Geophysical Prospecting

EAGE EUROPEAN ASSOCIATION OF GEOSCIENTISTS & ENGINEERS

Geophysical Prospecting, 2011, **59**, 834–856 doi: 10.1111/j.1365-2478.2011.00966.x

Three-dimensional parallel frequency-domain visco-acoustic wave modelling based on a hybrid direct/iterative solver

Florent Sourbier,[1] Azzam Haidar,[2] Luc Giraud,[3] Hafedh Ben-Hadj-Ali,[1†] Stéphane Operto[1*] and Jean Virieux[4]

[1]*GéoAzur, CNRS - UNSA - IRD - OCA, La Darse BP 48, 06230 Villefranche-sur-mer, France,* [2]*Innovative Computing Laboratory, Department of Electrical Engineering and Computer Science, University of Tennessee, Suite 413 Claxton, 1122 Volunteer Blvd, Knoxville, TN 37996-3450, USA,* [3]*INRIA, joint INRIA-CERFACS Laboratory, Avenue G. Coriolis, 31057 Toulouse Cedex 1, France, and* [4]*Institut des Sciences de la Terre (ISTerre), Université Joseph Fourier, CNRS, BP53, 38041 Grenoble Cedex 9, France*

Received September 2010, revision accepted March 2011

ABSTRACT

We present a parallel domain decomposition method based on a hybrid direct-iterative solver for 3D frequency-domain modelling of visco-acoustic waves. The method is developed as a modelling engine for frequency-domain full waveform inversion. Frequency-domain seismic modelling reduces to the solution of a large and sparse system of linear equations, resulting from the discretization of the heterogeneous Helmholtz equation. Our approach to the high-performance, scalable solution of large sparse linear systems is to combine direct and iterative methods. Such a hybrid approach exploits the advantages of both direct and iterative methods. The iterative component uses a small amount of memory and provides a natural way for parallelization. The direct part has favourable numerical properties for multiple right-hand side modelling. The domain decomposition is based upon the algebraic Schur complement method, which allows for the iterative solution of a reduced system, the solution of which is the wavefield at the interfaces between the subdomains. Once the interface unknowns have been computed, the wavefield at the interior of each subdomain is efficiently computed by local substitutions. The reduced Schur complement system is solved with the generalized minimum residual method and is preconditioned by an algebraic additive Schwarz preconditioner. A direct solver is used to factorize local impedance matrices defined on each subdomain. Theoretical analysis shows that the time complexity of the hybrid solver is the same as that of iterative solver and time-domain approaches for single frequency modelling. Simulations are performed in the SEG/EAGE overthrust and the salt models for frequencies up to 12.5 Hz. The number of iterations increases linearly with the number of subdomains for a given computational domain but the elapsed time of the iterative resolution remains almost constant. The number of iterations also increases linearly with frequencies, when the grid interval is adapted to the frequencies and the size of the subdomains is kept constant over frequency. These results make the cost of the hybrid solver of the same order as that of finite-difference time-domain modeling for one-frequency modelling. Although the hybrid approach allows one to tackle larger problems than the direct-solver approach, further improvements are needed to mitigate the computational

†Now at: Total, Pau, France
*E-mail: operto@geoazur.obs-vlfr.fr

burden of the iterative component in the context of multisource modelling. On the numerical side, the use of block iterative solvers and of incremental two-level deflating preconditioners and on the parallel implementation side, the use of two levels of parallelism in the domain decomposition method should mitigate this computational burden.

Key words: Preconditioners, Overthrust.

INTRODUCTION

An efficient algorithm for 3D visco-acoustic full wave modelling is a central ingredient for full waveform inversion methods. Full waveform inversion is a data matching procedure that seeks to reconstruct one or several physical properties of the subsurface, which govern the propagation of seismic waves (see Virieux and Operto (2009) for a recent review). Since the pionneering work on full waveform inversion in the 1980's (Tarantola 1984; Lailly 1984), full waveform inversion has been developed both in the time domain and in the frequency domain. The frequency domain provides a natural framework to design multiscale imaging through successive inversion of increasing frequencies, which is useful to mitigate the non-linearity of the inverse problem (Pratt and Worthington 1990; Pratt 1999). Moreover, computationally-efficient frequency-domain full waveform inversion algorithms can be designed by limiting the inversion to a few discrete frequencies when wide-aperture acquisition geometries are considered (Sirgue and Pratt 2004). The relevance of the efficient frequency-domain full-waveform inversion was demonstrated on 3D real data case studies by Plessix (2009), Plessix and Perkins (2010) and Sirgue *et al.* (2010).

In the frequency domain, wave modelling reduces to the solution of a large and sparse system of linear equations per frequency, its right-hand side is the monochromatic source and its solution is the monochromatic wavefield (Marfurt 1984). The complex-valued matrix, which results from the discretization of the time-harmonic wave equation (i.e., the heterogeneous Helmholtz equation), is generally referred to as the impedance matrix. In the context of full waveform inversion, this system must be solved efficiently for a large number of sources. This prompts many authors to solve the system with sparse direct solvers for 2D problems (Pratt and Worthington 1990). A direct solver first performs one lower-upper triangular decomposition of the sparse impedance matrix followed by the forward-backward substitution step for each right-hand side (Duff, Erisman and Reid 1986). The drawbacks of the direct approach are the computational and the memory cost of the lower-upper decomposition, which results from the fill-in of the impedance matrix and the limited scalability of sparse direct solvers on large-scale distributed-memory computational platforms (by scalability we mean the ability of the algorithm to efficiently use an increased number of processors for a problem of a given size; nowadays referred to as strong scalability). Accurate finite-difference stencils with local spatial support spanning over two grid intervals have been developed to minimize the numerical bandwidth of the impedance matrix and, hence, its fill-in during the lower-upper decomposition (Jo, Shin and Suh 1996; Stekl and Pratt 1998; Hustedt, Operto and Virieux 2004). The direct-solver approach has been shown to be the most efficient one for 2D acoustic full waveform inversion, when a few frequencies need to be modelled for a large number of sources. For 3D problems, the time and memory complexities of the direct solvers dramatically increase ($\mathcal{O}(N^6)$ and $\mathcal{O}(N^4)$, respectively, where N denotes the size of a N^3 cubic grid) and limit the size of the applications that can be performed with such approaches (by complexity we mean the increase of the computational cost of the modelling with the size of the problem) (Operto *et al.* 2007). However, Brossier *et al.* (2010) recently showed that visco-acoustic modelling can be efficiently performed with the MUMPS direct solver (MUMPS-team 2009) in the SEG/EAGE overthrust and salt models for frequencies of the order of 7 Hz on a limited number of message passing interface processes with a significant amount of shared memory per process (15 Gbytes). A new approach, which exploits the low-rank property of the Helmholtz equation for compression, has also been recently proposed to mitigate the memory and time complexities of the LU decomposition performed with the multifrontal method (Wang, Xia and de Hoop 2010).

An alternative approach for 3D frequency-domain visco-acoustic wave modelling relies on preconditioned iterative methods (Saad 2003). Unlike direct solution methods that seek to compute the exact solution of the system (assuming no rounding errors) by a finite sequence of operations,

iterative methods start from an initial guess of the solution to find successive approximations of the solution through a projection process. Iterative methods for the Helmholtz equation are generally based on preconditioned Krylov subspace methods such as the generalized minimal residual method (Saad and Schultz 1986) and the biconjugate gradient stabilized method (Van der Vost 1992). A central ingredient of iterative methods is the preconditioner, which is an operator that transforms the original linear system into another one having the same solution but better spectral properties (e.g., clustering of the eigenvalues away from the origin). Designing an efficient preconditioner for the Helmholtz equation is a difficult issue because the impedance matrix is indefinite with positive and negative eigenvalues along the real axis. Shifting and clustering eigenvalues of the impedance matrix in the positive part of the real axis should result in faster convergence as shown by Magolu monga Made, Beauwens and Warzée (2000). This clustering has been achieved by Plessix 2007, Erlangga, Oosterleeand and Vuik (2006), Riyanti *et al.* (2006) and Riyanti *et al.* (2007) by using as a preconditioner the approximate solution of the damped wave equation computed with one cycle of multigrid. Here, damping the wave equation is performed by adding an imaginary term to the Laplacian of the time-harmonic wave equation. With such a preconditioner, it was shown that the number of iterations increases linearly with frequency when the grid interval of the finite-difference grid increases linearly with frequency. The linear increase of iterations with frequency makes the time $\mathcal{O}(N^4)$(Plessix 2007). Alternatively, Pinel (2010) and Calandra *et al.* (2008) proposed to use only two levels in the multigrid preconditioner without a complex shift. The solution on the coarser grid can be computed with a sparse direct solver for sufficiently small problems (Duff *et al.* 2007) or inexactly using an iterative solver with a coarse stopping criterion of iterations (Pinel 2010). In this latter case, the method is referred to as the perturbed two-level method. The preconditioner based on a shifted Laplacian has been further improved by Erlangga and Naggen (2008) who developed a multigrid-multilevel Krylov method based on a deflated preconditioner. With such an approach, the eigenvalues are efficiently clustered away from zero almost independently of the frequency (i.e., of the grid interval). In this case, the time complexity of the iterative solver reduces to $\mathcal{O}(N^3)$. Compared to direct approaches, iterative solvers do not embed any source-independent tasks. Therefore, the time complexity of iterative methods generally increases linearly with the number of sources. Of note, the use of a block iterative solver should allow for a more efficient solving of the linear systems with multiple right-hand sides (Simoncini and Gallopoulos 1996). Alternatively, efficient parallelism for multi-source simulations can be implemented by a distribution of sources over processors because iterative solvers are not memory demanding (Plessix 2009).

A third approach to compute a monochromatic wavefield consists of using an explicit time-marching algorithm and extracting the monochromatic wavefields by discrete Fourier transform integrated within the loop over time steps (Sirgue, Etgen and Albertin 2008) or by phase sensitivity detection once the steady-state regime has been reached (Nihei and Li 2007). With such an approach, the time complexity is $\mathcal{O}(N^4)$, assuming that the number of time steps scales with N. One advantage of this approach is that several monochromatic wavefields can be extracted at a limited cost by discrete Fourier transform. Like iterative methods, time-domain modelling does not embed source-independent tasks and can be parallelized by a distribution of shots over processors for multi-source simulations. A drawback of time-domain methods is the more difficult and computationally expensive implementation of attenuation effects.

In this study, we present an algebraic domain decomposition method for 3D visco-acoustic wave modelling based on a hybrid direct-iterative solver. The hybrid domain decomposition method will be referred to as hybrid solver modelling in the following. The method can be viewed as an intermediate approach between the direct and iterative approaches. The governing idea is to solve with an iterative solver a reduced system, the so-called Schur complement system, for the unknowns located at the interfaces between the subdomains (Saad 2003; Smith, Bjorstad and Gropp 1996). This reduced system must be solved for each source. Once the interface unknowns are computed, the interior unknowns located in the interior of each subdomain are computed efficiently by local substitutions. A sparse direct solver is used to factorize the local impedance matrices defined on each subdomain. The preconditioner of the Schur complement system is an algebraic additive Schwarz preconditioner (Carvalho, Giraud and Meurant 2001) built from the damped wave equation. The method is purely algebraic in the sense that the preconditioner does not rely on the physics underlying the wave equation.

In the remainder of this paper, we first briefly review the theory and the computer implementation of the hybrid solver modelling method. The reader is referred to Saad (2003), Haidar (2008) and Smith *et al.* (1996) for more details. We estimate a suitable stopping criterion of iterations with numerical experiments and we review the theoretical time and memory complexities of the hybrid solver modelling method. We

validate these theoretical results against numerical simulations performed in the 3D SEG/EAGE overthrust and salt models for frequencies up to 12.5 Hz. We first show that introducing realistic physical attenuation in the model greatly improves the convergence of the iterative solver when the Schwarz preconditioner is built from the damped wave equation. By means of a scalability analysis, we show that the number of generalized minimum residual method iterations increases linearly with the number of subdomains for a computational domain of a given size. However, the computational cost of the source-dependent task remains almost constant, while the cost of the source-independent task decreases rapidly with the number of subdomains. We also show that the number of generalized minimum residual method iterations increases with frequency, when the grid interval is adapted to the frequency and the size of the subdomains is kept constant. We conclude the study by a discussion on the advantages and drawbacks of the hybrid-solver method with respect to alternative modelling methods based on direct solvers, iterative solvers and time-marching algorithms.

METHOD

We shall describe the method we use for solving the Helmholtz equation and discuss the numerical implementation before analysing the theoretical complexity.

Domain decomposition based on the Schur complement approach

We consider the 3D visco-acoustic wave equation in the frequency domain

$$\frac{\omega^2}{\kappa\left(\mathbf{x}\right)}p\left(\mathbf{x},\omega\right)+\nabla\left(\frac{1}{\rho\left(\mathbf{x}\right)}\nabla p\left(\mathbf{x},\omega\right)\right)=-s\left(\mathbf{x},\omega\right), \tag{1}$$

where the density is denoted by $\rho(\mathbf{x})$, the bulk modulus by $\kappa(\mathbf{x})$ and the angular frequency by ω. The monochromatic pressure wavefield and the source are denoted by $p(\mathbf{x}, \omega)$ and $s(\mathbf{x}, \omega)$, respectively. Intrinsic attenuation can be easily implemented in the frequency domain by means of complex-valued wave speeds in the expression of the bulk modulus.

Equation (1) can be recast in matrix form as

$$\mathbf{Ap}=\mathbf{s}, \tag{2}$$

where the complex-valued impedance matrix $\mathbf{A}$ depends on angular frequency ω and medium parameters κ and ρ. Discretization of the wave equation is performed with the compact finite-difference stencil of Operto *et al.* (2007) originally designed for direct methods, and suitable for substructuring methods because its local support allows for minimization of the size (width) of the interface between the neighboring subdomains, unlike high-order finite-difference methods. A discretization criterion of four-grid points per wavelength is used in the following of this study. For 3D problems, the stencil involves 27 coefficients spanning over two grid intervals in the three Cartesian directions, which lead to a numerical bandwith of $\mathbf{A}$ in $\mathcal{O}(N^2)$. Absorbing boundary conditions are implemented with perfectly matched layers (Berenger 1994). We applied a parallel algebraic domain decomposition method based on the Schur complement approach to solve the 3D heterogeneous Helmholtz equation. Here, the domain decomposition method belongs to the class of substructuring methods, where an overlap of order 1 between adjacent subdomains is considered (Smith *et al.* 1996; Saad 2003). The computational domain is decomposed into subdomains, and one processor is assigned to each subdomain. Of note, more sophisticated strategies involving several levels of parallelism can be implemented in domain decomposition methods (Giraud, Haidar and Pralet 2010): few preliminary results will be presented in the final discussion section. The domain decomposition is designed such that all the subdomains have roughly the same size for efficient workload balancing. The Schur complement method splits the unknowns (i.e., the value of the pressure wavefield at each node of the finite-difference grid) into two subsets associated with subdomain interiors and with subdomain interfaces respectively (Fig. 1).

Using the above-mentioned reordering, the linear system, equation (2), can be written in block form as

$$\begin{bmatrix}\mathbf{A}_{\mathcal{I}\mathcal{I}} & \mathbf{A}_{\mathcal{I}\Gamma}\\ \mathbf{A}_{\Gamma\mathcal{I}} & \mathbf{A}_{\Gamma\Gamma}\end{bmatrix}\begin{bmatrix}\mathbf{p}_{\mathcal{I}}\\ \mathbf{p}_{\Gamma}\end{bmatrix}=\begin{bmatrix}\mathbf{s}_{\mathcal{I}}\\ \mathbf{s}_{\Gamma}\end{bmatrix}, \tag{3}$$

where $\mathbf{p}_{\mathcal{I}}$ and $\mathbf{p}_{\Gamma}$ denote the interior and the interface unknowns, respectively. The matrix $\mathbf{A}_{\mathcal{I}\mathcal{I}}$ is a block matrix involving only interior unknowns of subdomains while the matrix $\mathbf{A}_{\Gamma\Gamma}$ involves only interface unknowns. Matrices $\mathbf{A}_{\mathcal{I}\Gamma}$ and $\mathbf{A}_{\Gamma\mathcal{I}}$ give influences of interfaces unknowns for solving interior unknowns and the opposite for the other one.

Eliminating the interior unknowns $\mathbf{p}_{\mathcal{I}}$ from the second block row of system (equation (3)) leads to the reduced system for the interface unknowns $\mathbf{p}_{\Gamma}$

$$\mathbf{Sp}_{\Gamma}=\left(\mathbf{A}_{\Gamma\Gamma}-\mathbf{A}_{\Gamma\mathcal{I}}\mathbf{A}_{\mathcal{I}\mathcal{I}}^{-1}\mathbf{A}_{\mathcal{I}\Gamma}\right)\mathbf{p}_{\Gamma}=\mathbf{s}_{\Gamma}-\mathbf{A}_{\Gamma\mathcal{I}}\mathbf{A}_{\mathcal{I}\mathcal{I}}^{-1}\mathbf{s}_{\mathcal{I}}, \tag{4}$$

where $\mathbf{S}=\mathbf{A}_{\Gamma\Gamma}-\mathbf{A}_{\Gamma\mathcal{I}}\mathbf{A}_{\mathcal{I}\mathcal{I}}^{-1}\mathbf{A}_{\mathcal{I}\Gamma}$ is the so-called Schur complement matrix. Once the Schur complement system has been solved for the interface unknowns, the interior unknowns are

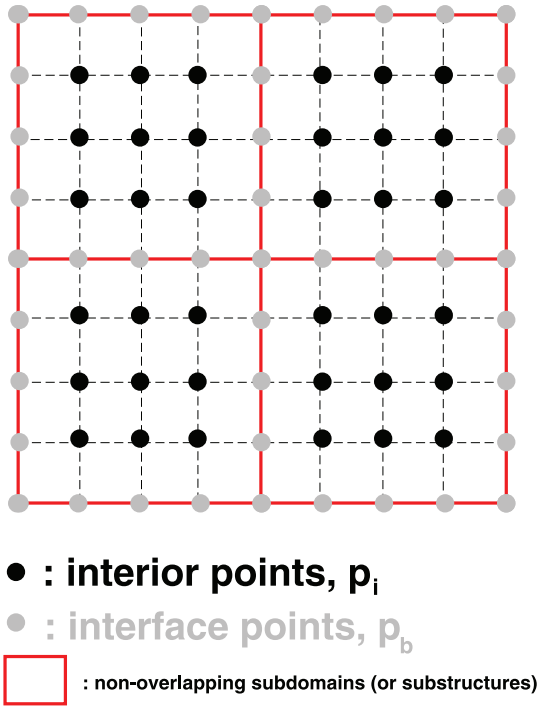

Figure 1 Sketch of the domain decomposition for a 2D finite-difference grid with an overlap of order 1. Four subdomains delineated by the red lines are considered. The nodes of the finite-difference grid are plotted with circles. The black and the grey circles correspond to interior and interface nodes, respectively.

recovered via

$$\mathbf{p}_{\mathcal{I}} = -\mathbf{A}_{\mathcal{I}\mathcal{I}}^{-1}\,\mathbf{A}_{\mathcal{I}\Gamma}\,\mathbf{p}_{\Gamma} + \mathbf{A}_{\mathcal{I}\mathcal{I}}^{-1}\,\mathbf{s}_{\mathcal{I}}. \tag{5}$$

Solving the system involving the block matrix $\mathbf{A}_{\mathcal{I}\mathcal{I}}$

Solving for the interface and interior unknowns, equations (4) and (5), requires solving several solutions of systems involving the matrix $\mathbf{A}_{\mathcal{I}\mathcal{I}}$ that is consequently factorized once using a sparse direct solver. This matrix has a block diagonal structure, where each diagonal block corresponds to the local impedance matrix built on one subdomain. Therefore, the solution of any system involving $A_{\mathcal{I}\mathcal{I}}$ translates into the solution of N_P independent systems involving the local impedance matrices. We solve the systems involving $A_{\mathcal{I}_i\mathcal{I}_i}$ on each processor using the sparse direct solver MUMPS (MUMPS-team 2009).

Solving the Schur complement system with a parallel iterative solver

The Schur complement system is solved with the generalized minimum residual Krylov-subspace method (Saad and Schultz 1986). The algorithm of the generalized minimum residual method is given in Saad (2003). The most demanding task performed at each iteration of the generalized minimum residual method is the computation of the matrix vector product $\mathbf{S}\mathbf{p}_{\Gamma}$. In the hybrid solver modelling method, any matrix-vector product involving the Schur complement matrix can be efficiently performed in parallel because the Schur complement matrix is distributed over the processors and never assembled. The Schur matrix can be viewed as the sum of the so-called local Schur complement matrices (Haidar 2008; Giraud *et al.* 2010)

$$\mathbf{S} = \sum_{i=1}^{N_P} \mathcal{R}_{\Gamma_i}^T \mathbf{S}_i \mathcal{R}_{\Gamma_i}. \tag{6}$$

In equation (6), $\mathcal{R}_{\Gamma_i} : \Gamma \rightarrow \Gamma_i$ denotes the canonical pointwise restriction that maps vectors defined on Γ into vectors defined on Γ_i, where Γ_i are the interface nodes that belong to subdomain i. The dense local Schur complement matrix $\mathbf{S}_i$ is built from the local impedance matrix $\mathbf{A}_i$ defined on the subdomain i with Neumann boundary conditions along the four edges of the subdomain. Using the same reordering as for the full matrix, equations (3) and (4), and Saad (2003) leads to:

$$\mathbf{S}_i = \mathbf{A}_{\Gamma_i\Gamma_i} - \mathbf{A}_{\Gamma_i\mathcal{I}_i}\mathbf{A}_{\mathcal{I}_i\mathcal{I}_i}^{-1}\mathbf{A}_{\mathcal{I}_i\Gamma_i}. \tag{7}$$

Of note, a slight modification of $\mathbf{A}_{\Gamma_i\Gamma_i}$ in the expression of $\mathbf{S}_i$ is generally required to satisfy equation (6). In the framework of finite-difference methods, this modification requires to weigh the coefficients of the matrices $\mathbf{A}_{\Gamma_i\Gamma_i}$ such that the sum of the coefficients associated with the subdomains sharing the same interface gives the same coefficient that of the full matrix $\mathbf{A}_{\Gamma\Gamma}$. Equation (6) shows that any matrix-vector product involving the Schur complement matrix $\mathbf{S}$ translates into N_P independent matrix-vector products involving the local Schur complement matrices $\mathbf{S}_i$ and performed concurrently and independently on each processor, followed by a neighbour to neighbour communication of the resulting vector.

Preconditioning of the Schur complement system

A right preconditioner is applied to the Schur complement system to achieve an acceptable convergence rate (Saad 2003).

The original Schur complement system $\mathbf{S}\mathbf{p}_{\Gamma} = \mathbf{b}$ is transformed in a better suited system

$$\mathbf{S}\mathbf{M}^{-1}\mathbf{t} = \mathbf{b}, \tag{8}$$

with $\mathbf{M}\,\mathbf{p}_{\Gamma} = \mathbf{t}$. In equation (8), $\mathbf{M}$ is the preconditioner of $\mathbf{S}$.

For $\mathbf{M}^{-1}$, we used an algebraic additive Schwarz preconditioner given by

$$\mathbf{M}^{-1} = \sum_{i=1}^{N_P} \mathcal{R}_{\Gamma_i}^T \hat{\mathbf{S}}_i^{-1} \mathcal{R}_{\Gamma_i}, \tag{9}$$

where $\hat{\mathbf{S}}_i$ are the so-called assembled local Schur complement systems. The assembled local Schur complement systems $\hat{\mathbf{S}}_i$ are defined by the restriction of the Schur complement to the interface Γ_i

$$\hat{\mathbf{S}}_i = \mathcal{R}_{\Gamma_i} \mathbf{S} \mathcal{R}_{\Gamma_i}^T. \tag{10}$$

Building the assembled local Schur complements $\hat{\mathbf{S}}_i$ requires that we assemble the diagonal blocks of the local Schur complements thanks to a few neighbour to neighbour communications (Haidar 2008). Note that, if the full domain is subdivided into two subdomains only, the additive Schwarz preconditioner is exact and the iterative solver will converge in one iteration. In the opposite reasoning, the accuracy of the preconditioner is expected to deteriorate as the number of subdomains and the number of interface unknowns increases. Therefore, an increasing number of generalized minimum residual iterations is expected. The increasing number of generalized minimum residual iterations should however be balanced by the fact that each processor performs a more limited number of operations per generalized minimum residual iteration because of smaller subdomains. The antagonist effects of the increasing number of generalized minimum residual iterations and of the decreasing size of the subdomains on the computational time will be quantified extensively in the following sections. Since the preconditioner is formulated as the sum of local operators on each subdomain, the solution of the systems $\mathbf{My} = \mathbf{t}$ translates into N_P concurrent and independent solutions of local systems $\hat{\mathbf{S}}_i \mathbf{y}_i = \mathbf{t}_i$, i = 1, N_P. These local dense systems are solved on each processor using a lower-upper decomposition of the matrix $\hat{\mathbf{S}}_i$ with the subroutine CGETRF of the LAPACK linear algebra package.

We computed the preconditioner from the damped Helmholtz equation, following the original idea of Magolu monga Made (2000) further investigated by Erlangga *et al.* (2006) and Riyanti *et al.* (2006) in iterative-solver modelling. The damped wave equation reads

$$(\beta_1 + \imath\beta_2) \frac{\omega^2}{\kappa(\mathbf{x})} p(\mathbf{x}, \omega) + \nabla \left(\frac{1}{\rho(\mathbf{x})} \nabla p(\mathbf{x}, \omega) \right) = -s(\mathbf{x}, \omega), \tag{11}$$

where $\imath = \sqrt{-1}$. Suitable values of β_1 and β_2 were heuristically estimated by numerical experiments, which show that $\beta_1 = 1$ and $\beta_2 \in [0.01; 0.05]$ were providing the best convergence (Ben-Hadj-Ali, Operto and Virieux 2011). It is worth mentioning that using $\beta_1 = 1$ and $\beta_2 \neq 0$ is equivalent to introducing physical attenuation to the medium with a complex wave speed given by

$$\bar{c} = c \left(1 - \imath \frac{1}{Q} \right), \tag{12}$$

where the attenuation factor Q is given by $Q \approx 1/\beta_2$.

The impact of the complex shift of the Laplace operator on the spectrum of the matrix $\mathbf{SM}^{-1}$ for the 1D Helmholtz equation is shown in Fig. 2. When no complex shift and no intrinsic attenuation are introduced in the medium, the eigenvalues of the preconditioned matrix are aligned along the negative part of the real axis (Fig. 2a). Applying the complex shift without physical attenuation in the medium rotates the eigenvalues along the imaginary axis of the spectrum (Fig. 2b). Considering physical attenuation in the medium without complex shift contributes to clustering the eigenvalues (Fig. 2c). Finally, combining the complex shift of the Laplace operator with intrinsic attenuation in the medium contributes to improve the clustering of the eigenvalues (Fig. 2d). This can be qualitatively justified by the fact that, when $\mathbf{M}$ is computed with a complex shift, $\mathbf{M}^{-1}$ provides a better estimation of $\mathbf{S}^{-1}$ when an intrinsic attenuation exists in the medium, according to the relationship between the coefficient β_2 and the attenuation factor Q, equation (12). Note that the use of the shifted preconditioner requires performing two series of local LU factorizations: one to factorize the local impedance matrices involved in the Schur complement matrix and one to factorize the damped local impedance matrices involved in the preconditioner.

Parallel algorithm

The hybrid solver modelling algorithm can be decomposed into source-independent tasks and source-dependent tasks, which are outlined in algorithm (1). In the framework of full waveform inversion that require multi-source simulations, the computational cost of the source-dependent tasks should be kept as small as possible. The source-independent tasks consist of the lower-upper factorization of local impedance matrices on each processor and building of the local assembled Schur complements for preconditioning. The matrix-vector products involving the local Schur complement is computed without explicitly storing the local Schur complement but using its implicit form (equation (6)). It allows us to save memory and time in 3D (Haidar 2008) compared to an explicit calculation that would be based on a dense matrix-vector product. The

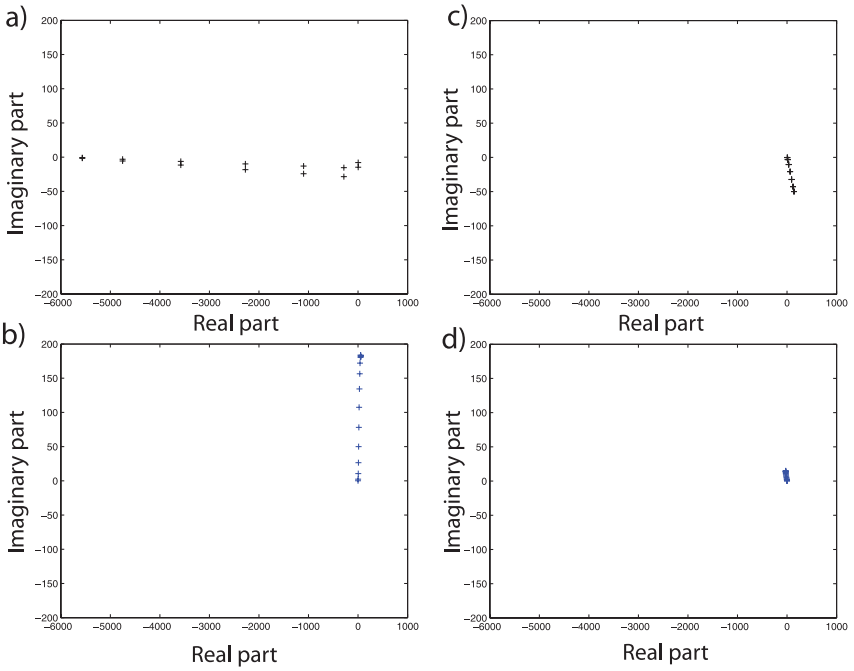

Figure 2 Spectrum of the matrix $\mathbf{SM}^{-1}$ computed for the 1D Helmholtz equation a) without intrinsic attenuation and $(\beta_1, \beta_2) = (1., 0.)$; b) without intrinsic attenuation and without complex shift $(\beta_1, \beta_2) = (1., 0.05)$; c) with intrinsic attenuation (Q = 2) and without complex shift $(\beta_1, \beta_2) = (1., 0.)$; d) with intrinsic attenuation (Q = 2) and with complex shift $(\beta_1, \beta_2) = (1., 0.05)$. The 1D medium was split into 15 subdomains.

two main source-dependent tasks consist of the iterative solution of the Schur complement system and the computation of the interior unknowns.

Stopping criterion of generalized minimum residual iterations

As a stopping criterion for the generalized minimum residual iteration, we used a backward error criterion (Wilkinson 1963) defined by

$$\epsilon = \| \mathbf{S}\mathbf{p}_{\Gamma_k} - \mathbf{b} \| / \| \mathbf{b} \| . \quad (13)$$

One advantage of this criterion is that it does not require computing $\|\mathbf{S}\|$. It is worth noting that the backward error given in equation (11) is the same for the right-preconditioned and unpreconditioned systems because $\|\mathbf{SM}^{-1}\mathbf{t} - \mathbf{b}\| = \|\mathbf{S}\mathbf{p}_\Gamma - \mathbf{b}\|$. This is a clear advantage of right preconditioning compared to left preconditioning.

The impact of the stopping criterion of the generalized minimum residual iterations on the accuracy of the solution is illustrated with an acoustic wave simulation in a 2D homogeneous medium subdivided into $2 \times 2 = 4$ subdomains of equal size. We performed the modelling for each frequency of the source bandwidth and transformed back the frequency-domain solutions in the time domain by inverse Fourier transform. Time-domain snapshots and time-domain seismograms are shown in Fig. 3, for $\epsilon = 10^{-1}$, 10^{-2} and 10^{-3}. Artificial reflections from the interfaces of the subdomains and a diffraction from the corner at the intersection of the subdomains in the middle of the medium are clearly observed when $\epsilon = 10^{-1}$ in Fig. 3(a–d). Smaller amplitude artifacts are still visible when $\epsilon = 10^{-2}$ (Fig. 3b–e). However, Ben-Hadj-Ali *et al.* (2008) concluded that the amplitude of these artefacts are sufficiently small for full waveform inversion applications. When $\epsilon = 10^{-3}$, no artefacts are visible (Fig. 3c–f).

The accuracy of the hybrid solver modelling algorithm is quantitatively validated against an analytical solution in a 3D homogeneous medium for a stopping criterion of $\epsilon = 10^{-3}$ (Fig. 4). The agreement between the hybrid solver modelling and the analytical solutions is as good as that obtained with a direct solver approach (Fig. 4c in Brossier *et al.* 2010). For an attenuation factor of $Q = 10000$, 1869 generalized minimum residual iterations are required using $8 \times 8 \times 4 = 256$ cubic subdomains of 30^3 grid points when the complex shift of the Laplacian is used in the additive Schwarz preconditioner. This number of generalized minimum residual iterations decreases down to 116, when a realistic attenuation of $Q = 200$ is used in the model. For 116 generalized minimum residual iterations, the elapsed time of the full simulation is 142 s. The elapsed times to build the preconditioner and to compute the solution are of the same order (28 s), while the elapsed time to perform sequential LU factorization on one processor is 13 s. The simulation is performed on the IBM Power 6 with 3 Gb of RAM memory per processor.

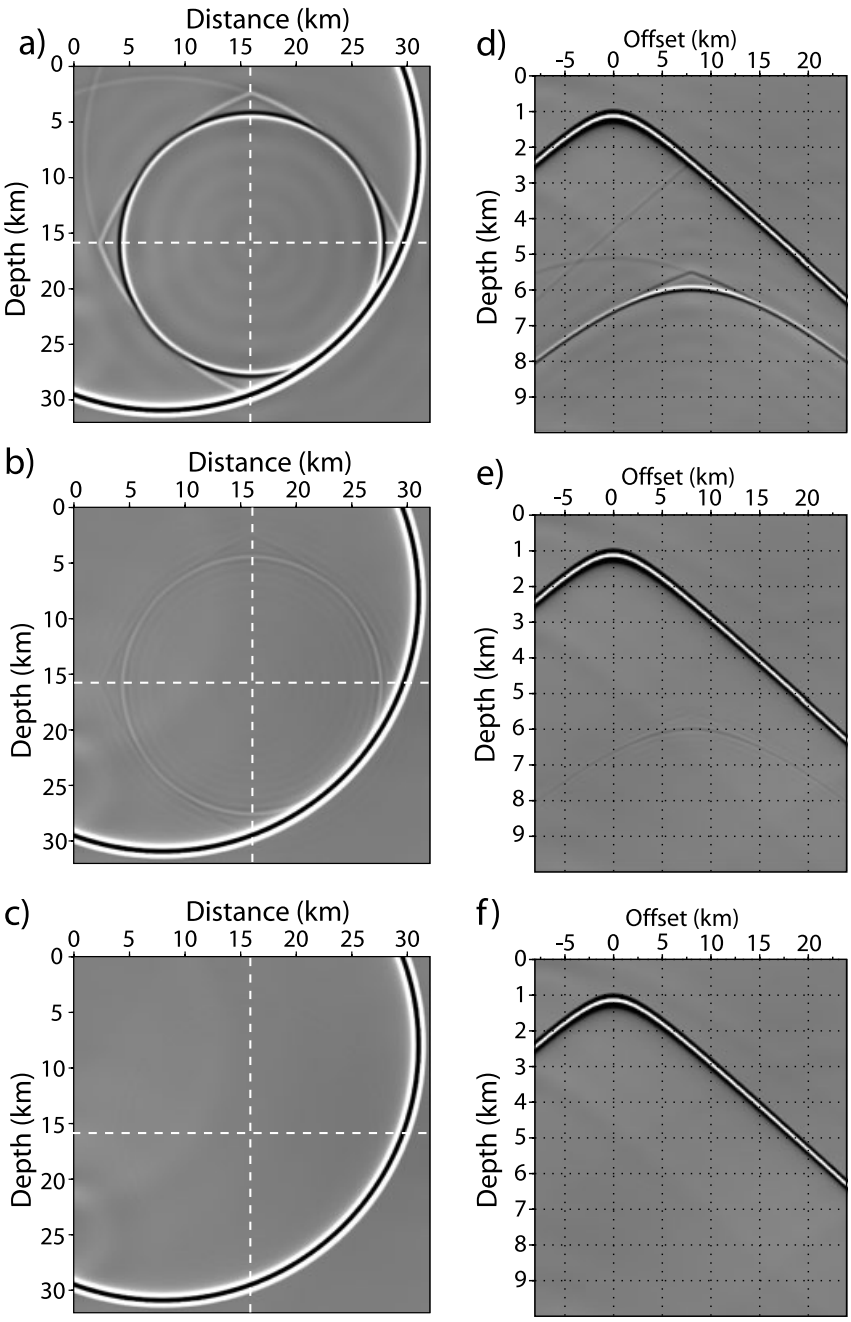

Figure 3 Modelling in 2D homogeneous medium ($V_P = 4$ km/s, $\rho = 2000$ kg/m^3). The main frequency of the Ricker wavelet is 5 Hz and the source coordinates are 8 km × 8 km. Grid interval is 40 m leading to 4 grid points per minimum wavelength. (a-c) Snapshots at traveltime 6 s. Four subdomains delineated by the dash lines are used. Stopping criterion of iterations are: a) $\epsilon = 10^{-1}$, b) $\epsilon = 10^{-2}$, c) $\epsilon = 10^{-3}$, d-f) Corresponding seismograms for a receiver line near the surface. d) $\epsilon = 10^{-1}$, e) $\epsilon = 10^{-2}$, f) $\epsilon = 10^{-3}$.

THEORETICAL COMPLEXITIES

Memory complexity

When a nested dissection algorithm is used for reordering, the theoretical memory used to perform a lower-upper factorization of a sparse matrix resulting from the discretization of the Laplacian operator with a second-order accurate finite-difference stencil is $\mathcal{O}(N^4)$, where N is the dimension of a 3D square computational grid (George and Liu 1981). In the hybrid solver modelling approach, let us denote with n_p the number of subdomains in one direction and let us consider the same number of subdomains in the three directions, ($n_p^3 = N_P$). The dimension of the cubic subdomains is N/n_p, which leads to a memory complexity for the n_p^3 local

Algorithm 1. Algorithm of the domain decomposition algorithm based on the hybrid solver modelling solver

1. On each processor i, assemble local impedance matrix $\mathbf{A}_{\mathcal{I}_i\mathcal{I}_i}$.
2. On each processor i, perform the sequential lower-upper factorization of $\mathbf{A}_{\mathcal{I}_i\mathcal{I}_i}$.
3. On each processor i, build the local Schur complement matrices $\mathbf{S}_i$ and the assembled local Schur complement matrices $\hat{\mathbf{S}}_i$.
4. On each processor i, perform the lower-upper factorization of $\hat{\mathbf{S}}_i$ (Lapack routine CGETRF)
5. **for** $l = 1$ to N_{shot} **do**
6. On each processor i, build source terms $\mathbf{s}_{\mathcal{I}_i}$ and $\mathbf{s}_{\Gamma_i}$.
7. On each processor i, build the right-hand side of the local Schur complement system $\mathbf{s}_{\Gamma_i} - \mathbf{A}_{\Gamma_i\mathcal{I}_i}\mathbf{A}^{-1}_{\mathcal{I}_i\mathcal{I}_i}\mathbf{s}_{\mathcal{I}_i}$.
8. Solve the preconditioned Schur complement system with distributed generalized minimum residual method.
9. Build the right-hand side of the system for the remaining interior-point unknowns,
10. On each processor i, compute the solution of the interior unknowns by substitutions, $\mathbf{p}_{\mathcal{I}_i} = -\mathbf{A}^{-1}_{\mathcal{I}_i\mathcal{I}_i}\mathbf{A}_{\mathcal{I}_i\Gamma_i}\mathbf{p}_{\Gamma_i} + \mathbf{A}^{-1}_{\mathcal{I}_i\mathcal{I}_i}s_{\mathcal{I}_i}$.
11. **end for**

lower-upper factorizations of $\mathcal{O}(n_p^3(N/n_p)^4) = \mathcal{O}(N^4/n_p)$. The dimension of the dense local Schur complement matrices and of the preconditioners is $(N/n_p)^2$. Therefore, the memory complexity associated with the storage of the dense n_p^3 local Schur complements and preconditioners is $\mathcal{O}(n_p^3(N/n_p)^4) = \mathcal{O}(N^4/n_p)$ and equals to that of the LU factorizations. We conclude that the memory complexity of the hybrid solver modelling approach is $\mathcal{O}(N^4/n_p)$ and that the memory request decreases with the number of subdomains for standard parallel implementation where each subdomain is assigned to one computing core. This is a clear advantage compared to the parallel direct solver methods where the memory cost increases with the number of processors because of the memory overheads associated with the parallel lower-upper factorizations.

Floating point operation complexity

The operation count complexity of a lower-upper factorization of a sparse matrix for a 3D finite-difference problem is $\mathcal{O}(N^6)$. Therefore, the complexity of the local lower-upper factorizations on each processor is $\mathcal{O}((N/n_p)^6)$ and becomes negligible as the number of subdomains increases. The complexity of the lower-upper factorization of the dense preconditioner is $\mathcal{O}((N/n_p)^4)$ and is also negligible when n_p increases. To assess the operation count complexity of the iterative component of the hybrid solver modelling method, we shall first consider that the grid interval is linearly adapted to the frequency following a discretization criterion of four grid points per minimum wavelength. Second, we shall keep constant the workload of each processor over frequencies by adapting the number of subdomains to the frequencies such that the physical dimensions of the subdomains remain the same, whatever the frequency. The two criteria lead to the relationship: $\mathcal{O}(n_p) = \mathcal{O}(N)$.

The operation count complexity of the iterative component of the hybrid solver for one shot is $\mathcal{O}(N_P N_{it}(N/n_p)^4)$ and corresponds to the computational cost of the matrix-vector product of dense matrices of dimension $(N/n_p)^2$ performed N_{it} times on each of the N_P processors. Given that we consider $\mathcal{O}(n_p) = \mathcal{O}(N)$, the operation count complexity of the generalized minimum residual solver is $\mathcal{O}(N^3 N_{it})$. This theoretical time complexity is basically the same as that of iterative solvers, which requires $\mathcal{O}(N^3)$ operations per iteration and shot. If the number of iterations increases linearly with N (or, the frequency) as observed by Plessix (2007) for iterative solver modelling, then the theoretical time complexity of the hybrid solver will be $\mathcal{O}(N^4)$. We expect, however, the number of generalized minimum residual iterations to be smaller with the hybrid approach than with a purely iterative one, because a smaller linear system more amenable to solution is solved by the hybrid approach.

THREE-DIMENSIONAL NUMERICAL SIMULATIONS

We present 3D simulations performed in the SEG/EAGE salt and overthrust models. We used two different distributed-memory computational platforms of the IDRIS computer centre (http://www.idris.fr). The first one is an IBM Blue Gene (Babel computer) composed of 1024 nodes with four 32-bit PowerPC450 cores. The clock frequency is 850 MHz, the shared memory per node is 2 Gbytes. The computational power per node is 3.4 Gflops per core. The interconnect is composed of different networks allowing for high-performance message passing interface communications. The second one is an IBM Power 6 (Vargas computer) composed of SMP p575 IH nodes. The clock frequency is 4.7 GHz. The maximum memory that can be allocated by the message passing interface process is 3 Gb and the maximum number of the

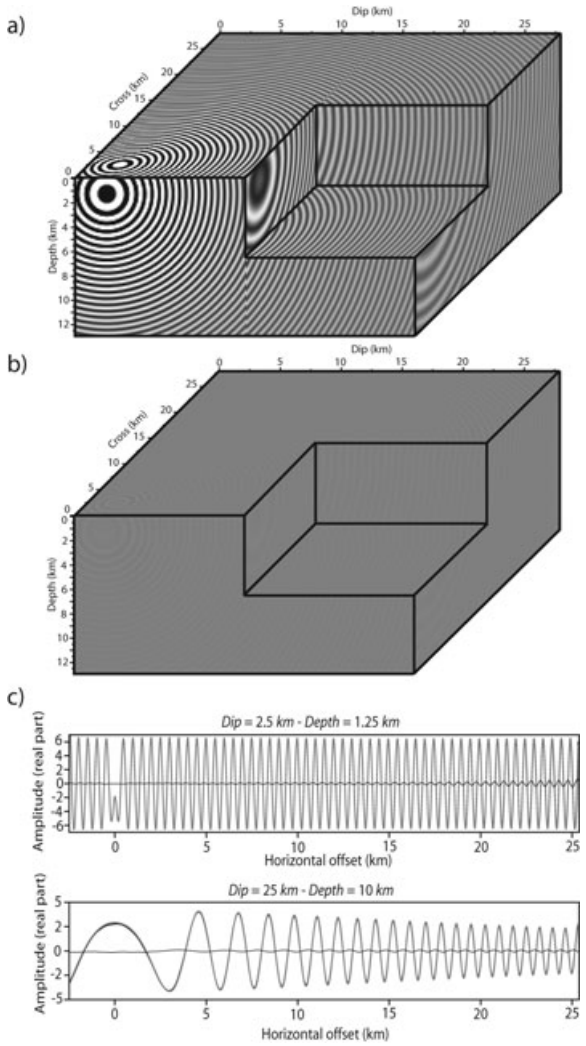

Figure 4 Modelling in 3D homogeneous medium ($V_P = 4$ km/s, $\rho = 2000$ kg/m^3). Frequency is 8 Hz. Grid interval is 125 m. Source coordinates are ($x = 2.5$ km, $y = 2.5$ km, $z = 1.25$ km). a) 3D monochromatic wavefield computed with the hybrid solver modelling (HSM) method. b) Difference with the analytical solution plotted with the same amplitude scale as in a). c) Comparison between profiles extracted from the HSM (black line) and the analytical (grey line) wavefields. The difference is the thin black line. The top profile in the cross direction runs across the source position. Amplitudes were corrected for geometrical spreading. The bottom profile is at a distance of 25 km in the dip direction and at a depth of 10 km.

message passing interface process is 512. The interconnect of the Vargas computer is Infiniband x4 DDR.

We validate the monochromatic wavefield solutions computed with the hybrid solver modelling method against the solution computed with an acoustic finite-difference time-domain method, referred to as the time-domain approach in the following. The finite-difference time-domain code is based on a classic velocity-stress formulation of the acoustic wave equation discretized with $\mathcal{O}(\Delta t^2, \Delta x^4)$ staggered-grid stencils (Levander 1988). Absorbing boundary conditions are convolutional perfectly matched layers (Komatitsch and Martin 2007). The memory variables associated with the convolutional perfectly matched layers are implemented in the full computational physical domain to mimic intrinsic attenuation. Therefore, the computational time of the time-domain simulation provided hereafter is representative of visco-acoustic modelling. The perfectly matched layers are set along the six faces of the model for both hybrid solver modelling and time-domain approaches. Therefore, no free surface is considered. The finite-difference time-domain code is parallelized by domain decomposition using Cartesian topology suitable for finite-difference grids. Monochromatic wavefields are extracted from the time-domain solutions using a discrete Fourier transform computed within the time loop (Sirgue *et al.* 2008). In the following, we shall use the same grid interval for HSM and time-domain simulations.

We assess the parallelism efficiency by the quantity E given by:

$$E = \frac{T_{ref} N_P^{ref}}{T_{N_P} N_P}, \tag{14}$$

where T_{ref} and T_{N_P} are the elapsed times obtained on the smallest number of processors, N_P^{ref} and on N_P processors, respectively. This efficiency should remain close to 1 as N_P increases.

3D EAGE/SEG salt model

We perform simulation in the SEG/EAGE salt model of dimensions $13.5 \times 13.5 \times 4.18$ km^3 (Fig. 5a) (Aminzadeh, Brac and Kunz 1997). The minimum and maximum velocities are 1.5 and 4.482 km/s respectively.

Validation against finite-difference time-domain solutions

The salt model was resampled with a grid interval of 30 m to perform simulation at a frequency of 12.5 Hz corresponding to four grid points per minimum propagated wavelengths (Table 1 and Fig. 5). The number of grid points in the perfectly matched layers are $6 \times 6 \times 5$ for the hybrid solver modelling and $20 \times 20 \times 20$ for the time-domain simulation (a larger number of grid points in the perfectly matched layer is used in the time-domain simulation because all the frequencies, including the small frequencies, are simultaneously modelled in

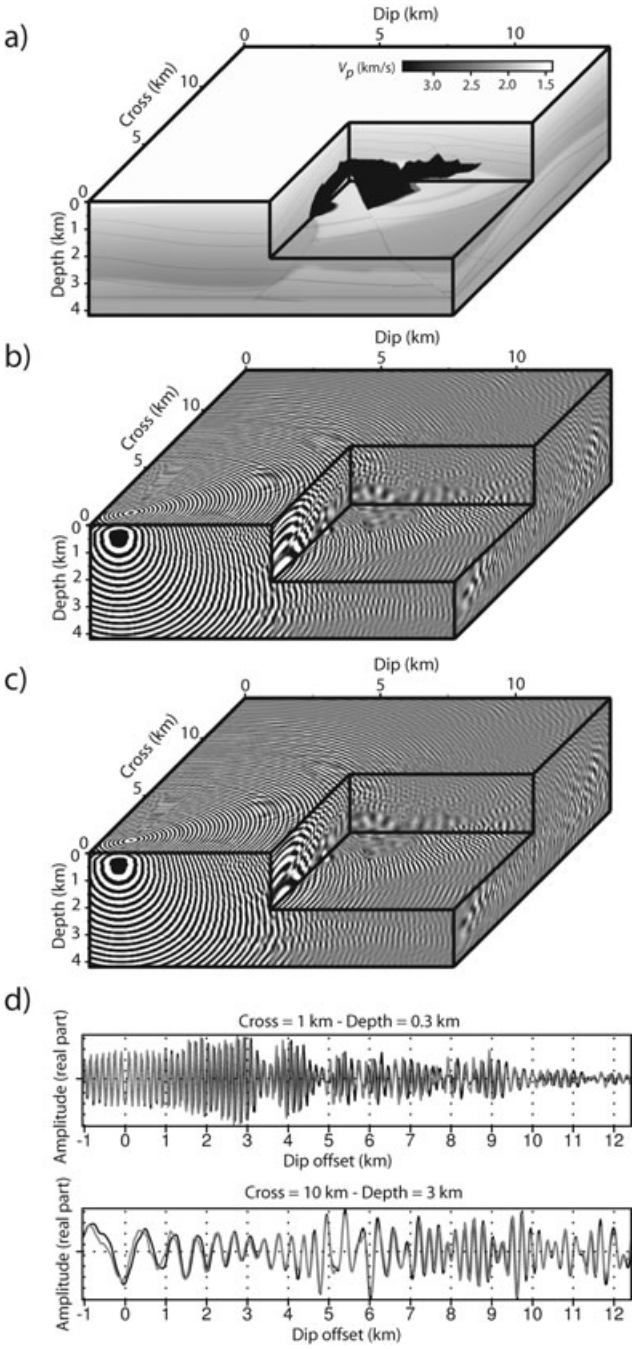

Figure 5 Modelling in the acoustic salt model. a) Salt model (h = 30 m). b-c) Monochromatic wavefields computed with the hybrid solver modelling (b) and the time-domain (c) methods. Frequency is 12.5 Hz. Source coordinates are ($x = 1.05$ km, $y = 1.05$ km, $z = 0.3$ km). d) Comparison between profiles of wavefields computed with the hybrid solver modelling (black line) and the time-domain method (grey line). The top profile in the dip direction runs across the source position. Amplitudes were corrected for geometrical spreading. The bottom profile is at a distance of 10 km in the cross direction and at a depth of 3 km. See also Table 1.

the time domain, unlike in the frequency domain where the width of the perfectly matched layers can be matched to the mean propagated wavelength). The number of unknowns for the hybrid solver modelling simulation is 32 million (Table 1). The hybrid solver modelling and the time-domain simulations are performed on the IBM Blue Gene at the IDRIS computer centre. The hybrid solver modelling simulation is performed on $14 \times 14 \times 5 = 980$ processors using subdomains of $33 \times 33 \times 30$ grid points (Table 1). Two gigabytes of RAM memory were allocated per message passing interface process. The time-domain simulation is performed on 64 processors for a simulation length of 10 s corresponding to 3300 time steps and a time interval of 3 ms.

A first simulation is performed for an attenuation $Q = 10000$ to compare the hybrid solver modelling solution with the acoustic solution of the time-domain approach. The hybrid solver modelling simulation is done with the weights of the mixed grid stencil corresponding to $G_m = 4, 6, 8, 10$ (Brossier *et al.* 2010). A qualitative comparison between the monochromatic wavefields computed with the hybrid solver modelling and the time-domain approaches shows a reasonable agreement given the coarse parameterization used for both approaches (4 grid points per minimum wavelength) and the different numerical stencils used for the two modellings (Fig. 5). When $Q = 10000$, the number of generalized minimum residual iterations for $\epsilon = 10^{-3}$ and $\epsilon = 10^{-2}$ are 1375 and 954, respectively. When an heterogeneous attenuation model is used with $Q = 50$ in the salt body (Fig. 6a), the number of iterations decreases to 111 and 57, respectively. The footprint of the attenuation on the wavefield amplitudes is shown in Fig. 6(b-c). For the heterogeneous attenuation model, the elapsed time to build the preconditioner is 203 s, the elapsed time to perform the generalized minimum residual iterations is 75 s (for $\epsilon = 10^{-3}$), the elapsed time to compute the solution (i.e., the full source-dependent task) is 109 s and the elapsed time for one sequential lower-upper factorization is 103 s. The total elapsed time is 797 s. The total allocated memory is roughly 2 Tb. The elapsed time for the time-domain simulation was 2115 s on 64 processors (Table 1).

Scalability

To assess the scalability of the hybrid solver modelling method, we perform a series of simulations in the salt model for an increasing number of processors. The scalability analysis is performed on the IBM Blue Gene and the IBM Power 6 (Table 2). On the IBM Blue Gene, the modelled frequency

Table 1 Simulation in the salt and overthrust models. *FDTD*: finite-difference time-domain modelling. $F(Hz)$: modelled frequency in Hz. $h(m)$: grid interval. $n_{pml_{x,y,z}}$: number of grid points in the perfectly matched layers along x, y and z. $N_u(10^6)$: number of unknowns in millions. N_P^{HSM}: number of processors (or subdomains) for the hybrid solver modelling approach. $k_x \times k_y \times k_z$: number of subdomains along the three Cartesian directions for the hybrid solver modelling approach. $n_x \times n_y \times n_z$: dimensions of the subdomains in *HSM*. $T_t^{HSM}(s)$: total elapsed time in seconds for the hybrid solver modelling approach. $T_r^{HSM}(s)$: elapsed time in seconds of the source-dependent task in the hybrid solver modelling approach. N_P^{TDM}: number of processors for the time-domain approach. $T_r^{FDTD}(s)$: elapsed time in seconds for the finite-difference time-domain approach. The stopping criterion of iterations was $\epsilon = 10^{-3}$

F	$n_{pml_{x,y,z}}$	N_u	N_P^{HSM}	$k_x \times k_y \times k_z$	$n_x \times n_y \times n_z$	T_t^{HSM}	T_r^{HSM}	N_P^{FDTD}	T_r^{FDTD}
Salt model - Grid interval h: 30 m - Grid dimensions: $450 \times 450 \times 140$									
12.5	$6 \times 6 \times 5$	32	980	$14 \times 14 \times 5$	$33 \times 33 \times 30$	797	109	64	2115
Overthrust model - Grid interval $h = 50$ m - Grid dimensions: $400 \times 400 \times 94$									
10.8	$8 \times 8 \times 9$	19.4	1024	$16 \times 16 \times 4$	$26 \times 26 \times 28$	425	175	64	1700

is 7.5 Hz and the grid interval h is 50 m, which leads to a Cartesian grid of dimensions $270 \times 270 \times 84$. On the IBM Power 6, the modelled frequency is 5 Hz and the grid interval is 75 m, which leads to a finite-difference grid of dimensions $180 \times 180 \times 56$. The domain decompositions are designed such that the subdomains geometry is as close as possible to that of a square to minimize the number of interface points. The number of subdomains increases from 300 to 1944 on the IBM Blue Gene and from 50 to 432 on the IBM Power 6 (Table 2). The total elapsed time and the elapsed time of the source-dependent tasks, the number of generalized minimum residual iterations and the efficiency of the full computation and of the source-dependent tasks are plotted in Fig. 7. A first conclusion is that the number of GMRES iterations roughly increases linearly with the number of subdomains (Fig.7c,d), which illustrates the degradation of the preconditioner accuracy with the number of subdomains. Second, the elapsed time of the source-dependent tasks remains almost constant when the number of subdomain increases, in particular, on the IBM Power 6 where faster processors are used (Fig. 7c,d). This suggests that, when the number of subdomains increases, the computational burden resulting from the increasing number of generalized minimum residual iterations is balanced by the computational saving on each processor provided by the decreasing of the size of the subdomains. Third, the computational cost of the source-independent tasks, measured by the difference between the total time and the source-dependent time, rapidly decreases with the number of subdomains (Fig. 7a,b), which is consistent with the theoretical time complexity of the lower-upper factorization and of the preconditioner building. The efficiency of the full run always remains greater than 1 and increases with the number of processors when fast processors are used (Fig. 7e,f). By contrast, the efficiency of the source-dependent tasks decreases with the number of processors since the elapsed time of the source-dependent phase remains almost constant regardless of the number of processors used. The behavior of the hybrid solver modelling algorithm is slightly better on the IBM Power 6 than on the IBM Blue Gene, as shown by the efficiency curves (Fig. 7), because of the faster processors of the Power 6 and the limited amount of communications in the hybrid solver modelling method. When simulations must be performed for a large number of sources, the best strategy is clearly to use a number of subdomains as small as possible to maintain the efficiency of the source-dependent phase close to 1. Groups of sources can be distributed over groups of N_P processors where N_P denotes the number of processors required to perform one domain decomposition such that each group of N_P processors processes a limited number of shots (if N_P^{tot} is the total number of processors, the number of source per groups of N_P processors will be $N_{shot}N_P/N_P^{tot}$). By contrast, if a simulation must be performed for one source as it can be the case when source encoding is used (Krebs *et al.* 2009; Ben-Hadj-Ali *et al.* 2011), the best strategy is to use a significant number of processors to make negligible the computational cost of the source-independent tasks.

Complexity

We assess numerically the time complexity of the hybrid solver modelling method on the IBM Blue Gene computer. Simulations are performed for frequencies 5, 7.5, 10 and 12.5 Hz.

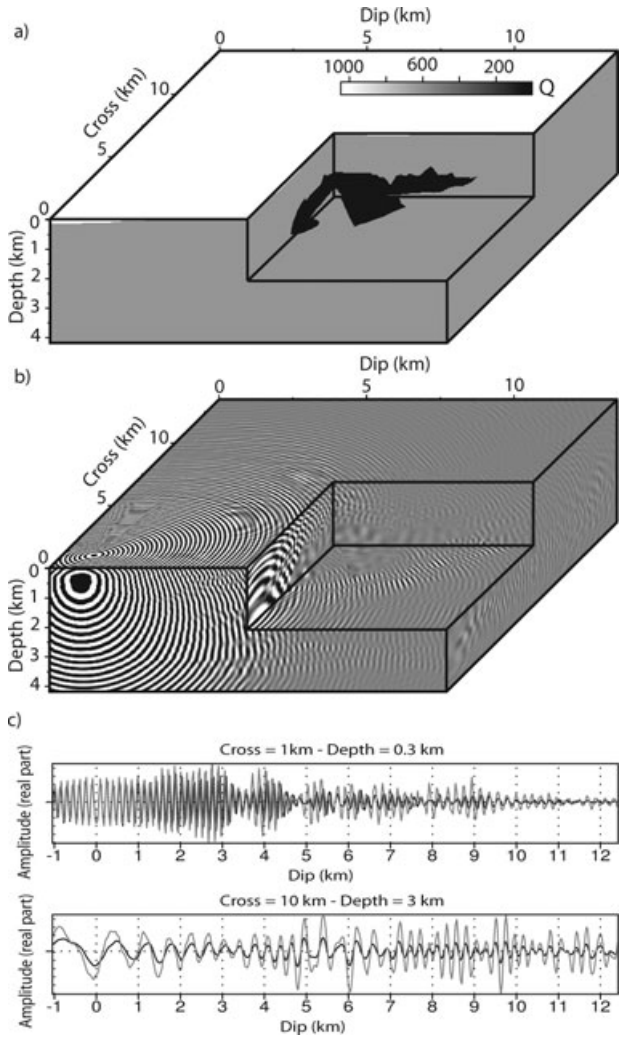

Figure 6 Modelling in the visco-acoustic salt model. a) Attenuation model. b) Monochromatic wavefield computed with the hybrid solver modelling approach. Frequency is 12.5 Hz. Wavefield amplitudes can be compared with that shown in Fig. 5(b). c) Comparison between profiles extracted from the hybrid solver modelling wavefields computed in the attenuating (black line) and in the non attenuating (grey line) models. The profiles are at the same positions as in Fig. 5.

The grid interval is adapted to each frequency to satisfy a discretization criterion of four grid points per minimum wavelength (h = 75 m, 50 m, 37.5 m and 30 m for the above-mentioned frequencies). The corresponding grid dimensions are provided in Table 3. The problem sizes increases from 2.4–32 millions of unknowns. The number of processors for each frequency modelling was chosen such that the size of the subdomains (i.e., the processor workload) is kept roughly constant, whatever the frequency. The size of the subdomain is, an average, 31 × 31 × 31, which is close to the maximum size that can be tackled with 2 Gb of memory per message passing interface process. The number of processors increases with frequencies, from 72 to 980. All the simulation parameters are outlined in Table 4. The total elapsed time, the elapsed time for the source-dependent tasks and the number of generalized minimum residual iterations are shown in Fig. 8 and are outlined in Table 4. The number of generalized minimum residual iterations increases linearly with frequency, which is consistent with the linear increase of iterations with the number of subdomains shown during the scalability analysis. The total elapsed time and the source-dependent elapsed time also increase roughly linearly with frequency. To estimate the observed time complexity, we plot the quantity $N_P T/N^c$ as a function of the frequency in Fig. 8(c). N_P is the number of processors, T is the elapsed time, N is the number of grid points in the model along the horizontal direction and c is an exponent such that the plotted curve remains centred around the value of 1. The expression N^c gives the observed complexity of the hybrid solver modelling algorithm. We found an observed complexity of $\mathcal{O}(N^{3.4})$ and $\mathcal{O}(N^{3.8})$ for the total run and the source-dependent tasks, respectively. The observed complexity of the full algorithm $\mathcal{O}(N^{3.4})$ is slightly smaller than the theoretical one $\mathcal{O}(N^4)$, assuming a linear increase of the number of iterations with N (see section Theoretical complexity analysis).

3D EAGE/SEG overthrust model

We followed the same procedure for the SEG/EAGE onshore overthrust model. The dimension of the overthrust model is 20 km × 20 km × 4.65 km (Fig. 9a). The minimum and maximum velocities are 2.178 km/s and 6 km/s, respectively. The overthrust model represents a thrusted area above a decollement level with a weathered layer in the near-surface and, therefore, provides a more structurally-complex geological environment than the salt model. Then, it provides a complementary case study to assess the robustness of the hybrid solver modelling approach for modelling in a heterogeneous medium.

Validation against finite-difference time-domain solutions

A comparison with the time-domain approach is performed for the 10.8 Hz frequency. The model is resampled with a grid interval of 50 m to satisfy the discretization criterion of four grid points per wavelength used for both the hybrid solver modelling and the time-domain simulations. All the

Table 2 Scalability analysis of the hybrid solver modelling performed on the IBM Blue Gene and the IBM Power 6. The same nomenclature as in Table 1 is used. E_{tot}: parallel efficiency for the full hybrid solver modelling. E_{rhs}: parallel efficiency for the source-depedent tasks. N_{it}: number of generalized minimum residual iterations

N_P	$k_x \times k_y \times k_z$	$n_x \times n_y \times n_z$	$npml_{x,y,z}$	N_u	$T_t(s)$	$T_r(s)$	N_{it}	E_{tot}	E_{rhs}
IBM Blue Gene - Salt model - Frequency: 7.5 Hz - h = 50 m - Grid dimensions: 270 × 270 × 84									
300	10 × 10 × 3	29 × 29 × 32	10 × 10 × 6	8.1	520	77	83	1	1
484	11 × 11 × 4	26 × 26 × 26	8 × 8 × 10	8.5	255	48	114	1.26	1.00
980	14 × 14 × 5	21 × 21 × 20	12 × 12 × 8	8.6	97	30	202	1.64	0.79
1620	18 × 18 × 5	16 × 16 × 20	9 × 9 × 8	8.3	88	58	472	1.09	0.25
1944	18 × 18 × 6	16 × 16 × 16	9 × 9 × 6	8.0	56	30	612	1.42	0.39
IBM Blue Gene - Overthrust model - Frequency: 7.5 Hz - h = 75 m - Grid dimensions: 266 × 266 × 62									
243	9 × 9 × 3	32 × 32 × 28	11 × 11 × 11	7.0	599	106	119	1	1
576	12 × 12 × 4	23 × 23 × 21	5 × 5 × 11	6.4	132	43	229	1.92	1.00
980	14 × 14 × 5	20 × 20 × 16	7 × 7 × 9	6.3	96	59	534	1.54	0.45
1536	16 × 16 × 6	18 × 18 × 12	11 × 11 × 5	6.0	110	89	842	0.86	0.19
1944	18 × 18 × 6	16 × 16 × 12	11 × 11 × 5	6.0	101	85	1124	0.74	0.16
IBM Power 6 - Salt model - Frequency: 5 Hz - h = 75 m - Grid dimensions: 180 × 180 × 56									
50	5 × 5 × 2	38 × 38 × 33	5 × 5 × 5	2.4	224	23	30	1	1
72	6 × 6 × 2	32 × 32 × 33	6 × 6 × 5	2.4	129	14	36	1.21	1.18
192	8 × 8 × 3	24 × 24 × 22	6 × 6 × 5	2.4	38	7	68	1.53	0.88
400	10 × 10 × 4	19 × 19 × 17	5 × 5 × 6	2.5	13	4.6	102	2.15	0.64
432	12 × 12 × 3	16 × 16 × 22	6 × 6 × 5	2.4	12	5	129	2.18	0.54
IBM Power 6 - Overthrust model - Frequency: 5.4 Hz - h = 100 m - Grid dimensions: 200 × 200 × 46									
72	6 × 6 × 2	35 × 35 × 28	5 × 5 × 5	2.5	125	15.1	49	1	1
98	7 × 7 × 2	30 × 30 × 28	5 × 5 × 5	2.5	90	11.2	60	1.	0.99
243	9 × 9 × 3	24 × 24 × 20	8 × 8 × 7	2.8	33	9.8	135	1.13	0.46
363	11 × 11 × 3	20 × 20 × 20	10 × 10 × 7	2.9	24	10.5	220	1.05	0.29
432	12 × 12 × 3	18 × 18 × 20	8 × 8 × 7	2.8	19	10.1	285	1.10	0.25

simulation parameters are outlined in Table 1 using the same nomenclature as for the salt model. For comparison with the acoustic finite-difference time-domain solution, an attenuation factor of $Q = 10000$ is first used. In the time-domain simulation, the simulation length is 14.8 s leading to 3700 time steps for a time interval of 4 ms. The monochromatic wavefields computed with the hybrid solver modelling and the time-domain approaches are compared in Fig. 9(b-d) and show an acceptable agreement. When $Q = 10000$, the number of generalized minimum residual method iterations is 1767 and 1158 for $\epsilon = 10^{-3}$ and $\epsilon = 10^{-2}$, respectively. For a homogeneous attenuation model with $Q = 200$, the number of iterations decreases to 371 and 163 for $\epsilon = 10^{-3}$ and $\epsilon = 10^{-2}$, respectively (Fig. 10). For the hybrid solver modelling simulation, the total elapsed time and the source-dependent elapsed time are 425 s and 175 s, respectively. The elapsed time on 64 processors for the time-domain simulation is 1700 s (Table 1).

Scalability

The scalability analysis is performed on the IBM Blue Gene and the IBM Power 6 for a frequency of 7.5 Hz ($h = 75$ m) and 5.4 Hz ($h = 100$ m), respectively (Table 2). All the simulation parameters and the results are outlined in Table 3 and Fig. 11 using the same nomenclature as for the salt model. As for the salt model, the number of generalized minimum residual method iterations increases linearly with the number of subdomains and the resulting computational burden is balanced by the decrease of the subdomain dimensions. The higher structural complexity of the overthrust model compared to the salt model is, however, illustrated by the fact that the elapsed times increase more significantly after a critical number of processors in the overthrust model compared to the salt-model case. This is more obvious on the IBM Blue Gene where the processors have a slower clock frequency.

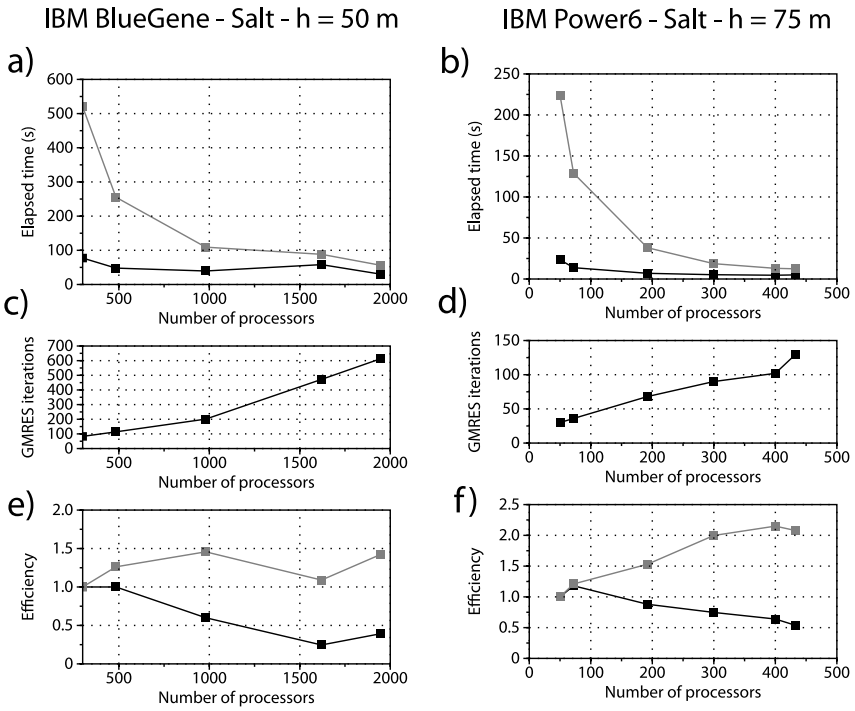

Figure 7 Hybrid solver modelling scalability (salt model). a-b) Total elapsed time (grey line) and elapsed time for the source-dependent tasks (black line) versus the number of processors. The square gives the number of processors for which simulations were performed. c-d) Number of generalized minimum residual iterations versus the number of processors. e-f) Efficiency of the full algorithm (grey line) and of the source-dependent tasks (black line). On the left panels (a, c, e), the simulations were performed on the IBM Blue Gene, while on the right panels (b, d, f), the simulations were performed on IBM Power 6. See also Table 2.

Table 3 Complexity analysis of the hybrid solver modelling (HSM) approach. The same nomenclature as for Table 1 is used. $N_x \times N_y \times N_z$: dimensions of the finite-difference grid

F	$N_x \times N_y \times N_z$	N_u	N_P	$k_x \times k_y \times k_z$	$n_x \times n_y \times n_z$	$n_{pml_{x,y,z}}$	T_t	T_r	N_{it}
Salt model (From top to bottom, $h = 75$, 50, 37.5, 30 m).									
5	180 × 180 × 56	2.4	72	6 × 6 × 2	32 × 32 × 33	6 × 6 × 5	588	61	40
7.5	269 × 269 × 84	8.3	243	9 × 9 × 3	31 × 31 × 32	5 × 5 × 6	672	91	74
10	360 × 360 × 112	17.2	576	12 × 12 × 4	31 × 31 × 31	6 × 6 × 6	689	114	96
12.5	450 × 450 × 140	32.0	980	14 × 14 × 5	33 × 33 × 30	6 × 6 × 5	797	109	111
Overthrust model (From top to bottom, $h = 150$, 100, 75, 60, 50 m).									
3.6	134 × 134 × 32	1.1	50	5 × 5 × 2	30 × 30 × 24	8 × 8 × 8	222	24	32
5.4	200 × 200 × 46	3.1	128	8 × 8 × 2	28 × 28 × 31	12 × 12 × 8	313	35	66
7.5	266 × 266 × 62	7.0	243	9 × 9 × 3	32 × 32 × 28	11 × 11 × 11	599	106	119
9	334 × 334 × 78	12.4	432	12 × 12 × 3	30 × 30 × 32	13 × 13 × 9	699	174	234
10	400 × 400 × 94	19.4	1024	16 × 16 × 4	26 × 26 × 28	8 × 8 × 9	425	175	487

Complexity

Simulations in the overthrust model at frequencies 3.6, 5.4, 7.2, 9 and 10.8 Hz are performed on the IBM Blue Gene to perform the complexity analysis. All the simulation parameters and the results are outlined in Table 3 and Fig. 12 using the same nomenclature as for the salt model. We find a time complexity of $\mathcal{O}(N^{3.5})$ and $\mathcal{O}(N^{4.2})$ for the full run and for the source-dependent tasks. The time complexities of the full algorithm are slightly higher than the theoretical one, unlike those inferred from the salt model. Indeed, the time complexity of the source-dependent tasks is significantly degraded compared to that obtained with the salt model, which highlights the sensitivity of the iterative component of the hybrid solver to the structural complexity of the subsurface model.

DISCUSSION

We have provided an analysis of the accuracy, scalability and complexity of the hybrid solver modelling approach, which

Table 4 Comparison between iterative and hybrid solvers. The results of R. E Plessix and X. Pinel are taken from Plessix (2007) and Pinel (2010), respectively. $N_x \times N_y \times N_z$: Dimensions of the finite-difference grid. N_P: Number of processors used for the simulation. $T(mn)$: Parallel elapsed time. $T_{seq}(mn) = T \times N_P$

Author	F(Hz)	$N_x \times N_y \times N_z$	N_P	T (mn)	$T_{seq}(mn)$
Salt model					
Plessix	5	295 × 295 × 140	1	154	154
Pinel	5	571 × 571 × 199	256	2	507
Sourbier	5	190 × 190 × 66	50	0.4	19
Plessix	10	520 × 520 × 210	1	1111	1111
Pinel	10	1112 × 1112 × 367	2048	4.5	9216
Sourbier	12.5	460 × 460 × 150	980	1.8	1780
Overthrust model					
Plessix	5	306 × 306 × 126	1	126	126
Sourbier	5	210 × 210 × 56	125	0.25	19
Plessix	10	541 × 541 × 181	1	1380	1380
Sourbier	10	418 × 418 × 110	1024	2.9	2986
Pinel	7.5	863 × 863 × 231	256	7	1800
Sourbier	7.5	276 × 276 × 72	106	4	429

is developed as a modelling engine for 3D frequency-domain full waveform inversion. Full waveform inversion generally requires performing modelling for a large number of sources. Alternatively, full waveform inversion can be performed for a limited number of sources or even for one single source if source encoding techniques are used (Krebs *et al.* 2009; Ben-Hadj-Ali *et al.* 2011). The advantages and drawbacks of the hybrid solver modelling method compared to alternative approaches are discussed in the following in light of the numerical experiments presented in this study. A complementary review of modelling approaches for full waveform inversion is provided in Plessix (2007) and Virieux *et al.* (2009).

Direct-solver versus hybrid-solver

Overcoming limitations of frequency-domain modelling based on the direct solver resulting from the memory cost and the limited scalability of lower-upper factorization of large matrices (Operto *et al.* 2007; Brossier *et al.* 2010) has been the guide line of this investigation on the hybrid solver modelling method. Recent applications of the direct solver approach are presented in Brossier *et al.* (2010), where modelling in the overthrust and salt models is performed on a SGI ALTIX ICE supercomputer at the 7-Hz frequency using 32–48 message passing interface processes with 15 Gb of memory per process. On the SGI platform, the computation of one monochromatic wavefield by forward/backward substitutions takes between 0.97 s and 1.4 s, while the lower-upper factorization of the impedance matrix took between 1822–2863 s. These results can be qualitatively compared with those obtained during the scalability analysis performed on the IBM Blue Gene at 7.5 Hz (Table 2). For the salt model, the elapsed time to compute one monochromatic wavefield varies between 77 s and 30 s, depending on the number of processors used, while the elapsed time to compute one monochromatic wavefield varies between 106 s and 85 s for the overthrust model. Although the total time of the simulation for one shot is less with the hybrid solver modelling, the efficiency of the substitution step in the direct-solver approach will make this approach definitively superior for problems involving a few million of unknowns and a large number of sources (a few thousands). This problem size is representative of the inversion of low frequencies (<7 Hz) for imaging targets spanned by wide-azimuth surveys performed with node-instruments and ocean-bottom-cables technologies (Plessix 2009; Sirgue *et al.* 2010).

We have shown that the memory requirement in the hybrid solver modelling approach decreases with the number of processors, unlike parallel direct solver approaches where the parallel lower-upper factorization introduces some memory overheads. We have also illustrated the good scalability of the hybrid solver modelling approach, which allows us to perform the hybrid solver modelling on a large number of processors with a limited amount of shared memory. These two factors allow us to tackle larger problems than with the direct-solver

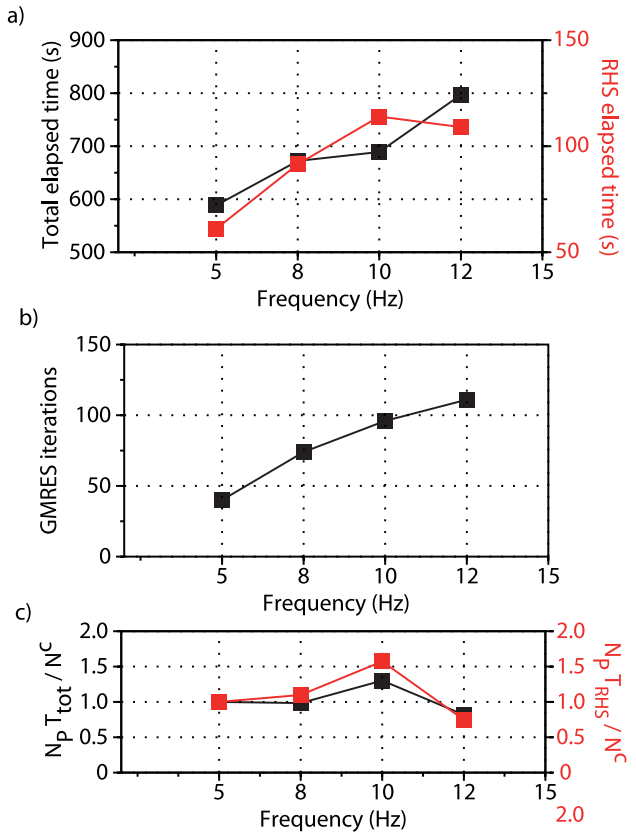

Figure 8 Hybrid solver modelling complexity (salt model). a) Total elapsed time (black line) and elapsed time for the source-dependent task versus frequency (see text for details). b) Number of generalized minimum residual iterations versus frequency. c) Estimation of the time complexity summed over all the processors (i.e., as if the code was run in sequential) for the full algorithm (black line) and the source-dependent task (red line) when $\mathcal{O}(k) = \mathcal{O}(N)$. $c = 3.4$ and $c = 3.8$ were found for the full algorithm and the source-dependent task, respectively. See also Table 2.

approach. It is worth mentioning also that the direct solver approach is more adapted to fixed-spread acquisition such as node or ocean bottom cable acquisitions in marine environment or land acquisitions, because all the sources of the experiment can be processed from one lower-upper decomposition performed for the full computational domain. In contrast, the hybrid solver modelling can deal in a more flexible way with other acquisitions such as streamer acquisition. Indeed, the hybrid solver modelling domain decomposition can be applied to subdomains of the subsurface spanned by a few number of shots. Since the number of shots in the subsurface subdomain can be small, the number of subdomains in the hybrid solver modelling domain decomposition can be optimally designed to minimize the cost of the source-independent tasks.

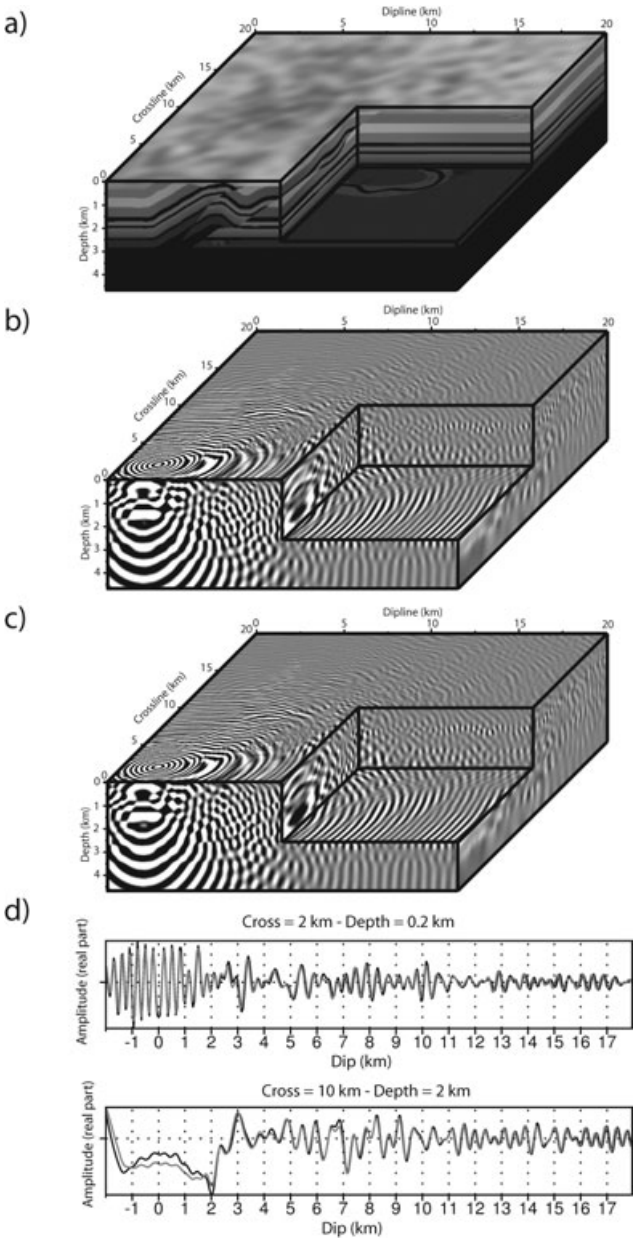

Figure 9 Modelling in the acoustic overthrust model. a) Overthrust model (h = 50 m). (b-c) Monochromatic wavefields computed with the hybrid solver modelling, (b) and the time-domain (c) methods. Frequency is 8 Hz. Source coordinates are ($x = 2.0$ km, $y = 2.0$ km, $z = 0.2$ km). d) Comparison between profiles of wavefields computed with the HSM (black line) and the time-domain method (grey line). The top profile in the dip direction runs across the source position. Amplitudes were corrected for geometrical spreading. The bottom profile is at a distance of 10 km in the cross direction and at a depth of 2 km. See also Table 1.

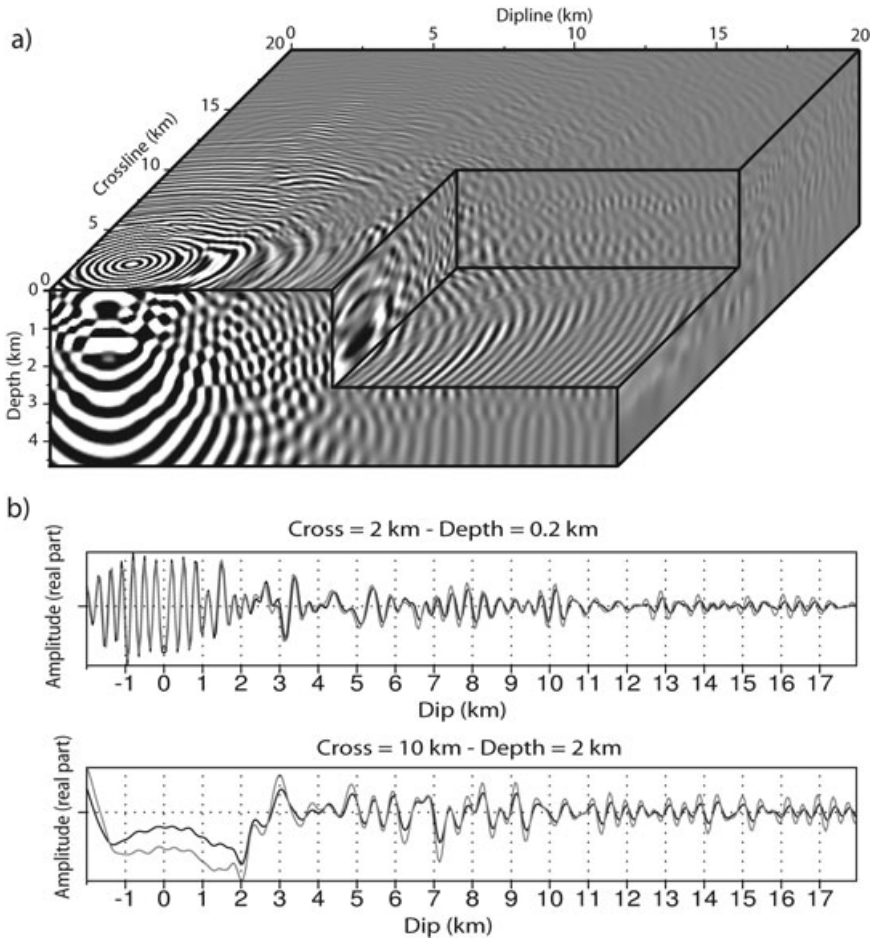

Figure 10 Modelling in the visco-acoustic overthrust model. a) Monochromatic wavefield computed with the hybrid solver modelling approach. Frequency is 10.8 Hz. The attenuation model is homogeneous with $Q = 200$. The visco-acoustic wavefield amplitudes can be compared with that of the acoustic wavefield shown in Fig. 9(b). b) Comparison between profiles extracted from the HSM wavefields computed in the attenuating (black line) and in the non attenuating (grey line) models. The profiles are at the same positions as in Fig. 9.

Time-domain explicit solver versus hybrid-solver

The hybrid solver modelling approach is validated against finite-difference time-domain modelling in Figs 5–9. For the 12.5-Hz modelling in the salt model, the elapsed time of the source-dependent phase in the hybrid solver modelling is 109 s on 980 processors (Table 1). The elapsed time of the time-domain simulation for one source is 2115 s on 64 processors. Taking the product of the elapsed time by the number of used processors allows us to assess which of the two approaches is the most efficient one for multi-source modelling. We obtained 29.67 hours for the hybrid solver modelling against 37.6 hours for the time-domain simulation. For the 10.8-Hz modelling in the overthrust model, the elapsed time of the source-dependent phase is 175 s on 1024 processors. The elapsed time of the time-domain simulation for one source is 1700 s on 64 processors. The product of the elapsed time by the number of processors gives 49.78 hours and 30.22 hours for the hybrid solver modelling and the time-domain approaches, respectively. These results suggest that the performance of the hybrid solver modelling and time-domain approaches are of the same order and that the iterative solver used in the hybrid solver modelling can be sensitive to the structural complexity of the subsurface model as already mentioned. Of note, the time-domain approach and the hybrid solver modelling are compared here for the modelling of one frequency. One advantage of the time-domain approach is to allow for the extraction of several frequencies by discrete Fourier transform at a reasonable computational cost. Although the hybrid solver modelling allows for several levels of parallelism (Haidar 2008; Giraud *et al.* 2010), two levels

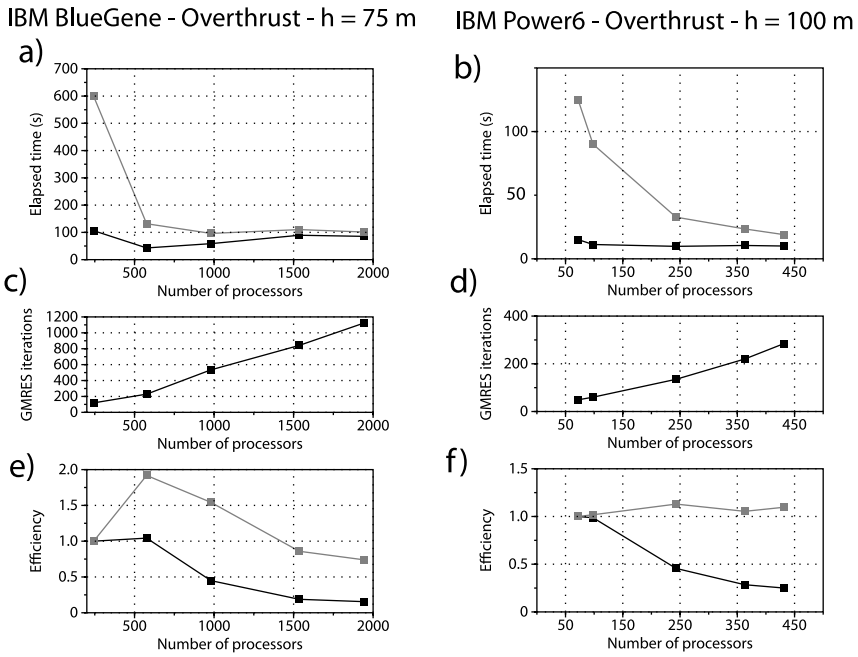

Figure 11 Hybrid solver modelling scalability (overthrust model). The same representation is used as in Fig. 7. See also Table 3.

of parallelism by domain decomposition and by distribution of shots over processors can be easily implemented in the time domain and tuned according to the number of shots to be modelled. Another issue concerns data preconditioning by time windowing, which can be performed in a more flexible way in time-domain modelling. The main drawback of the time-domains approach compared to frequency-domain approaches remains the implementation of attenuation effects by means of memory variables.

Iterative solver versus hybrid solver

Three-dimensional visco-acoustic frequency-domain wave modelling based on iterative solvers has been developed for full waveform inversion applications (Plessix 2007; Calandra *et al.* 2008; Plessix 2009; Pinel 2010). Plessix (2007) and Pinel (2010) presented results of simulations performed in the salt and overthrust models. Plessix (2007) used an eight-order accurate finite-difference stencil allowing for a discretization of five grid points per wavelength. Sponge-like absorbing boundary conditions require up to 35 points in the sponge layers. Pinel (2010) used a seven-point finite-difference stencil, which requires 12 grid points per minimum wavelength and 16 grid points in the perfectly matched layers. The differences in the finite-difference stencils, in the discretization, and in the absorbing boundary conditions used in Plessix (2007), Pinel (2010) and in this study make the size of the finite-difference grids different for a given frequency and a given physical

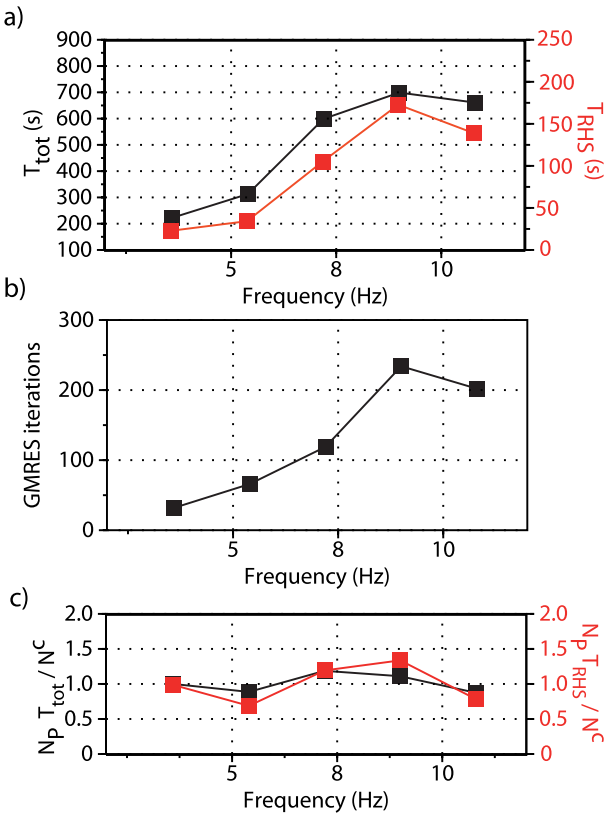

Figure 12 Hybrid solver modelling complexity (overthrust model). The same representation is used as in Fig. 8. See also Table 4.

domain. Moreover, the solution accuracies controlled by the stopping criterion of iterations can also be different in the three studies just mentioned. Therefore, the conclusions that can be inferred from the comparison between the three studies are highly speculative. The results of Plessix (2007), Pinel (2010) and this study are outlined in Table 4 for simulations in the salt and overthrust models.

One tentative conclusion is that the iterative solver has a better complexity that the hybrid solver approach, since the iterative solver shows improved relative performance with respect to the hybrid solver as the size of the computational domain increases (for example, compare the results of Plessix (2007) and our results in the salt and in the overthrust models at 5 and 10 Hz). Another advantage of the iterative solver is clearly the low-memory requirement since the geometric multigrid solver can be implemented in a matrix-free fashion; this allows for an efficient parallelism over shots. We notice however that using unstructured meshes will strongly affect these solvers while it will not change much the hybrid solver. A second level of parallelism can be viewed by means of the parallelization of the matrix-vector products performed by the iterative solver (Riyanti *et al.* 2007). A drawback of the iterative solver is that the impedance matrix should be less well conditioned than the Schur complement system. If the number of sources can be reduced by source encoding such that the number of blended sources is significantly smaller than the number of available processors, one can take advantage of the high scalability and flexibility of the hybrid solver to design a domain decomposition, which will provide the best trade-off between the computational costs of the source-independant tasks and of the source-dependent tasks. This might be an advantage compared to conventional iterative methods, where the full computational cost is controlled by the iterative resolution.

Two-level parallelism in the hybrid solver

As usually observed, increasing the number of subdomains leads to increasing the number of iterations to converge. For a large number of subdomains/processors the convergence rate might be significantly deteriorated and the solution process becomes ineffective. In order to alleviate the numerical growth of the iterations, when the number of subdomains is increased to feed each processor, we might keep the number of subdomains small while handling each subdomain by more than one processor introducing 2-levels of parallelism in the hybrid solver modelling method (Haidar 2008; Giraud *et al.* 2010). Such an implementation enables us to express parallelism between the subdomains but also within the treatment of each subdomain. Concerning the source-independent tasks, the lower-upper factorizations of the local impedance matrices as well as the dense factorizations of the assembled local Schur complement matrices involved in the preconditioner can be performed in parallel by each group of processors. The local matrix-vector product performed by a generalized minimum residual solver during the source-dependent tasks can also be parallelized with linear algebra subroutines developed for high-performance computers (SCALAPACK and PBLAS libraries). In Table 5 we display preliminary results using a two-level parallel implementation. In these experiments with 192 processors, it can be observed that the source-independent tasks are slightly more time consuming using the two-level parallel approach. This is due to the fact that having fewer subdomains leads to larger local impedance matrices and larger subdomain interfaces; even though the source-independent tasks are performed using parallel kernels, the non-linear complexity of these kernels is not completely overcome by the parallel treatment. However, a significant saving can be observed for the source-dependent kernels. The gains are twofold. First we have fewer subdomains, consequently the convergence is faster (less generalized minimum residual iterations). Furthermore, the source-dependent kernels have a computational complexity mainly linear with the problem size, they are naturally parallel and their parallel implementations are efficient. For this example, the two-level parallel implementation is more effective than the one-level as soon as more than two sources have to be considered.

Future work

The HSM method can be improved within several ways in the framework of full waveform inversion applications. A first improvement of the hybrid solver modelling method concerns the efficiency of the multisource and multifrequency modellings. In addition to the two-level implementation, the multisource modelling can be improved by using the block generalized minimum residual method where multiple sources are efficiently processed thanks to the BLAS3 library, on the one hand and block Krylov subspace on the other (Saad 2003). Multifrequency modelling can be required in frequency-domain full waveform inversion when the simultaneous inversion of multiple close frequencies improves the reconstruction (see Brossier, Operto and Virieux 2009 and Ben-Hadj-Ali *et al.* 2011 for two illustrations). Various numerical techniques based on recycling Krylov subspace information between various solution exist, either through subspace augmentation (Chapman and

Table 5 Comparison between the one-level and two-level parallel implementation of the hybrid solver. N_i: number of interface points. T_p: elapsed time for the source-independent tasks. T_r: elapsed time for the source-dependent tasks

Implementation	# subdomains	proc. per subdomain	subdomain		$T_p(s)$	$T_r(s)$
			$n_x \times n_y \times n_z$	N_i		
1-level	192	1	$33 \times 33 \times 21$	5578	168	86
2-levels	96	2	$45 \times 33 \times 31$	7405	254	45

Sadd 1997; Parks *et al.* 2006) or through incremental deflating preconditioners (Giraud, Gratton and Martin 2007). Investigating those techniques in a full waveform inversion framework would deserve to be undertaken in the future. It is still unclear whether the computed solution in a given subsurface model can be exploited to speedup the modelling in an updated model of the subsurface after one inversion iteration.

A last issue concerns the choice of the preconditioner. The algebraic preconditioner does not exploit the modelled physical phenomena. An alternative could be provided by using non-reflecting boundary conditions such as perfectly matched layers at interfaces between subdomains to design a genuine domain decomposition approach (Heikkola *et al.* 1998).

CONCLUSION

We have presented a massively parallel algebraic domain decomposition method based on a hybrid direct-iterative solver for 3D frequency-domain visco-acoustic wave modelling. Compared to modelling approaches based on sparse direct solvers, the hybrid-solver approach scales well on large-scale distributed memory platforms and has a lower memory requirement, which decreases with the number of processors. Therefore, the hybrid approach allows one to tackle problems of larger dimensions in terms of model size or modelled frequencies than the direct-solver approach. However, the direct solver method remains faster, when the lower-upper factorization remains tractable on a limited number of computer nodes, that is, for problems involving a few million of unknowns. Both theoretical complexity analysis and numerical experiments suggest that the computational cost of the hybrid approach is of the same order as iterative methods or time-marching methods for single-frequency multi-source modelling on distributed-memory platforms. The memory cost of the hybrid approach for one source modelling is significantly higher than that of iterative solvers and time-marching algorithms. This limits the ability of the hybrid-solver method to distribute sources over processors in the framework of multi-source modelling. However, the hybrid-solver method allows for the iterative resolution of a reduced (i.e., better preconditioned) system than conventional iterative methods and, therefore, should show improved convergence in complex media. The hybrid approach should be quite efficient for full waveform inversion applications, where the number of sources can be reduced with source encoding techniques. In this framework, one can take advantage of the high scalability and the flexibility of the hybrid solver to design the most suitable domain decomposition for the number of available processors, the number of blended sources and the model dimension. The domain decomposition can be chosen such that the computational time is minimized, which is equivalent to finding the best trade-off between the computational costs of the source-independent tasks (local lower-upper factorizations, preconditioner building) and of the source-dependant tasks (multisource resolution of the Schur complement system) in the hybrid solver. At least two improvements of the method can be identified and they concern the implementation of a two-level parallelism and the efficient processing of multiple sources with block iterative solvers.

ACKNOWLEDGEMENTS

We thank the associate editor and anonymous reviewers for their comments, which helped to improve the manuscript. We are grateful to Romain Brossier (ISTerre, France) with whom we had fruitful discussions. This work was granted access to the HPC resources of [CCRT/CINES/IDRIS] under allocation 2009-82280 made by GENCI (Grand Equipement National de Calcul Intensif) and of Mesocentre SIGAMM (http://crimson.oca.eu), hosted by Observatoire de la Cote d'Azur. We thank P. Amestoy (ENSEEIHT-IRIT) and J.Y-L'Excellent (INRIA-ENS Lyon) for providing us the MUMPS solver. We thank the CERFACS for providing us the GMRES solver. Finally, this work was carried out within the frame

of the SEISCOPE consortium (http://seiscope.unice.fr) sponsored by BP, CGGVeritas, Eni, ExxonMobil, Saudi Aramco, Shell, Statoil and Total. This work was partly funded by the ANR project ANR-05-NT05-2-42427.

REFERENCES

Aminzadeh F., Brac J. and Kunz T. 1997. 3-D Salt and overthrust models. SEG/EAGE 3-D Modeling Series No. 1.

Ben-Hadj-Ali H. 2009. *Three dimensional visco-acoustic frequency domain full waveform inversion*. PhD thesis, Université de Nice-Sophia-Antipolis.

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

Ben-Hadj-Ali H., Operto S., Virieux J. and Sourbier F. 2008. 3D frequency-domain full-waveform tomography based on a domain decomposition forward problem. 78th SEG meeting, Las Vegas, Nevada, USA, Expanded Abstracts.

Berenger J.-P. 1994. A perfectly matched layer for absorption of electromagnetic waves. *Journal of Computational Physics* **114**, 185–200.

Brossier R., Etienne V., Operto S. and Virieux J. 2010. Frequency-domain numerical modelling of visco-acoustic waves based on finite-difference and finite-element discontinuous Galerkin methods. In: *Acoustic Waves* (ed. D. W., Dissanayake), pp. 125–158. SCIYO.

Brossier R., Operto S. and Virieux J. 2009. Seismic imaging of complex onshore structures by 2D elastic frequency-domain full-waveform inversion. *Geophysics* **74**, WCC63–WCC76.

Calandra H., Duff I., Gratton S., Pinel X. and Vasseur X. 2008. Massively parallel computations for the solution of the 3D-Helmholtz equation in the frequency domain. 62nd EAGE meeting, Glasgow, UK, Expanded Abstracts.

Carvalho L.M., Giraud L. and Meurant G. 2001. Local preconditioners for two-level non-overlapping domain decomposition methods. *Numerical Linear Algebra with Applications* **8**, 207–227.

Chapman A. and Saad Y. 1997. Deflated and augmented krylov subspace techniques. *Numerical Linear Algebra with Applications* **4**, 43–66.

Duff I.S., Erisman A.M. and Reid J.K. 1986. *Direct Methods for Sparse Matrices*. Clarendon Press.

Duff I.S., Gratton S., Pinel X. and Vasseur X. 2007. Multigrid based preconditioners for the numerical solution of two-dimensional heterogeneous problems in geophysics. *International Journal of Computer Mathematics* **84–88**, 1167–1181.

Erlangga Y.A. and Nabben R. 2008. On a multilevel Krylov method for the Helmholtz equation preconditioned by shifted Laplacian. *Electronic Transactions on Numerical Analysis* **31**, 403–424.

Erlangga Y.A., Oosterleeand C. and Vuik C. 2006. A novel multigrid based preconditioner for heterogeneous Helmholtz problems. *SIAM – Journal of Scientific Computing* **27**, 1471–1492.

George A. and Liu J.W. 1981. *Computer Solution of Large Sparse Positive Definite Systems*. Prentice-Hall.

Giraud L., Gratton S. and Martin E. 2007. Incremental spectral preconditioners for sequences of linear systems. *Applied Numerical Mathematics* **57**, 1164–1180.

Giraud L., Haidar A. and Pralet S. 2010. Using multilevel of parallelism to enhance the performance of domain decomposition solvers. *Parallel Computing* **36**, 285–296.

Haidar A. 2008. *On the parallel scalability of hybrid linear solvers for large 3D problems*. PhD thesis, Institut National Polytechnique de Toulouse.

Heikkola E., Kuznetsov Y.A., Neittaanmäki P. and Toivanen J. 1998. Fictitious domain methods for the numerical solution of two-dimensional scattering problems. *Journal of Computational Physics* **145**, 89–109.

Hustedt B., Operto S. and Virieux J. 2004. Mixed-grid and staggered-grid finite difference methods for frequency domain acoustic wave modelling. *Geophysical Journal International* **157**, 1269–1296.

Jo C.H., Shin C. and Suh J.H. 1996. An optimal 9-point, finite-difference, frequency-space 2D scalar extrapolator. *Geophysics* **61**, 529–537.

Komatitsch D. and Martin R. 2007. An unsplit convolutional perfectly matched layer improved at grazing incidence for the seismic wave equation. *Geophysics* **72**, SM155–SM167.

Krebs J., Anderson J., Hinkley D., Neelamani R., Lee S., Baumstein A. and Lacasse M.D. 2009. Fast full-wavefield seismic inversion using encoded sources. *Geophysics* **74**, WCC105–WCC116.

Lailly P. 1984. The seismic inverse problem as a sequence of before stack migrations: inverse scattering. In: *Theory and Application*, pp. 206–220. SIAM.

Levander A.R. 1988. Fourth-order finite-difference P-SV seismograms. *Geophysics* **53**, 1425–1436.

Magolumonga Made M., Beauwens R. and Warzée G. 2000. Preconditioning of discrete helmholtz operators perturbed by a diagonal complex matrix. *Communications in Numerical Methods in Engineering* **16**, 801–817.

Marfurt K. 1984. Accuracy of finite-difference and finite-elements modeling of the scalar and elastic wave equation. *Geophysics* **49**, 533–549.

MUMPS-team. 2009. MUMPS – MUltifrontal Massively Parallel Solver users' guide, version 4.9.2 (november 2009). ENSEEIHT INRIA, http://www.enseeiht.fr/apo/MUMPS/ or http://graal.ens-lyon.fr/MUMPS.

Nihei K.T. and Li X. 2007. Frequency response modelling of seismic waves using finite difference time domain with phase sensitive detection (TD-PSD). *Geophysical Journal International* **169**, 1069–1078.

Operto S., Virieux J., Amestoy P., L'Éxcellent J.-Y., Giraud L. and Ben-Hadj-Ali H. 2007. 3D finite-difference frequency-domain modeling of visco-acoustic wave propagation using a massively parallel direct solver: A feasibility study. *Geophysics* **72**, SM195–SM211.

Parks M., de Sturler E., Mackey G., Johnson D. and Maiti S. 2006. Recycling Krylov subspaces for sequences of linear systems. *SIAM Journal on Scientific Computing* **28**, 1651–1674.

Pinel X. 2010. *A perturbed two-level preconditioner for the solution of three-dimensional heterogeneous Helmholtz problems with applications to geophysics*. PhD thesis, University of Toulouse.

Plessix R.E. 2007. A Helmholtz iterative solver for 3D seismic-imaging problems. *Geophysics* **72**, SM185–SM194.

Plessix R.E. 2009. Three-dimensional frequency-domain full-waveform inversion with an iterative solver. *Geophysics* **74**, WCC53–WCC61.

Plessix R.E. and Perkins C. 2010. Full waveform inversion of a deep water ocean bottom seismometer dataset. *First Break* **28**, 71–78.

Pratt R.G. 1999. Seismic waveform inversion in the frequency domain, part I: theory and verification in a physic scale model. *Geophysics* **64**, 888–901.

Pratt R.G. and Worthington M.H. 1990. Inverse theory applied to multi-source cross-hole tomography. Part I: acoustic wave-equation method. *Geophysical Prospecting* **38**, 287–310.

Riyanti C.D., Erlangga Y.A., Plessix R.E., Mulder W.A., Vuik C. and Oosterlee C. 2006. A new iterative solver for the time-harmonic wave equation. *Geophysics* **71**, 57–63.

Riyanti C.D., Kononov A., Erlangga Y.A., Vuik C., Oosterlee C., Plessix R.E. and Mulder W.A. 2007. A parallel multigrid-based preconditioner for the 3D heterogeneous high-frequency Helmholtz equation. *Journal of Computational physics* **224**, 431–448.

Saad Y. 2003. *Iterative Methods for Sparse Linear Systems*. SIAM.

Saad Y. and Schultz M.H. 1986. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* **7**, 856–869.

Simoncini V. and Gallopoulos S. 1996. A hybrid block GMRES method for nonsymmetric systems with multiple right-hand sides. *Journal of Computational and Applied Mathematics* **36**, 457–469.

Sirgue L., Barkved O.I., Dellinger J., Etgen J., Albertin U. and Kommedal J.H. 2010. Full waveform inversion: the next leap forward in imaging at Valhall. *First Break* **28**, 65–70.

Sirgue L., Etgen J.T. and Albertin U. 2008. 3D Frequency domain waveform inversion using time domain finite difference methods. 70th EAGE meeting, Rome, Italy, Expanded Abstracts, F022.

Sirgue L. and Pratt R.G. 2004. Efficient waveform inversion and imaging: a strategy for selecting temporal frequencies. *Geophysics* **69**, 231–248.

Smith B.F., Bjørstad P.E. and Gropp W. 1996. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press.

Stekl I. and Pratt R.G. 1998. Accurate viscoelastic modeling by frequency-domain finite difference using rotated operators. *Geophysics* **63**, 1779–1794.

Tarantola A. 1984. Inversion of seismic reflection data in the acoustic approximation. *Geophysics* **49**, 1259–1266.

Van der Vorst H.A. 1992. BI-CGSTAB: a fast and smoothly converging variant of bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* **13**, 631–644.

Virieux J. and Operto S. 2009. An overview of full waveform inversion in exploration geophysics. *Geophysics* **74**, WCC127–WCC152.

Virieux J., Operto S., Ben-Hadj-Ali H., Brossier R., Etienne V., Sourbier F., Giraud L. and Haidar A. 2009. Seismic wave modeling for seismic imaging. *The Leading Edge* **28**, 538–544.

Wang S., Xia J. and de Hoop M.V. 2010. A 3d massively parallel structured approximate direct helmholtz solver: algorithms & methods. 80th SEG meeting, Denver Colorado, USA, Expanded Abstracts, 1039–1043.

Wilkinson J.H. 1963. *Rounding Errors in Algebraic Processes*. Prentice-Hall.