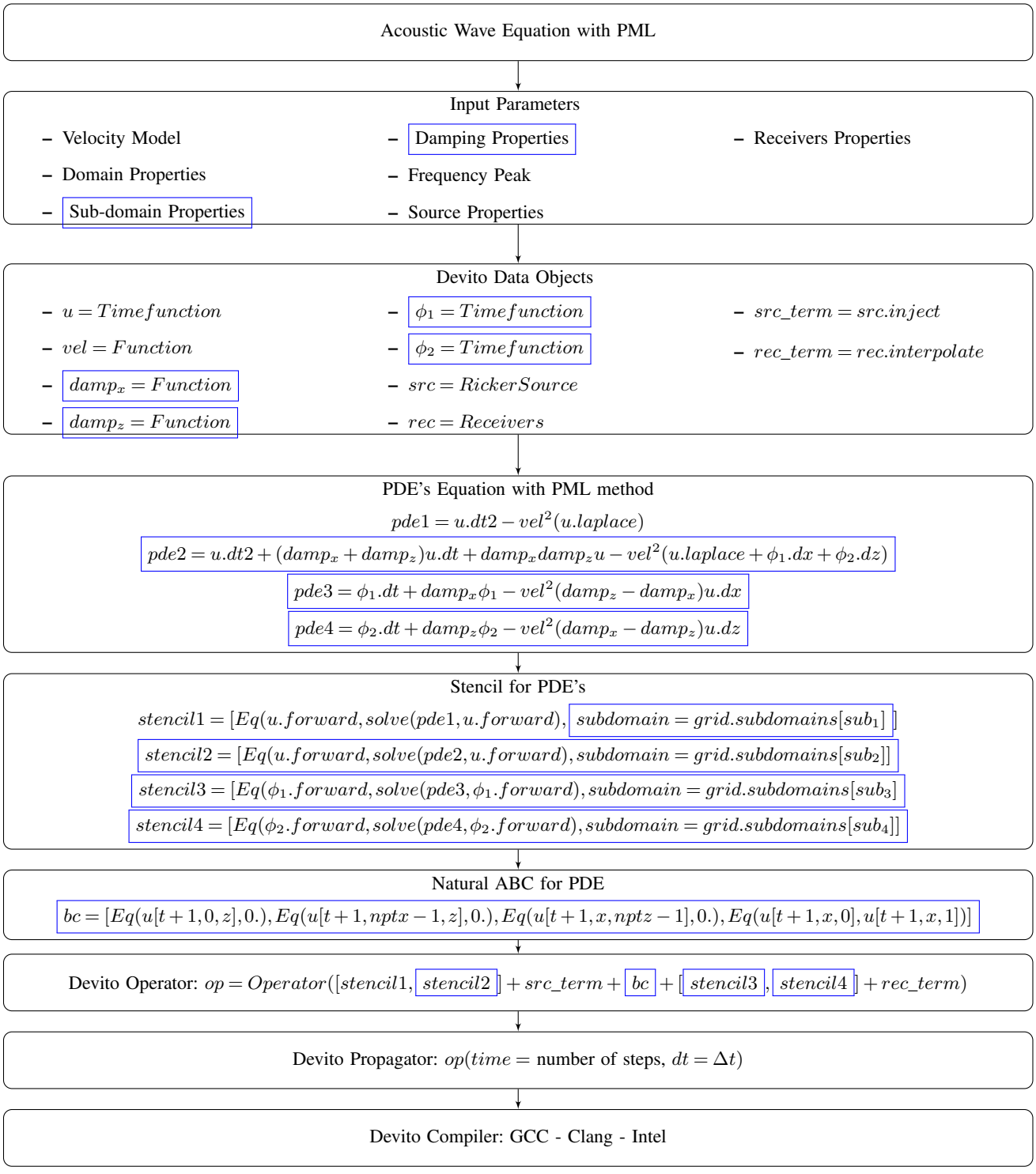



**Figure 3.** Example of logical implementation of PML method in the Devito coding framework. The diagram is similar to that of the original Devito development paper (Louboutin et al., 2019), but we highlight in blue boxes the kind of changes required for a PML implementation.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

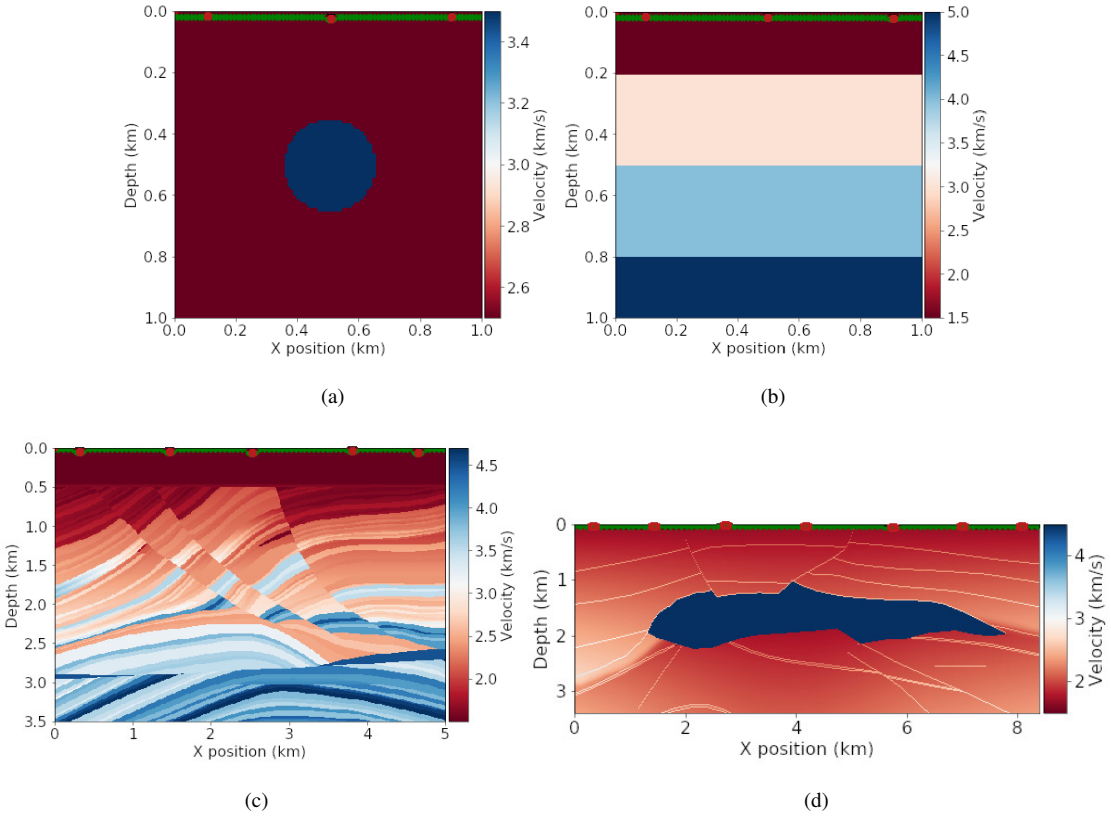



**Figure 4.** Velocity models: (a) circle; (b) horizontal layers; (c) part of Marmousi; (d) 2D SEG/EAGE. The red points illustrate the source positions, and the green points illustrate the receiver positions.

## 6 Analysis of ABCs

In this section, we assess the performance of the ABCs methods on the forward and adjoint wave equations. In the literature, analyses of ABCs are usually limited to the homogeneous velocity model (Gao et al., 2017; Grote and Sim, 2010; Liu and Sen, 2012). Also, to the best of our knowledge, those conditions have not yet been assessed for their role in the adjoint problem. In the present work, we propose to do precisely that: to evaluate ABCs for heterogeneous velocity models and to do so for the adjoint problem, as well.

The objective is to carry out the ABCs analysis for the usual setup adopted in a FWI problem, e.g., on considering heterogeneous velocity models. These are illustrated in Figure 4, they are respectively referred to as circle, horizontal layers, part of Marmousi (Martin et al., 2006) and a 2D SEG/EAGE salt (Aminzadeh and Brac, 1997). The placement of receivers and sources follows the usual configuration adopted in the literature to run a FWI case (Virieux and Operto, 2009). That is, sources and receivers located closer to the free surface. Tables 1 present the numerical setup that was used to run the analyses of both forward and adjoint solutions with the ABCs.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

| Velocity models | Part of Marmousi |
|---|---|
| Physical domain size | $L_x = 5.0Km$, $L_z = 3.5Km$ |
| Total time | $t = 6.4s$ |
| Source number ($n_s$) | 5 |
| Receiver number ($n_r$) | 350 |
| Source positions $[x, z](m)$ | $[(100 + i \times 960), 0.125]$, $i = 0, ..., n_s$ |
| Receiver positions $[x, z](m)$ | $[(100 + i \times 48), 0.225]$, $i = 0, ..., n_r$ |
| Mesh spacing (m) | $\Delta x = \Delta z = 10m$ |
| Velocity models | Circle and Horizontal layers |
| Physical domain size | $L_x = L_z = 1Km$ |
| Total time | $t = 1.0s$ |
| Source number ($n_s$) | 3 |
| Receiver number ($n_r$) | 100 |
| Source positions $[x, z](m)$ | $[(100 + i \times 266), 20]$, $i = 0, ..., n_s$ |
| Receiver positions $[x, z](m)$ | $[(100 + i \times 9.8), 30]$, $i = 0, ..., n_r$ |
| Mesh spacing (m) | $\Delta x = \Delta z = 10m$ |
| Velocity models | 2D SEG/EAGE salt |
| Physical domain size | $L_x = 8.0Km$, $L_z = 3.5Km$ |
| Total time | $t = 6.4s$ |
| Source number ($(n_s)$) | 8 |
| Receiver number ($n_r$) | 550 |
| Source positions $[x, z](m)$ | $[(100 + i \times 960), 0.2]$, $i = 0, ..., n_s$ |
| Receiver positions $[x, z](m)$ | $[(100 + i \times 15), 0.5]$, $i = 0, ..., n_r$ |
| Mesh spacing (m) | $\Delta x = 15m$, $\Delta z = 10m$ |

**Table 1.** Setup used in the ABCs analyses.

The reference fields used in all tests are designed to keep boundary reflections from ever reaching the actual domain of interest. To achieve this, the computational domain for the reference solution is extended, and the simulated time is set in such a way that neither the forward nor the backward waves have enough time to reach the outermost boundaries. Figures 5 and 6 show the reference solutions of the forward and adjoint solvers, respectively. In none of the cases, the waves have had time to reach the truncated boundaries $\partial\Omega_{i_0}$, $i_0 = 1, 2, 3$. Therefore, the reference fields that are used as a base for comparisons are free of reflections. These extended regions should not be confused with the absorbing layers used for the boundary schemes tested here. The goal of this particular very large extended region is solely to define adequate reference solutions with no inbound reflected waves. The quantitative error evaluation is given by the expression

$$E(s_{ref}, s) = \frac{||s_{ref}(\mathbf{x}, t_f) - s(\mathbf{x}, t_f)||_2}{||s_{ref}(\mathbf{x}, t_f)||_2}, \tag{52}$$

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

which computes the relative error of the forward/adjoint solution ($u/u^{\dagger}$) using ABC, with the corresponding one in the reference field on the physical (inner) domain of interest ($\Omega_0$). The variables $s_{ref}(\mathbf{x}, t_f)$ and $s(\mathbf{x}, t_f)$ represent respectively the reference solution and that which has made use of a particular ABC. The value $t_f$ is the final time of simulation.

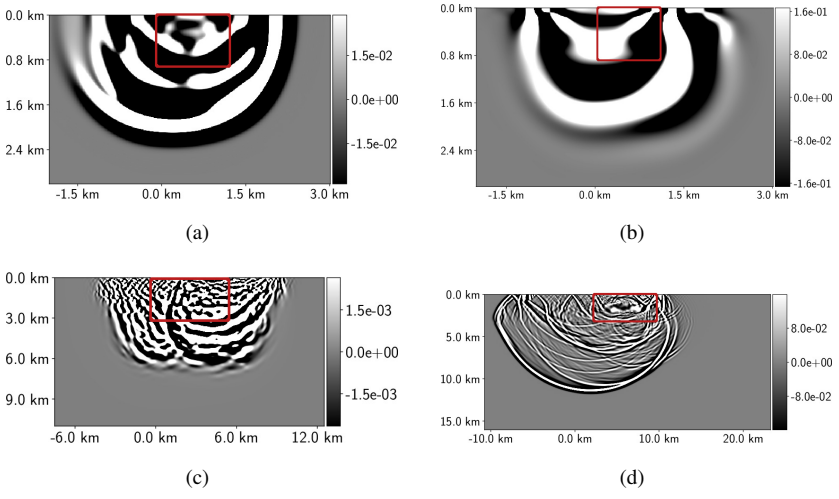



**Figure 5.** Reference solutions of the forward wave equation ($u$): (a) circle velocity model at $t = 1s$; (b) horizontal layers velocity model at $t = 1s$; (c) part o Marmousi velocity model at $t = 6.4s$; (d) 2D SEG/EAGE salt velocity model at $t = 6.4s$. The regions inside the red square are the physical domains, with the red lines indicating the boundaries ($\partial\Omega_0$), and the regions outside are the extended domains.

The ABCs were applied to domain extensions defined in terms of the physical domain length in $x$ direction, $l_x = |x_F - x_I|$. The extension width $l_w$ was set as a percentage $p$ of $l_x$, i. e., $l_w = (p/100) \times L_x$. Moreover, the same extension $l_w$ applies to the depth of the domain, in the $z$ direction. The range of $p$ was taken to be $1 < p < 20$. Previous works (Gao et al., 2017; Liu and Sen, 2018), have taken the number of points in the domain extension $pn_e$, instead, as a measure of its size. Then, $pn_e$ was picked between $5\%$ and $10\%$ of the number of points in the physical domain. Yet, that also meant an extension between $5\%$ and $10\%$ of the original domain length, since square domains with uniform grid spacing were usually adopted to carry out the analyses (Gao et al., 2017; Liu and Sen, 2018). Therefore, the range of domain extensions here is consistent with that of previous works.

To exemplify the choice of $l_w$, consider the domain of the circle velocity model (plotted in Figure 4(a)). In this case, $l_x = 1Km$ and, at $p = 2$, the width $l_w = (p/100) \times l_x = (2/100) \times 1Km = 20m$. At $p = 2$, the extended domain has the width $l_w = 20m$ in $x$ and $z$ directions, which means $pn_e = 2$ for the mesh grid $\Delta x = \Delta z = 10m$.

In the extended domain, the velocity model $c$ was built by employing a constant extrapolation of the physical values of $c$ in the boundary points $x \in [x_I, x_F]$ and $z = z_F$, $x = x_I$ and $z \in [z_I, z_F]$, and $x = x_F$ and $z \in [z_I, z_F]$.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions
Open Access
EGU

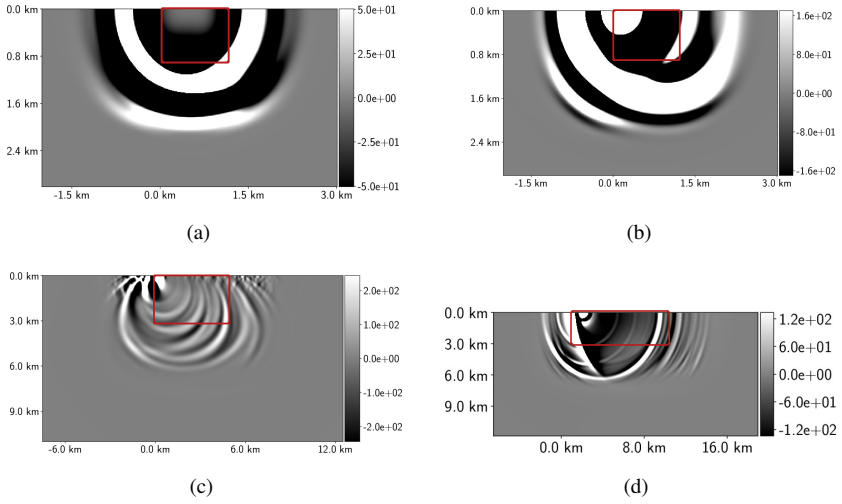



**Figure 6.** Reference solutions of the adjoint $u^{\dagger}$ wave equation related to the velocity models: (a) circle velocity model at $t = 1s$; (b) horizontal layers at $t = 1s$; (c) part o Marmousi velocity model at $t = 6.4s$; and (d) 2D SEG EAGE salt at $t = 5s$. The regions inside the red square are the physical domains, with the red lines indicating the boundaries ($\partial\Omega_0$), and the regions outside are the extended domain.

### 6.1 Forward Wave Equation

As a first step, a fixed $l_w$ and various frequency peaks $f$ of a Ricker wavelet are considered for an error analysis. The extended domain width was chosen to keep the major portion of the curves $\log_{10} E(u_{ref}, u)$ below 1, where $E(u_{ref}, u)$ is given by eq. (52).

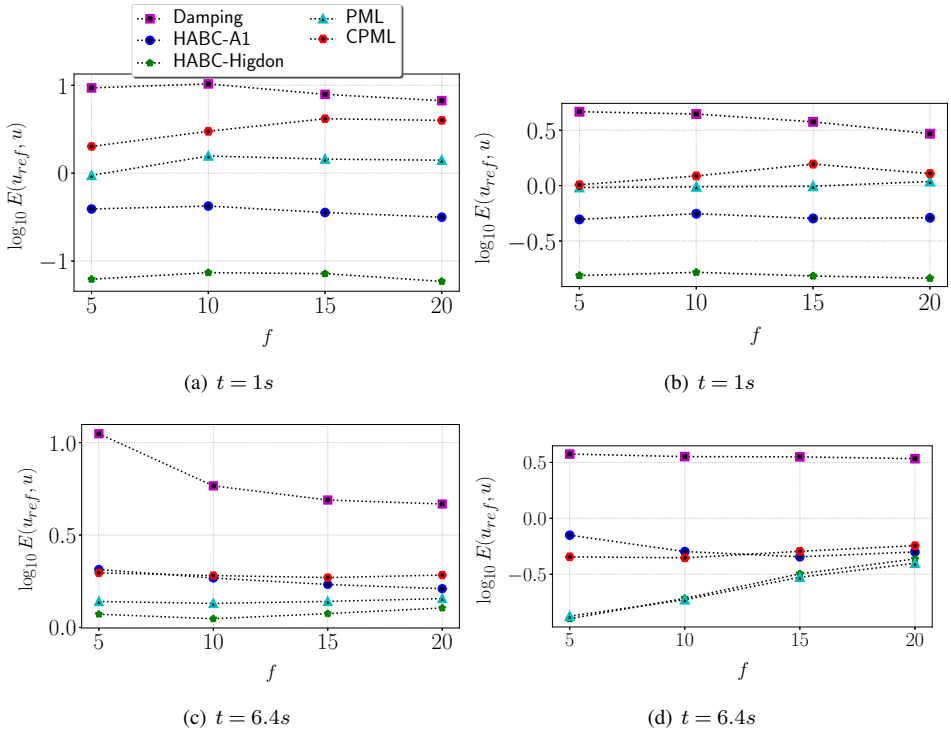
Figure 7 depicts $\log_{10} E(u_{ref}, u) \times f$. In essence, it shows that the frequency peak bears on the effectiveness of the ABCs methods. One notices in Figs. 7 (c) and (d) that, for more realistic velocity models (part of Marmousi and 2D SEG/EAGE salt), the error grows with $f$ for the PML, CPML and HABC-Higdon. For simpler models such as the circle and the horizontal layers, the error also exhibits a slight growth, but only for the PML and CPML methods. It is also clear in Fig. 7 that the HABC-Higdon incurs smaller errors. For the more realistic models, PML and CPML have gotten closer to HABC-Higdon. Whereas the damping method consistently exhibits the highest errors.

In order to ascertain whether similar behavior would be seen for different sizes of domain extension, the next test assesses the ABCs performance as a function of $l_w$. On accounting for the previous evidence of the peak frequency $f$ effects upon performance, this test only includes the more realistic models, namely, the part of Marmousi and the 2D SEG/EAGE salt.

Figure 8 depicts $\log_{10} E(u_{ref}, u)$ as a function of $p$, and the $l_w$ thereof. The errors decrease as $p$ increases. Once again, the relative errors grow with $f$ for PML, CPML, and HABC-Higdon alike. This behavior is observed in both Marmousi and 2D SEG/EAGE salt velocity models. On the other hand, the errors decrease as $f$ grows, for the Damping method. The Damping and HABC-A1 errors approximate those of the other ABCs (PML, CPML and HABC-Higdon) as $f$ grows, especially for the part of Marmousi model. Figure 8 (a) shows the PML and CPML errors much closer to each other. Whereas in 2D salt model,

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions




(a) $t = 1s$ (b) $t = 1s$

(c) $t = 6.4s$ (d) $t = 6.4s$

**Figure 7.** Error curves of the forward solution ($u$) with respect the frequency peak $f$ of Ricker wavelet. The analyses considered the wave solutions for the velocity models: circle (a); heterogeneous model built with horizontal layers (b); part of Marmousi model (c); 2D SEG/EAGE salt velocity model (d).

Figure 8 (b) shows PML lower error than that of the CPML. In all cases, the HABC-Higdon incurs errors that are either similar or smaller than those of the PML and CPML, while those of the Damping method are the highest.

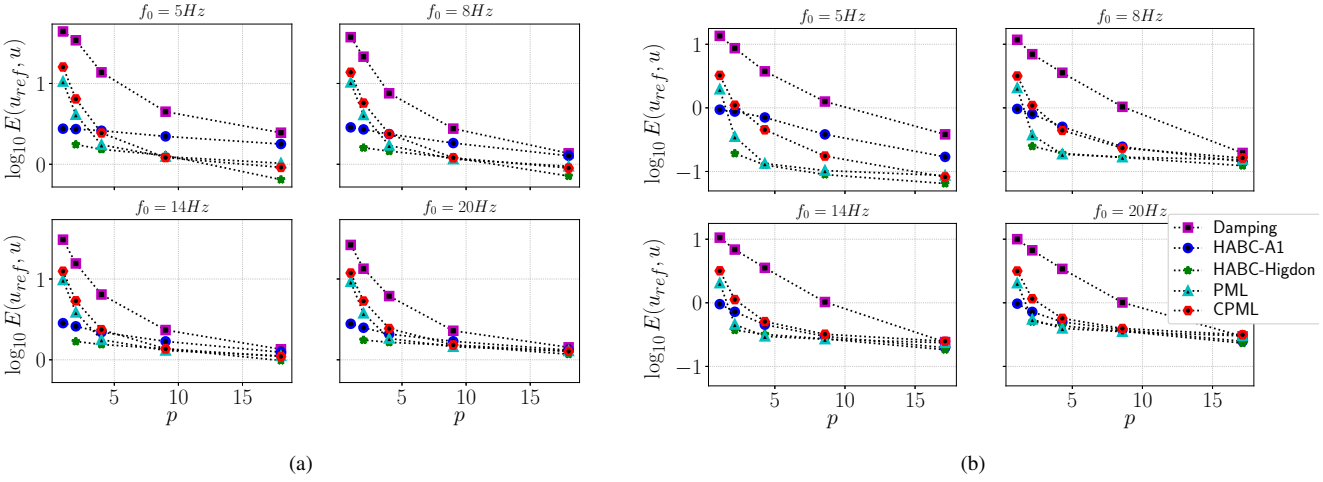
### 6.2 Adjoint Wave Equation

The previous section presented analyses of the ABCs' performance in the context of the forward wave equation. Here, we consider their performance in the adjoint wave equation, which is referred to as the backward problem.

The adjoint forcing term is given by $d = (u^{obs} - u)\boldsymbol{\delta}(\mathbf{x} - \check{\mathbf{x}})$. In that expression, $u^{obs}$ represents the observed, recorded, data from the true velocity model, whereas $u$ stands for an initial velocity model— henceforth termed guess velocity model. Any nonzero difference between them, and the forcing term, $d$, will give rise to a non-trivial solution. Hence, in the analysis that follows, $u^{obs}$ is based on the models shown in Figure 4 (true velocity), while $u$ is computed from the models shown in Figure 9 (guess velocity).

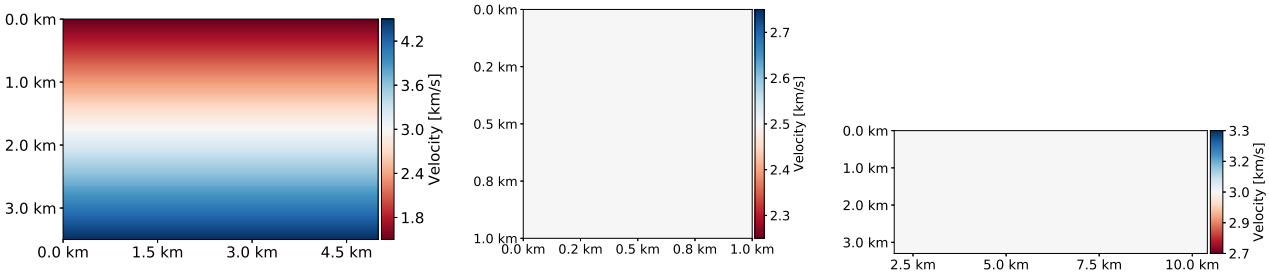
The following steps are taken to assess the adjoint ABCs performance: A reference adjoint solution is computed in the reference enlarged domain, to avoid spurious reflections; next, adjoint solutions subject to the various ABCs are computed, and their errors with respect to the reference are evaluated by eq. (52). Figure 10 shows the curves of $\log_{10} E(u^{\dagger}_{ref}, u^{\dagger}) \times f$. In 2D

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


(a) (b)

**Figure 8.** Error curves of the forward solution ($u$) with respect to $p$ (percent of $L_x$) for different peak of frequency $f_0$. The analyses considered the wave solutions for the velocity models: (a) part of Marmousi model; (b) 2D SEG/EAGE salt.




(a) (b) (c)

**Figure 9.** Guess velocity model: (a) part o Marmousi; (b) circle and horizontal layers velocity models; and (c) 2D SEG/EAGE salt.

SEG/EAGE velocity model, the PML, CPML and HABC-Higdon errors grow with $f$. Somewhat surprisingly, the errors get closer than that of the Damping method, for higher values of $f$, as shown in Fig. 10 (d). In circle velocity model, Fig. 10 (a) shows only the PML and CPML errors growing with f. In several cases, HABC-Higdon errors are either close or smaller than the PML errors. Figure 11 presents error curves with respect to the domain extension parameter $p$, for distinct frequency peaks $f$. In all cases, the error diminishes as $p$ increases. For part of Marmousi velocity model, Figure 11 (a) shows the PML, HABCs and Damping errors dropping as $f$ grows, whereas CPML errors rise with $f$. For 2D SEG/EAGE, Figures 11 (b) shows the ABCs errors going up with $f$.

Similar to the forward problem, the frequency peak $f$ bears on the adjoint ABCs' effectiveness, as well. Moreover, the HABC-Higdon error has shown to be either smaller than or close to that of the PML. It also appears that the CPML has not

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

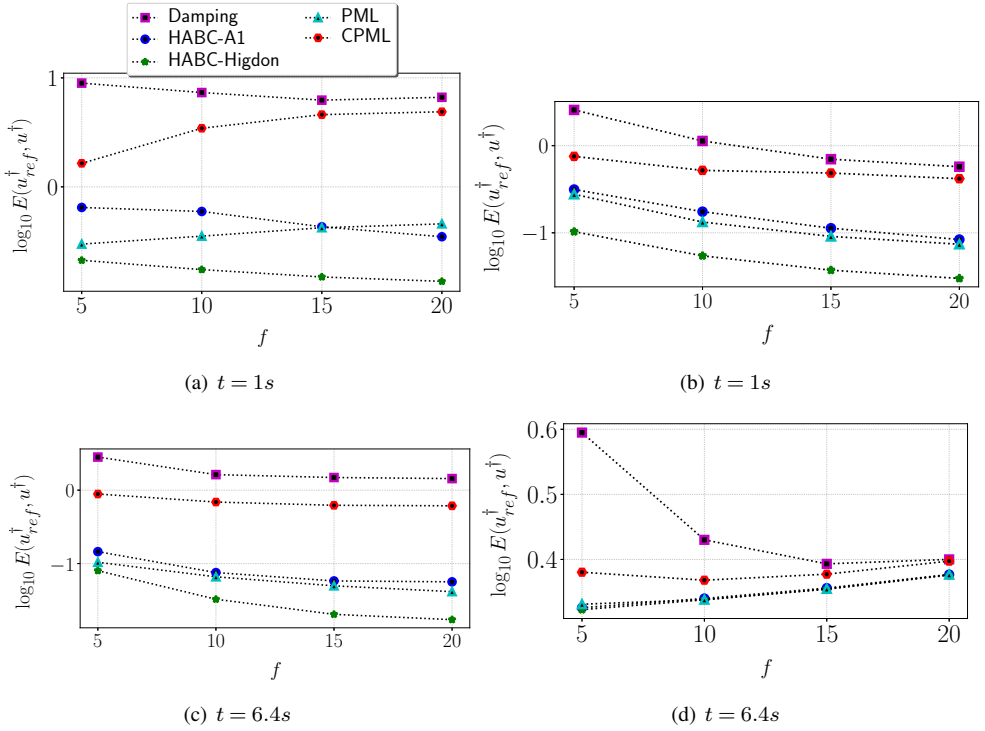



**Figure 10.** Error curves of the adjoint solution $u^{\dagger}$ with respect to peak of frequency. The tests consider the wave solutions for the velocity models: (a) circle; (b) heterogeneous model built with horizontal layers; (c) part of Marmousi model; and (d) 2D SEG/EAGE salt.

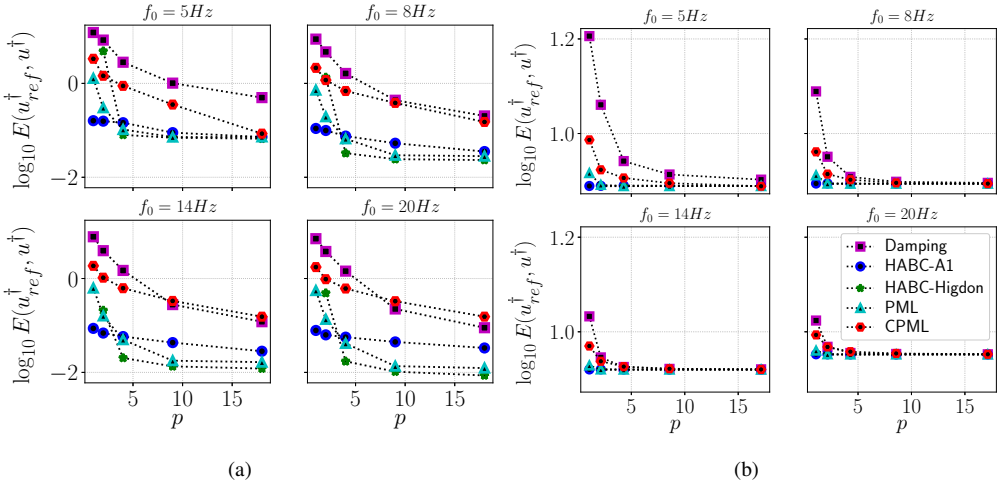



**Figure 11.** Error curves of the adjoint solution $u^{\dagger}$ with respect to $p$ (percent of $L_x$) for different frequency peaks. The tests consider the wave solutions for the velocity models: (a) part of Marmousi model; and (b) 2D SEG/EAGE salt.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

| Peak of memory usage (GB) | time of simulation (s) |
|---|---|
| 0.36 | 16.6 |

**Table 2.** Computational cost related to the execution of the forward solver when there is no employment of any ABCs methods. The velocity model was part of Marmousi with the settings shown in Table 1.

been quite effective in the adjoint problem. That can be noted especially for the part of Marmousi velocity model, where its errors are the highest for higher frequency peaks $f$.

### 6.3 Computational cost: memory usage and time of simulation

Given the above diversity of ABC characteristics, a question naturally arises as to their computational costs and memory usage. This section addresses precisely those topics. To that end, the range $1 \leq p \leq 20$ was adopted to run the forward solver, subject to ABCs. Part of Marmousi was the chosen velocity model for the experiments, with the setup shown in Table 1. Section 5.2 describes the settings, as libraries and machines, used in the computational performance measures.

Table 2 shows the average wall-clock runtime of simulations and the memory usage of a reference case, which used homogeneous Dirichlet boundary conditions (30), and no ABCs were employed. That is, the tests were performed in the physical domain $\Omega_0$ only, without any extensions. Such case is henceforth referred to as the no-ABC case, and it is used as a reference for comparison with cases that are subjected to the various ABC methods.

Figures 12(a) and 12(b) present results of such comparison, in the form of a relative increase in percentage of the time of simulation $t_g$ and memory usage $m_g$ of the ABCs, as compared to the no-ABC reference. Figure 12(a) shows that the growth of memory usage associated with the Damping method remains below $5\%$ within the whole range of domain extensions, $p$. However, as noticed in Figures 13(a) and 13(b), curves of $\log_{10} E(u_{ref}, u) \times m_g$ and $\log_{10} E(u_{ref}, u) \times t_g$ presents the highest errors when the Damping method is employed.

The CPML method is far more expensive, due to the number of additional variables to be solved: at $p \approx 1$, Figures 12(a) and 12(b) show the memory usage increasing more than $10\%$, while time of simulation grows more than $80\%$. On evaluating $\log_{10} E(u_{ref}, u) \times m_g$ and $\log_{10} E(u_{ref}, u) \times t_g$, Figures 13(a) and 13(b) show that the CPML errors are close to the PML errors, but the CPML computational performance has been more expensive than the PML.

The HABCs methods are more expensive than the Damping, but they have shown to attain lower values of $m_g$ and $t_g$ than those of the CPML and PML. In all cases, HABC-Higdon requires more time and memory usage than HABC-A1. However, Figures 13(a) and 13(b) depict the errors of HABC-Higdon smaller than that HABC-A1 errors for $p > 1$. Also, for $p > 1$, HABC-Higdon demands less wall-clock time and memory usage than that of PML and CPML, whereas its performance in decreasing the reflections (measured by $\log_{10} E(u_{ref}, u)$) has been similar.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

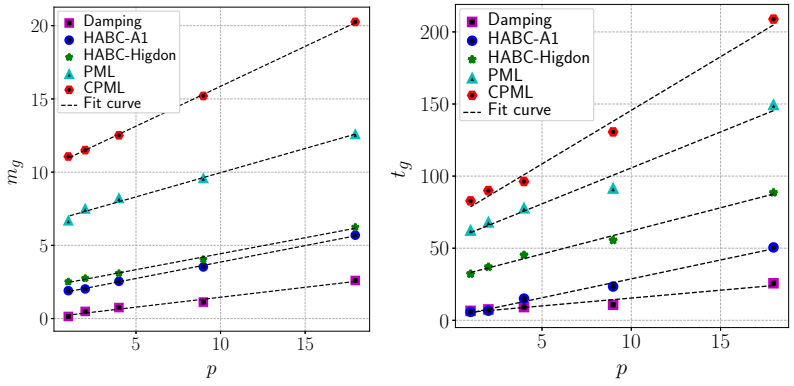



**Figure 12.** The percent growth of memory usage (a) and time of simulation (b), as compared to the no-ABC case. The measures are related to the execution of the forward solver. The velocity model was part of Marmousi with the settings shown in Table 1.

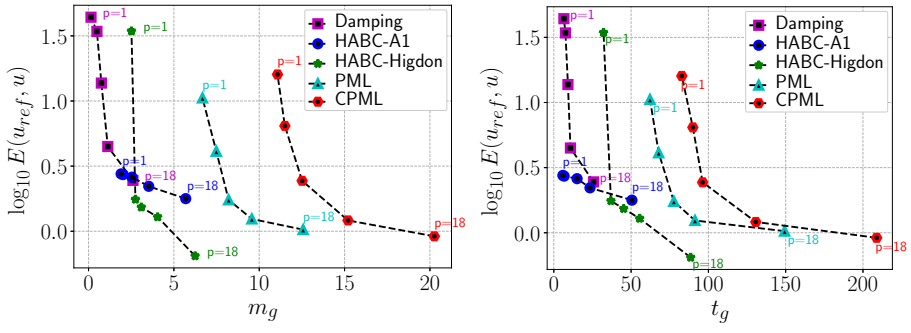



**Figure 13.** Analyses of $\log_{10} E(u_{ref}, u) \times m_g$ (a), and $\log_{10} E(u_{ref}, u) \times t_g$ (b). The measures are related to the execution of the forward solver. The velocity model was part of Marmousi with the settings shown in Table 1, and the peak of frequency $f_0 = 5Hz$.

## 7 Analysis of ABCs in a FWI Problem

Taking into account the effectiveness plus computational time and memory usage, one observes that HABC-Higdon is a proper choice to be employed in an FWI execution. However, the Damping method has presented time of simulation and memory usage much smaller than the other ABCs. Besides that, the PML method is commonly employed in FWI works (Abubakar et al., 2009; Asnaashari et al., 2012; Aghamiry et al., 2019; Ben-Hadj-Ali et al., 2011). Therefore, this section proposes to compare numerical results and the computational cost in FWI using Damping, PML and HABC-Higdon.

The FWI applications take the Marmousi as the true model. True receivers signals are obtained in the reference field, where the domain was extended to avoid spurious reflections.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

The setup used to run this FWI case is presented in Table 3. The Marmousi velocity model and the initial model are respectively illustrated in Figures 15 (a) and (b). The first case evaluates the ABCs performance in FWI for a fixed peak of frequency $f_0 = 7Hz$. A sub-sampling approach with ratio $r = 5$ (which meets the Nyquist criterion) was employed. The search algorithm has been the L-BFGS-B (Byrd et al., 1995), where the stop condition was the number of iterations. The extended domain width $l_w$ was set according to Kaltenbacher et al. (2013), i.e., $l_w = max(c)/f = 4.7/7 \approx 0.67Km$ (equivalent to $p \approx 4$).

Figure 14 shows the misfit and the error of the velocity model $E_c$ of the FWI runs. In particular, the error $E_c$ was computed by the expression:

$$E_c = \frac{||c_{true} - c_{comp.}||_2}{||c_{true}||_2},$$

where $c_{true}$ is the true velocity model, and $c_{comp}$ is the computed velocity model provided by the FWI run. To evaluate $E_c$, the velocity models $c_{true}$ and $c_{comp}$ were both defined in the same mesh setup.

On comparing the performance of the HABC-Higdon to that of the PML, the results in Figures 14 show smaller values of $I(m)$ and $E_c$ for the former. Furthermore, Table 4 shows that the percent growth of memory usage and the wall-clock time related to the HABC-Higdon required significant less time and a slightly small memory usage compared with the PML. The reason for this better performance is basically due to the fact that the HABC-Higdon does not require additional variables and equations to be solved, which incur in both added computational time and memory usage. Therefore, it is proper to conclude that HABC-Higdon has shown better overall performance compared to PML.

Comparing the HABC-Higdon with respect to the Damping scheme, we note that the damping scheme with the same damping layer size as used in the HABC-Higdon ($l_w = 0.67Km$) is considerably faster and uses less memory (see Table 4). This is mainly due to the computational overhead of HABC-Higdon having to successively apply boundaries conditions in nested domains, with added memory use due to a few auxiliary variables used in this nesting procedure. However, Figure 14 shows that $I(m)$ and $E_c$ of the HABC-Higdon (with $l_w = 0.67Km$) are lower than those of the Damping, with the Damping scheme requiring a much larger damping layer to reduce the error. Particularly in Fig. 14 (b), it is clear that, even on increasing $l_w$, the Damping error $E_c$ is still higher than that of the HABC-Higdon. Moreover, Table 4 shows that the percent growth of wall-clock time of simulation and memory usage of the Damping method with $l_w = 2.5Km$ is significantly larger than that of the HABC-Higdon results with $l_w = 0.67Km$. Therefore, HABC-Higdon advantages seem twofold, in that it combines a good performance in mitigating spurious reflections with a relatively low computational cost.

A second case considers the peak of frequency $f_0 = 15Hz$ and applies the multiscale approach in frequency (Bunks et al., 1995). The numerical setup is displayed in Table 3. The cut of frequencies are $f_c = 3Hz$, $f_c = 5Hz$ and $f_c = 8Hz$. The extended domain choice was based in the peak of frequency $f_0 = 15Hz$. Therefore, the extended domain had $l_w = max(c)/f = 4.7/15 \approx 0.32Km$ (equivalent to $p \approx 2$). The initial velocity model used in the FWI executions is displayed in Figure 15(b), and $r = 4$ was set in the sub-sampling approach.

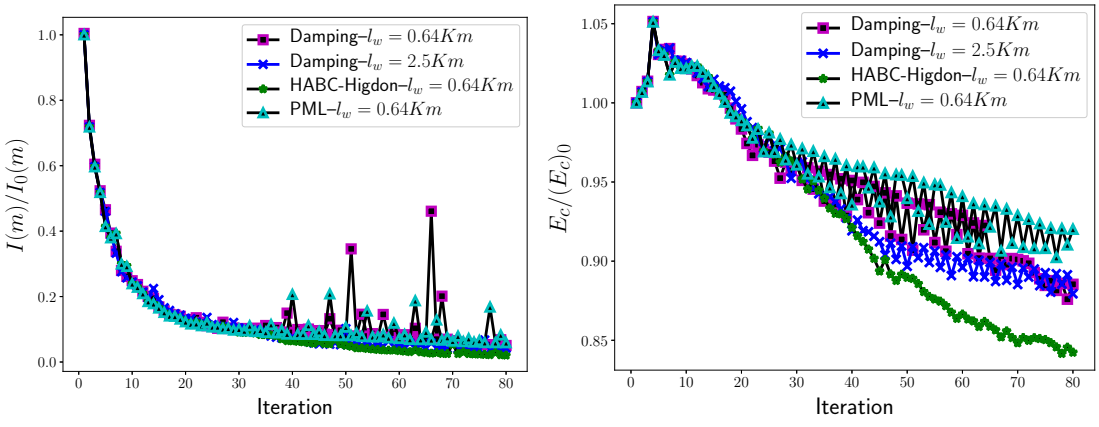
Figures 16 and 17 present the misfit and the error of the velocity model. Once again, FWI with the Damping methods has shown the worst performance, the misfit values and the velocity errors are the highest. FWI executions with HABC-Higdon and PML have presented similar performance. Differences are observed in the velocity errors $E_c$, Figure 17(a) shows smaller

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

| Velocity models | Marmousi |
|---|---|
| Physical domain size | $L_x = 17.0Km$, $L_z = 3.5Km$ |
| Mesh spacing | $\Delta x = \Delta z = 10m$ |
| Total time | $t = 5s$ |
| Source number ($n_s$) | 40 |
| Receiver number ($n_r$) | 850 |
| Sources positions $[x, z](m)$ | $[(100 + i \times 420), 0.125]$, $i = 0, ..., n_s$ |
| Receiver positions $[x, z](m)$ | $[(100 + i \times 20), 0.225]$, $i = 0, ..., n_r$ |

**Table 3.** Setup used to carry out the numerical FWI simulations, which are used to compute the velocity models that are then matched with Marmousi.




(a) (b)

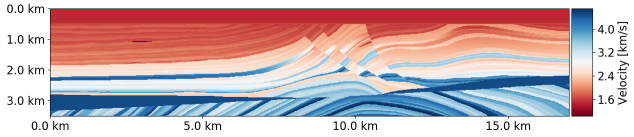
**Figure 14.** Comparisons of the misfit values $I(m)/I_0(m)$ (a) and of the velocity model errors $E_c/(E_c)_0$ (b) for velocity model computation that is then matched with Marmousi, where $I_0(m)$ and $(E_c)_0$ are the values obtained at the first iteration.

| ABC method | Time | Memory |
|---|---|---|
| Damping ($l_w = 0.67Km$) | 14.6% | 25.2 % |
| Damping ($l_w = 2.5Km$) | 94.6% | 112% |
| PML ($l_w = 0.67Km$) | 108% | 29.2% |
| HABC-Higdon ($l_w = 0.67Km$) | 44% | 27% |

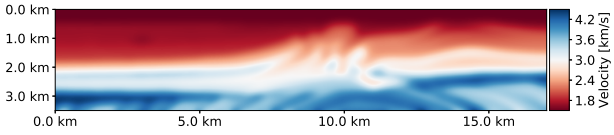
**Table 4.** The percent growth of wall-clock time and RAM memory usage, as compared to the no-ABC case ($l_w = 0.0Km$). The measures are related to a source of a single shot wave of the FWI execution for the velocity model that is matched with Marmousi. In this case, the execution of FWI with no-ABC required 75 seconds of wall clock time, and 5GB of memory usage.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022

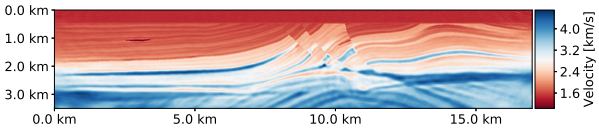

Geoscientific Model Development Discussions




(a) True model




(b) Initial model




(c) Computed model - HABC-Higdon

**Figure 15.** Velocity models comparisons.

| ABC method | Time | memory |
|---|---|---|
| Damping ($l_w = 0.32Km$) | 13.3% | 12.3% |
| PML ($l_w = 0.32Km$) | 94.6% | 14.9% |
| HABC-Higdon ($l_w = 0.32Km$) | 65% | 13.2% |

**Table 5.** The percent growth of wall-clock time and RAM memory usage, as compared to the no-ABC case ($l_w = 0.0Km$). The measures are related to a source of a single shot wave of the FWI execution with the multiscale approach, for the velocity model that is matched with Marmousi. It this case, the execution of FWI with no-ABC required 75 seconds of wall clock time, and 6.11GB of memory usage.

errors associated to HABC-Higdon for cut of frequency $f_c = 3Hz$. Whereas for $f_c = 5Hz$ and $f_c = 8Hz$, $E_c$ associated to the PML method has been smaller.

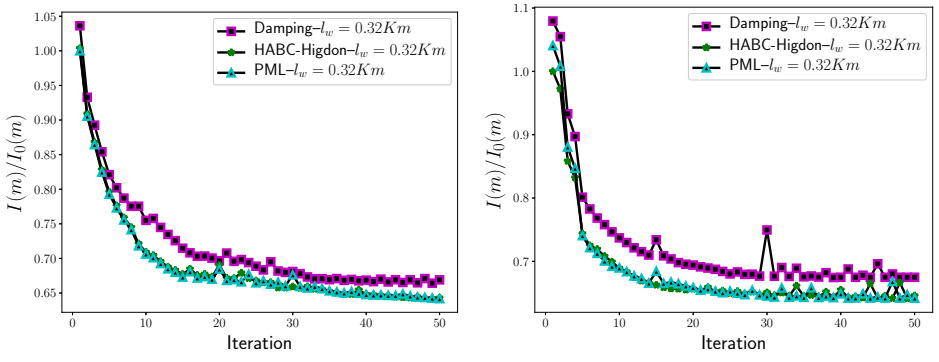
Once again, on evaluating the performance in mitigating the reflections and the computational cost that is presented in Table 5, one verifies the HABC-Higdon as a proper choice to be employed in an FWI execution.

## 8 Conclusions

This work evaluates the effectiveness of the ABCs methods in mitigating spurious boundary reflections, by employing a setting that is usually adopted in FWI applications. The analyses were carried out for the forward and adjoint wave equations, and our findings clearly show that the adjoint problem also experiences spurious boundary reflections. Indeed, that should be expected, owing to the hyperbolic character those equations share with their physical counterparts. In view of such evidence, we have formally derived adjoint boundary conditions that correspond to each of the ABCs. This formulation of forward and adjoint problems, along with their corresponding ABCs, have been extensively tested to assess the effectiveness of the latter. A number of application cases has been run for heterogeneous velocity models, ranging from the simplest models to realistic ones.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions




(a) $f_c = 3Hz$ (b) $f_c = 5Hz$

(c) $f_c = 8Hz$

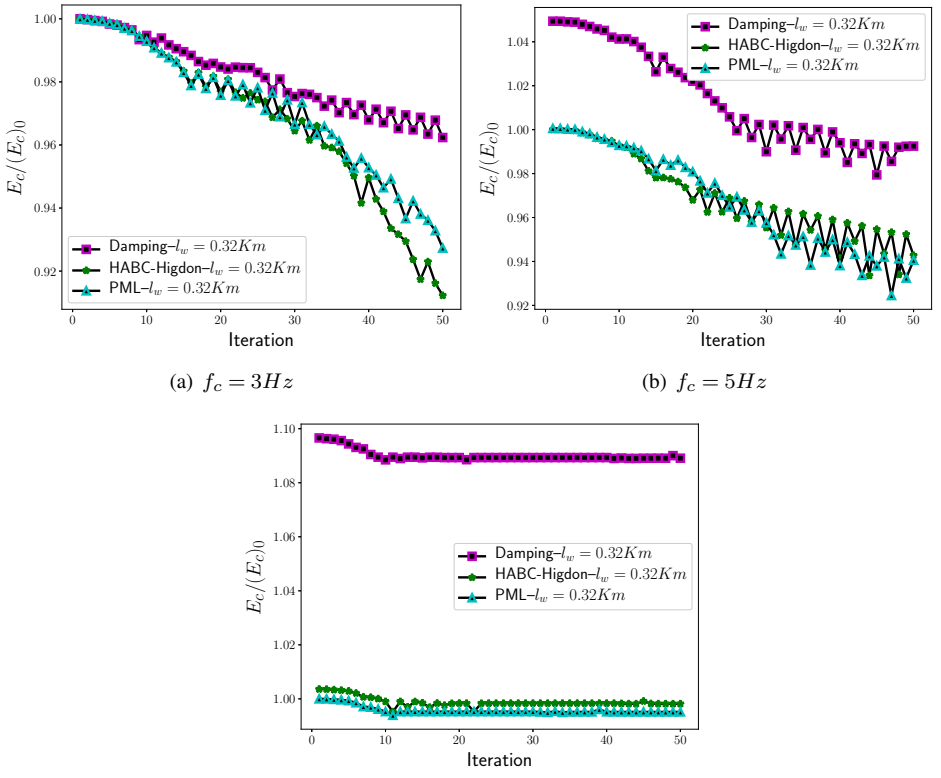
**Figure 16.** Comparisons of the misfit values $I(m)/I_0(m)$ for velocity model computation that is then matched with Marmousi, where the peak of frequency is $f_0 = 15Hz$ and multiscale approach in frequency (Bunks et al., 1995) is employed. $I_0(m)$ is computed at the first iteration, which is the minimal misfit value given by the employment of the Damping, HABC-Higdon and PML methods.

Code development was carried out in the domain specific language (DLS) computational framework, Devito, which allows an ease of implementation of the absorbing conditions described here. Furthermore, these schemes are readily available in the Devito repository (see Devito tutorials on ABCs) to be used in more realistic problems, and they may be adapted to three-dimensional problems by means of symbolic operations, alone.

Analyses of the ABCs' effectiveness in the forward and adjoint problems have shown that the PML and HABC-Higdon are more effective for both of them. On the other hand, the Damping is the least efficient method in attenuating reflections. Figures 16 and 17 depict it as being less effective than the HABC-Higdon, even when an effort is made to improve matters, by increasing the size of the domain extension. The CPML method has presented higher errors than the PML and HABC-Higdon, and it has not kept a pattern, with different effectiveness on the forward and adjoint problems. For instance, Figure 8 shows smaller errors for the CPML than for the HABC-A1 and Damping methods. Yet, for the adjoint problem, Figure 11(a) shows CPML errors to be higher than those of the Damping, as the peak of frequency $f$ increases.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions




(a) $f_c = 3Hz$ (b) $f_c = 5Hz$

(c) $f_c = 8Hz$

**Figure 17.** Comparisons of the misfit values $E_c/(E_c)_0$ for velocity model computation that is then matched with Marmousi, where the peak of frequency is $f_0 = 15Hz$ and multiscale approach in frequency (Bunks et al., 1995) is employed. $(E_c)_0$ is computed at the first iteration, which is the minimal velocity error given by the employment of the Damping, HABC-Higdon and PML methods.

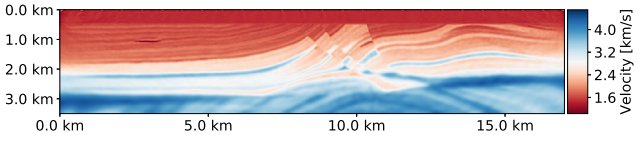



**Figure 18.** Velocity model computed by the FWI execution with HABC-Higdon and a multiscale approach (Bunks et al., 1995).

On evaluating the computational cost of ABCs methods, HABC-Higdon has shown the best performance, since its errors are either close to or smaller than those of the PML in several cases, and its computational cost is lower than the PML, or Damping with larger extensions. A similar conclusion may be drawn for the FWI applications, where the HABC-Higdon has shown to require less memory usage and wall-clock time than the FWI with PML method. On accounting for computational cost and effectiveness, the tests have indicated that the HABC-Higdon also performs better than the Damping method. To be more specific, Table 4 shows the percent of growth of memory usage and wall-clock time of the Damping method to be

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

higher than those of HABC-Higdon, when the extended domain was increased from $l_w = 0.67Km$ to $2.5Km$. In such case, the HABC-Higdon with $l_w = 0.67Km$ was more effective in mitigating reflections than the Damping method with $l_w = 2.5$.

Regarding the extension to 3–D problems, previous works (Grote and Sim, 2010; Xie et al., 2014) on PML methods did not report differences in attenuation effectiveness, on going from 2–D to 3–D domain. Owing to the symmetric nature of the acoustic wave propagation, we also expect the effectiveness of the ABCs in 3–D problems to be similar to those shown here. The computational cost, however, may be considerably different, which should, in principle, raise the differences between them. The computational costs in the 3–D applications may be estimated by using data from Tables 4 and 5. For instance, take the computational cost from Table 5 as a basis, with the third additional dimension, the $y$ direction of length of $l_y$, which is discretized for finite differences for a grid spacing of $\Delta y$. In this case, the growth of computational costs of an FWI application (and memory usage) becomes at least $l_y/\Delta y$ times larger than those of the 2–D using the no-ABC case, damping, and HABC-Higdon. However, with the Damping scheme requiring a larger extension region, the memory savings of the HABC-Higdon in 3–D problems become even more evident than in the 2–D. With both Damping and HABC-Hybrid schemes, no additional variables or equations are required, on going from 2 to 3 dimensions, whereas the PML does entail two additional PDEs in that case. This makes the rise in computational costs of the PML even higher when one adds the third dimension, when compared to the corresponding cases of the no-ABC or the HABC-Higdon, as can be seen in Tables 4 and 5.

To conclude, while this work has adopted synthetic velocity models to generate the true seismogram data in the FWI problems, our finds regarding the ABCs are expected to hold for real seismograms just as well, since the spurious reflections arise on the computational solver, where artificial outer boundaries are imposed. Hence, they are just as prone to exhibit spurious reflections as the above tests have shown. To the best of our knowledge, the effects the ABCs may have upon the adjoint problem, and on the FWI thereof, have not yet received attention in the literature. In addition to that, this work contributes to implementing further options of ABCs in Devito, and presents an HABC approach based on the Higdon method Higdon (1986), that has shown to be more effective, and computational more efficient, than the well-know and used PML method.

*Code availability.* The reproducible code can be founded in the following Zenodo directory https://zenodo.org/record/6003038#.YgKWl3XMJhF.

*Data availability.* The velocities data sets used in this work were created synthetically (Circle) or obtained in a open data set repository (Marmousi (Martin et al., 2006) and 2D/SEG EAGE (Aminzadeh and Brac, 1997) https://wiki.seg.org/wiki/Open_data.)

*Author contributions.* Dolci, Daiane I.: Conceptualization, Methodology, Software, Investigation, Writing–Original draft preparation
Silva, Felipe .A.G: Conceptualization, Methodology, Software, Validation, Writing–Original draft preparation
Peixoto, Pedro S.: Supervision, Writing–Reviewing and Editing Review, Resources, Funding acquisition;
Volpe, Ernani V.: Supervision, Writing–Reviewing and Editing Review, Resources, Funding acquisition.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


*Competing interests.* The authors report no conflict of interest.

*Acknowledgements.* This research was carried out in association with the ongoing R& D project registered as ANP 20714-2 *STMI - Software Technologies for Modelling and Inversion, with applications in seismic imaging* (USP/Shell Brasil/ANP).

The authors would also like to acknowledge and express their deepest appreciation for the invaluable contribution the late Prof. Saulo R. M. de Barros (∗1958 – †2021 ) has made to this work. Prof. Saulo has done so much for our research. Sadly, though, he did not live to see its bearing fruit.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022


Geoscientific Model Development Discussions

## References

A. Fichtner, H.-P. Bunge, H. I.: The adjoint method in seismology: I. Theory, Physics of the Earth and Planetary Interiors, 157, 86–104, https://doi.org/10.1016/j.pepi.2006.03.016, 2006.

Abubakar, A., Hu, W., Habashy, T. M., and Van den Berg, P.: Application of the finite-difference contrast-source inversion algorithm to seismic full-waveform data, GEOPHYSICS, 74, https://doi.org/doi.org/10.1190/1.3250203, 2009.

Aghamiry, H. S., Gholami, A., and Operto, S.: Improving full-waveform inversion by wavefield reconstruction with the alternating direction method of multipliers, GEOPHYSICS, 84, 2019.

Aminzadeh, F. and Brac, J. ad Kunz, T.: SEG/EAGE 3-D Salt and Overthrust Models, 1, Distribution CD of Salt and Overthrust models, SEG book series., 1997.

Asnaashari, A., Brossier, R., Garambois, S., Audebert, F., Thore, P., and Virieux, J.: Regularized seismic full waveform inversion with prior model information, GEOPHYSICS, 78, https://doi.org/doi.org/10.1190/geo2012-0104.1, 2012.

Ben-Hadj-Ali, S., Operto, S., and Virieux, J.: An efficient frequency-domain full waveform inversion method using simultaneous encoded sources, GEOPHYSICS, 76, https://doi.org/doi.org/10.1190/1.3581357, 2011.

Berenger, J.-P. et al.: A perfectly matched layer for the absorption of electromagnetic waves, Journal of computational physics, 114, 185–200, https://doi.org/doi:10.1006/jcph.1994.1159, 1994.

Bunks, C., Saleck, F. M., Zaleski, S., and Chavent, G.: Multiscale seismic waveform inversion, GEOPHYSICS, 60, 1457–1473, https://doi.org/doi.org/10.1190/1.1443880, 1995.

Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C.: A limited memory algorithm for bound constrained optimization, SIAM Journal on Scientific and Statistical Computing, 16, 1190–1208, https://doi.org/doi.org/10.1137/0916069, 1995.

Clayton, R. and Engquist, B.: Absorbing boundary conditions for acoustic and elastic wave equations, Bulletin of the seismological society of America, 67, 1529–1540, 1977.

Fichtner, A.: Full seismic waveform modelling and inversion, Springer Science & Business Media, 2010.

Fornberg, B.: Generation of finite difference formulas on arbitrarily spaced grids, Mathematics of computation, 51, 699–706, 1988.

Gao, Y., Song, H., Zhang, J., and Yao, Z.: Comparison of artificial absorbing boundaries for acoustic wave equation modelling, Exploration Geophysics, 48, 76–93, https://doi.org/doi:10.1071/EG15068., 2017.

Grote, M. J. and Sim, S.: Efficient pml for the wave equation, arXiv preprint arXiv, 2010.

Higdon, R. L.: Absorbing boundary conditions for difference approximations to the multidimensional wave equation, Mathematics of computation, 47, 437–459, 1986.

Higdon, R. L.: Numerical absorbing boundary conditions for the wave equation, Mathematics of computation, 49, 65–90, https://doi.org/doi:10.1090/S0025-5718-1987-0890254-1, 1987.

Kaltenbacher, B., Kaltenbacher, B., and Sim, I.: A modified and stable version of a perfectly matched layer technique for the 3-d second order wave equation in time domain with an application to aeroacoustics, J. Comput. Phys., 235, 407–422, https://doi.org/doi.org/10.1016/j.jcp.2012.10.016, 2013.

Kukreja, N., Louboutin, M., Vieira, F., Luporini, F., Lange, M., and Gorman, G.: Devito: Automated fast finite difference computation, in: 2016 Sixth International Workshop on Domain-Specific Languages and High-Level Frameworks for High Performance Computing (WOLFHPC), pp. 11–19, IEEE, 2016.

https://doi.org/10.5194/gmd-2022-48
Preprint. Discussion started: 31 March 2022




Liu, Y. and Sen, M. K.: A hybrid scheme for absorbing edge reflections in numerical modeling of wave propagation, Geophysics, 75, A1–A6, https://doi.org/doi:10.1190/1.3295447, 2010.

Liu, Y. and Sen, M. K.: A hybrid absorbing boundary condition for elastic staggered-grid modelling, Geophysical Prospecting, 60, 1114–1132, https://doi.org/doi:10.1111/j.1365-2478.2011.01051.x, 2012.

Liu, Y. and Sen, M. K.: An improved hybrid absorbing boundary condition for wave equation modeling, Journal of Geophysics and Engineering, 15, 2602–2613, 2018.

Louboutin, M., Lange, M., Luporini, F., Kukreja, N., Witte, P. A., Herrmann, F. J., Velesko, P., and Gorman, G. J.: Devito (v3.1.0): an embedded domain-specific language for finite differences and geophysical exploration, Geoscientific Model Development, 12, 1165–1187, https://doi.org/10.5194/gmd-12-1165-2019, 2019.

Luporini, F., Lange, M., Louboutin, M., Kukreja, N., Hückelheim, J., Yount, C., Witte, P., Kelly, P. H. J., Herrmann, F. J., and Gorman, G. J.: Architecture and performance of Devito, a system for automated stencil computation, CoRR, abs/1807.03032, http://arxiv.org/abs/1807.03032, 2018.

Martin, G. S., Wileya, R., and Kurt, J.: Marmousi2 : An elastic upgrade for Marmousi, The Leading Edge, 25, 156–166, https://doi.org/doi.org/10.1190/1.2172306, 2006.

Mora, P.: Nonlinear two-dimensional elastic inversion of multioffset seismic data, Geophysics, 52, n.9, 1211, 1987.

Pasalic, D. and McGarry, R.: Convolutional perfectly matched layer for isotropic and anisotropic acoustic wave equations, in: SEG Technical Program Expanded Abstracts 2010, pp. 2925–2929, Society of Exploration Geophysicists, 2010.

Plessix, R.-E.: A review of the adjoint-state method for computing the gradient of a functional with geophysical applications, Geophys. J. Int, 167, 495–503, 2006.

Sochacki, J., Kubichek, R., George, J., Fletcher, W., and Smithson, S.: Absorbing boundary conditions and surface waves, Geophysics, 52, 60–71, https://doi.org/doi:10.1190/1.1442241, 1987.

Tarantola, A.: Inversion of seismic reflection data in the acoustic approximation, GEOPHYSICS, 49, 1259–1266, https://doi.org/doi.org/10.1190/1.1441754, 1984.

Tarantola, A.: A strategy for nonlinear elastic inversion of seismic reflection data, GEOPHYSICS, 51, 1893–1903, https://doi.org/10.1190/1.1442046, 1986.

Tarantola, A.: Theoretical background for the inversion of seismic waveforms including elasticity and attenuation, Pure and Applied Geophysics PAGEOPH, 128, n. 1–2, 365–399, 1988.

Virieux, J. and Operto, S.: An overview of full-waveform inversion in exploration geophysics, GEOPHYSICS, 74, https://doi.org/doi.org/10.1190/1.3238367, 2009.

Wiki, S.: Ricker wavelet, https://wiki.seg.org/wiki/Dictionary:Ricker_wavelet, 2019.

Xie, Z., Komatitsch, D., Martin, R., and Matzen, R.: Improved forward wave propagation and adjoint-based sensitivity kernel calculations using a numerically stable finite-element PML, Geophysical Journal International, 198, 1714–1747, https://doi.org/doi:10.1093/gji/ggu219, 2014.