MULTISCALE MODEL. SIMUL.
Vol. 4, No. 4, pp. 1305–1338

OPERATOR UPSCALING FOR THE ACOUSTIC WAVE EQUATION*

TETYANA VDOVINA†, SUSAN E. MINKOFF†, AND OKSANA KOROSTYSHEVSKAYA†

Abstract. Modeling of wave propagation in a heterogeneous medium requires input data that varies on many different spatial and temporal scales. Operator-based upscaling allows us to capture the effect of the fine scales on a coarser domain without solving the full fine-scale problem. The method applied to the constant density, variable sound velocity acoustic wave equation consists of two stages. First, we solve small independent problems for approximate fine-scale information internal to each coarse block. Then we use these subgrid solutions to define an upscaled operator on the coarse grid. The fine-grid velocity field is used throughout the process (i.e., no averaging of input fields is required). An equivalence between the variational form of the problem and a staggered finite-difference scheme allows us to use finite differences to solve the subgrid wave propagation problems. Due to the homogeneous Neumann boundary conditions imposed on each coarse block, the subgrid problems decouple, which leads to the natural parallelization of the first stage of the method. The algorithm requires none of the ghost cell (edge) communication required by standard data parallelism. Timing studies indicate that the parallel algorithm has near optimal speedup. Three variable velocity numerical experiments illustrate that operator-based upscaling captures the essential fine-scale information (even details contained within a single coarse grid block) and models wave propagation quite accurately at considerably less expense than full finite differences. The resulting coarse-grid solution has the advantage of being less prohibitive to store and manipulate.

Key words. upscaling, multiscale methods, acoustic wave propagation, parallel computing, seismology

AMS subject classifications. 35L05, 86-08, 65M06, 86A15

DOI. 10.1137/050622146

1. Introduction.

To model physical systems in science and engineering one uses input data which often varies on many different spatial scales [11]. The direct numerical simulation of problems involving multiple scales is usually not feasible in reasonable time due to computational power limitations. To deal with this limitation one might solve the problem on a coarser scale than the one on which the parameters are originally defined or measured. *Upscaling* is the process of redefining the physical system's parameters up to a coarser grid, forming *effective* or *equivalent* parameters [11], [21].

There is a large literature on upscaling techniques. The simplest and the most common method is arithmetic or harmonic averaging [11]. One averages the fine-grid physical parameters and then solves the equation(s) on a coarse grid. The performance of the method is reliable in certain cases but is completely unsatisfactory in others.

The renormalization technique has been developed to overcome the inadequacies of simple averaging [11], [21]. The method breaks a large problem into a hierarchy of manageable problems and solves them at several stages. At each stage the current grid is subdivided into coarse cells, each consisting of a small number of fine cells. The effective parameters on each coarse cell are determined by averaging. At the next

*Received by the editors January 6, 2005; accepted for publication (in revised form) July 11, 2005; published electronically December 7, 2005. This research was supported by the Collaborative Math-Geoscience Program at the NSF (grant EAR-0222181).
http://www.siam.org/journals/mms/4-4/62214.html
†Department of Mathematics and Statistics, University of Maryland Baltimore County, 1000 Hilltop Circle, Baltimore, MD 21250 (tvdovin1@math.umbc.edu, sminkoff@math.umbc.edu, oksana1@math.umbc.edu).

stage a group of coarse cells becomes a single grid block, and the whole process is repeated.

Another class of methods is based on homogenization theory [8], [9], [23], [1]. The central goal in this case is to obtain accurate coarse-scale solutions efficiently without directly solving the full fine-scale problem. Homogenization assumes that the medium under study depends on two separate spatial scales. In this case the solution can be asymptotically expanded with respect to a small parameter ϵ, which characterizes the fine scale of the problem. This expansion leads to a homogenized operator with constant (*effective*) coefficients. The range of application of homogenization methods is limited by two key assumptions:

1. Parameters in the model vary on two distinct scales.
2. The medium is periodic.

The multiscale finite element method [10], [17] is closely related to homogenization. The main idea is to construct finite element basis functions which capture the fine-scale information within each element. The basis functions are constructed from the leading-order, homogeneous elliptic equation in each element. The fine-scale information is then included in the coarse scale through the global stiffness matrix. In addition "oversampling" has been developed to allow relaxation of assumption 1 above [10], [32].

An alternative approach, known as mortar upscaling, was developed in the spirit of domain decomposition [28], [33]. The method consists of decomposing the physical domain into a number of blocks and using independently constructed numerical grids and different discretization techniques in each block. As a result, fine grids internal to coarse blocks can be nonmatching from one coarse block to another. Mortar finite element spaces are used to impose physically meaningful matching conditions on block interfaces. The resulting coarse-grid solution incorporates local fine-grid information.

The operator-based upscaling technique was first introduced for elliptic equations in [7] and later developed in [2], [3], [4], [6]. This method does not produce new effective coefficients on the coarse grid. Instead, an effective *solution* is defined directly. The central idea is to use a two-scale decomposition of the solution. The first (coarse) component is defined on the coarse computational grid. The second (subgrid) component is defined on the fine scale within each coarse element. The numerical algorithm based on this decomposition consists of two stages. First, the problem is solved for the fine-scale (subgrid) unknowns internal to each coarse grid block. Then these solutions are used to redefine the coarse-grid (upscaled) operator.

We have adapted the two-scale operator-based upscaling technique to the acoustic wave equation. The method has been developed in the context of mixed finite elements. The choice of finite element spaces enables us to solve each of the subgrid problems independently of the others, which leads to natural parallelization of the method [4]. However, finite-difference methods are often preferred for solving the wave equation. They are less computationally expensive and easier to implement than finite element methods in general. We have made use of an equivalence between the finite element formulation of the subgrid problems and finite differences [30]. Therefore, in the first stage of the method we are able to solve each of the subgrid problems using finite differences.

At the second stage of the upscaling algorithm, we use the subgrid solution found for each coarse block to modify the coarse problem solved on the total domain. We use the original boundary conditions on the physical domain in Step 2 of the algorithm, and the fine-grid parameter field in both stages of the algorithm. *The method allows us to capture local information accurately without solving the original full fine-*

grid problem. The main cost savings in operator upscaling is made evident in the parallel version of the algorithm, which provides near perfect speedup without the communication overhead standard data parallelism requires.

In section 2 we introduce the model problem (the acoustic wave equation) in strong and variational form. Section 3 contains a description of the method. Using a two-scale decomposition of the solution, we formulate the subgrid problems and then apply a numerical Green's function technique to find the subgrid solution in terms of the coarse unknowns. Finally, we formulate the upscaled problem on the coarse computational grid using the fine-scale information from the subgrid stage. The numerical implementation of the method (both serial and parallel versions) and associated cost estimates are discussed in section 4. Section 5 contains numerical examples which compare the upscaled and full finite-difference solutions for three different variable velocity earth models. We see that operator-based upscaling does an excellent job of capturing details of the solution even when fine-scale heterogeneities (features within a single coarse grid block) are present in the earth model. While the algorithm is more complex than a straightforward finite-difference solution of the wave equation, timing tables show the cost savings associated with the parallel version of the algorithm. A problem of size 3600×3600 can be solved 5 times faster when upscaled to 60×60 grid blocks and solved on 12 processors than it could be using straight finite differences. Perhaps more importantly, the coarse-grid solution requires less memory for storage. Working with a solution that captures fine-scale details but lives on a grid of size 60×60 is more computationally tractable than working with a solution of size 3600×3600.

2. Operator-based upscaling for the acoustic wave equation. Our ultimate goal is to develop a method that captures the effect of the fine scale on the coarse scales but does not require exact resolution of all fine-scale details. In the one-dimensional case, the operator-based upscaling technique reduces to standard finite differences. (This fact is discussed in detail in [22].) To illustrate the method on relatively realistic problems, we focus on the two-dimensional case.

As we mentioned above, the implementation is based on a mixed variational form. We construct finite element spaces in such a way that we can separate scales and decompose the solution into coarse and subgrid parts. The numerical method based on this decomposition produces the coarse solution that incorporates fine-scale information. The algorithm consists of two stages. At the first stage, the fine-scale problems internal to each coarse block are solved using numerical Green's functions [18]. At the second stage, we solve a matrix problem involving only the coarse-scale unknowns.

2.1. The acoustic wave equation. For our model problem, let Ω be a rectangular domain with boundary Γ, and consider the constant density, variable sound velocity acoustic wave equation:

$$\frac{1}{c^2}\frac{\partial^2 p}{\partial t^2} - \Delta p = f, \tag{2.1}$$

where p is pressure, $c(x, y)$ is the sound velocity (a function of spatial location only), and f is a source of acoustic energy [12].

This equation can be rewritten as a first-order system by introducing acceleration $\mathbf{u}$ as a spatial gradient of pressure:

$$\mathbf{u} = -\nabla p, \tag{2.2}$$

$$\frac{1}{c^2(x,y)}\frac{\partial^2 p}{\partial t^2} + \nabla \cdot \mathbf{u} = f. \tag{2.3}$$

To this set of differential equations we add the following boundary and initial conditions:

$$\mathbf{u} \cdot \boldsymbol{\nu} = 0 \quad \text{on } \Gamma, \tag{2.4}$$

$$p(0,x,y) = p_0, \qquad \frac{\partial p(0,x,y)}{\partial t} = p_1, \tag{2.5}$$

where $\boldsymbol{\nu}$ is the unit outward normal to the boundary Γ. (Note that the upscaling method can easily be applied to the wave equation with any set of boundary and initial conditions prescribed on the total domain. These conditions were chosen for simplicity.)

Our method is based on a mixed finite element variational form. Let $H_0(\mathrm{div};\Omega)$ be the set of vector functions $\mathbf{V} \in (L^2(\Omega))^2$ such that $\nabla\cdot\mathbf{v} \in L^2(\Omega)$ and $\mathbf{v}\cdot\boldsymbol{\nu} = 0$ on Γ. To obtain the variational form of our problem, multiply (2.2) and (2.3) by $\mathbf{v} \in H_0(\mathrm{div};\Omega)$ and $w \in L^2(\Omega)$, respectively, integrate over Ω, and apply the divergence theorem to (2.2).

Then problem (2.2)–(2.3) reduces to finding $\mathbf{U} \in H_0(\mathrm{div};\Omega)$ and $P \in L^2(\Omega)$ such that

$$(\mathbf{U},\mathbf{v}) - (P, \nabla\cdot\mathbf{v}) = 0, \tag{2.6}$$

$$\left(\frac{1}{c^2}\frac{\partial P^2}{\partial t^2}, w\right) + (\nabla\cdot\mathbf{U}, w) = (f,w) \tag{2.7}$$

for all $\mathbf{v} \in H_0(\mathrm{div};\Omega)$ and $w \in L^2(\Omega)$.

2.2. Finite element spaces. To define the mixed finite element spaces, we first construct a two-scale mesh [4]. Decompose Ω into a coarse mesh $\mathcal{T}_H(\Omega)$, and then decompose each coarse element E_c into the subgrid mesh $\mathcal{T}_h(E_c)$. Our composite fine mesh is defined by

$$\mathcal{T}_h(\Omega) = \bigcup_{E_c \in \mathcal{T}_H(\Omega)} \mathcal{T}_h(E_c). \tag{2.8}$$

The acceleration finite element space over $\mathcal{T}_h(\Omega)$ is given by

$$\mathbf{V}_{H,h} = \mathbf{V}_H \oplus \delta\mathbf{V}_h, \tag{2.9}$$

where $\mathbf{V}_H \subset H_0(\mathrm{div};\Omega)$ is defined over Ω, and $\delta\mathbf{V}_h \subset H(\mathrm{div};E_c)$ is defined for each coarse element. Both spaces consist of linear functions of the form $(\alpha_1 x+\beta_1, \alpha_2 y+\beta_2)$ with nodes at the edges of the coarse cells for $\mathbf{V}_H$ and at the edges of the fine cells for $\delta\mathbf{V}_h$ (see Figure 1). Thus, the full finite element space is $\mathbf{V}_{H,h} \times W_h$, defined over $\mathcal{T}_h(\Omega)$.

We will refer to solution of the subgrid problems as "Step 1" of the algorithm. The solution of the upscaled or coarse problem which includes this subgrid information will be referred to as "Step 2." In order to localize the fine-scale information and separate the subgrid problems, we impose homogeneous Neumann boundary conditions on each coarse block in Step 1 of the algorithm:

$$\delta\mathbf{v} \cdot \boldsymbol{\nu} = 0 \quad \text{on } \partial E_c. \tag{2.10}$$

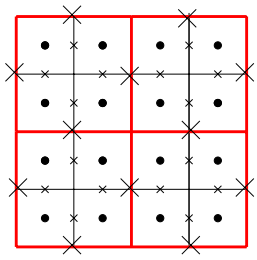


FIG. 1. *The large crosses represent the nodes in the coarse acceleration space* $\mathbf{V}_H$*, and the small crosses represent the nodes for the subgrid acceleration space* $\delta\mathbf{V}_h(E_c)$*. The dots represent the nodes for the pressure space* W_h*.*

This choice of boundary condition leads to an efficient parallel implementation of the method, as we can solve for subgrid information internal to each coarse cell independently of every other coarse cell. (In other words, blocks of coarse cells can be parcelled out to different processors and the subgrid problems solved without requiring communication between the processors.) The purpose of upscaling is to solve a computationally intensive problem more efficiently (in terms of computer time and possibly memory). Thus some simplification is required, or one will be solving the full fine-grid problem at the usual cost. On scales of interest to geophysicists (centimeters to kilometers) the earth is neither periodic nor would it satisfy zero Neumann conditions. Neither type of boundary condition imposed on the coarse grid blocks is terribly realistic physically. Our goal is to find a way to speed up solution of the problem. The simplest such choice is zero Neumann conditions imposed in Step 1 of the algorithm on coarse block edges. This choice results in a decoupling of the subgrid problems between coarse blocks. However, other choices of boundary conditions could be investigated. It is likely that solution of the resulting subgrid system will be more complicated for most other choices of boundary conditions because some communication will be required between coarse cells in Step 1. Nonetheless, we emphasize that these boundary conditions are only imposed when solving the subgrid problems. In Step 2 of the algorithm, the subgrid solutions are used in solving the coarse problems. And the coarse problem is solved using the original physical boundary conditions imposed on the total domain. Thus the boundary condition simplification occurs at an intermediate stage of the algorithm and is in some sense compensated for by the second stage of the upscaling algorithm (solution of the coarse problem). (For a detailed analysis see Korostyshevskaya and Minkoff [22].)

In this work we upscale acceleration only. We could upscale pressure as well (decomposing the pressure space into the direct sum of coarse and subgrid parts as was done in [7]). In that work, the elliptic pressure equation was upscaled for both pressure and velocity. The subgrid basis functions for pressure (δw) were chosen to be orthogonal on a single coarse cell (i.e., $\int_{E_c} \delta w = 0$). For example, if we have four fine cells in a single coarse block, there are three orthogonal basis functions. These functions have value $+1$ on one of the four fine-grid cells and value -1 on another. The basis function has zero support on the other two fine-grid cells. See Figure 2 for a schematic of these functions. Much greater computational complexity arises from the fact that these basis functions all have some common support. Thus the resulting stiffness matrix will be less sparse if we upscale both variables (pressure and its gradient). The work we describe in this paper attempts to lay out the method for the wave equation. It is a first attempt to do so. (We have followed Arbogast in his

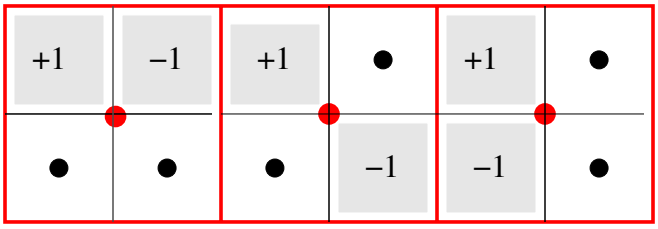


FIG. 2. *Three basis functions for upscaling pressure which would be used if each coarse grid block contained* 2 *fine blocks in* x *and* 2 *in* y.

more recent work [3], in which he upscales velocity but pressure lives only on the fine grid. A coarse-grid projection of pressure can also be computed.) Efforts are ongoing to analyze the upscaling problem for the wave equation when both acceleration and pressure are upscaled.

3. Method description.

3.1. The subgrid scale. Since $\mathbf{V}_{H,h} = \mathbf{V}_H \oplus \delta\mathbf{V}_h$, we can decompose $\mathbf{U}$ (the acceleration unknowns) uniquely into

$$\mathbf{U} = \mathbf{U}_H + \delta\mathbf{U}, \tag{3.1}$$

where $\mathbf{U}_H \in \mathbf{V}_H$ and $\delta\mathbf{U} \in \delta\mathbf{V}_h$.

If we restrict to the subgrid test functions in (2.6)–(2.7) and use decomposition (3.1) for the unknowns, we obtain a series of subgrid problems, one for each coarse element E_c:

$$\left(\frac{1}{c^2}\frac{\partial^2 P}{\partial t^2}, w\right)_{E_c} + (\nabla\cdot(\delta\mathbf{U}+\mathbf{U}_H), w)_{E_c} = (f,w)_{E_c}, \tag{3.2}$$

$$(\delta\mathbf{U}+\mathbf{U}_H, \delta\mathbf{v})_{E_c} - (P, \nabla\cdot\delta\mathbf{v})_{E_c} = 0 \tag{3.3}$$

for all $\delta\mathbf{v} \in \delta\mathbf{V}(E_c)$ and $w \in W(E_c)$. We would like to use finite differences rather than finite elements to solve (3.2)–(3.3).

If we use second-order finite differences to approximate the derivative with respect to time, then problem (3.2)–(3.3) is equivalent to the second-order in space and time, staggered finite-difference problem [30]:

$$\begin{aligned}&\frac{P\big|^{k+1}_{i+\frac{1}{2},j+\frac{1}{2}} - 2P\big|^{k}_{i+\frac{1}{2},j+\frac{1}{2}} + P\big|^{k-1}_{i+\frac{1}{2},j+\frac{1}{2}}}{\Delta t^2}\\ &= -c^2_{i+\frac{1}{2},j+\frac{1}{2}}\left[(D_x+D_y)(\delta\mathbf{U}+\mathbf{U}_H)\Big|^{k}_{i+\frac{1}{2},j+\frac{1}{2}} - f^k_{i+\frac{1}{2},j+\frac{1}{2}}\right],\end{aligned} \tag{3.4}$$

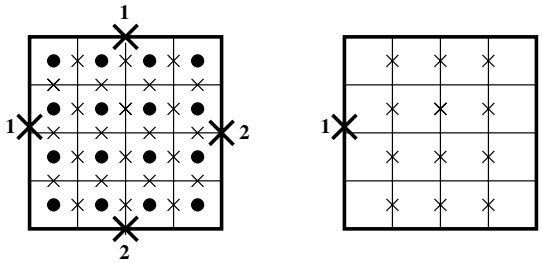
$$\delta U^x\Big|^{k+1}_{i,j+\frac{1}{2}} = -U^x_H\Big|^{k+1}_{i,j+\frac{1}{2}} - D_x\, P\big|^{k+1}_{i,j+\frac{1}{2}}, \tag{3.5}$$

$$\delta U^y\Big|^{k+1}_{i+\frac{1}{2},j} = -U^y_H\Big|^{k+1}_{i+\frac{1}{2},j} - D_y\, P\big|^{k+1}_{i+\frac{1}{2},j}, \tag{3.6}$$

$$\delta U^x\Big|^{k+1}_{0,j+\frac{1}{2}} = \delta U^x\Big|^{k+1}_{N_x,j+\frac{1}{2}} = 0, \tag{3.7}$$

$$\delta U^y\Big|^{k+1}_{i+\frac{1}{2},0} = \delta U^y\Big|^{k+1}_{i+\frac{1}{2},N_y} = 0, \tag{3.8}$$

where k corresponds to the time level, D_x and D_y denote first-order, centered, finite-difference operators, and N_x and N_y correspond to the number of the subgrid cells



(a) (b)

FIG. 3. *Coarse cell E_c with 4×4 fine-grid elements.* (a) *Large crosses represent the nodes for coarse acceleration basis functions v_1^x and v_2^x in the x-direction and v_1^y and v_2^y in the y-direction (the influence functions).* (b) *Small crosses represent the subgrid nodes influenced by the basis function v_1^x.*

inside each coarse cell in the x- and y-directions, respectively. The local indices i and j range from 0 to N_x in x and 0 to N_y in y, and these indices denote the location of the unknown inside a particular coarse cell.

At this point, our goal is to solve problem (3.4)–(3.8) on the $(k+1)$st time level for the subgrid unknowns only. Therefore, we need to eliminate the coarse unknowns from (3.5)–(3.6). The numerical Green's function or influence function technique [6], [18] described below allows us to solve the subgrid equations and begin the upscaling process.

The numerical Green's function technique consists of breaking problem (3.4)–(3.8) into two sets of subproblems using superposition of solutions. This approach allows us to determine the local fine-scale information independently of the coarse grid. First, we assume that the coarse-scale information on the $(k+1)$st time level is set to zero:

$$P\Big|_{i+1/2,j+1/2}^{k+1} = 2P\Big|_{i+1/2,j+1/2}^{k} - \Big|P_{i+1/2,j+1/2}^{k-1} \tag{3.9}$$
$$-c_{i+1/2,j+1/2}^2 \Delta t^2 \left[(D_x + D_y)\left(\delta \mathbf{U} + \mathbf{U}_H\right)\Big|_{i+1/2,j+1/2}^{k} - f\Big|_{i+1/2,j+1/2}^{k} \right],$$

$$\delta U_0^x\Big|_{i,j+1/2}^{k+1} = -D_x P\Big|_{i,j+1/2}^{k+1}, \tag{3.10}$$

$$\delta U_0^y\Big|_{i+1/2,j}^{k+1} = -D_y P\Big|_{i+1/2,j}^{k+1}. \tag{3.11}$$

The solution $(P, \delta \mathbf{U}_0)$ of this problem represents the influence of the physical source f.

To determine the effect of the coarse scale on the subgrid problem, we note from Figure 3(a) that each coarse cell has four coarse basis functions living on its edges (v_1^x and v_2^x in the x-direction and v_1^y and v_2^y in the y-direction). Since coarse acceleration is a linear combination of these four basis functions, the subgrid unknowns on a single coarse cell will only be affected by these four coarse functions. (Figure 3(b) shows the subgrid nodes affected by coarse basis function v_1^x.) If in system (3.4)–(3.8) we take $U_H^x|_{i+\frac{1}{2},j}^{k+1} = v_1^x|_{i+\frac{1}{2},j}$ (i.e., we assume the subgrid acceleration is the coarse basis function v_1^x evaluated at the subgrid nodes) and set all other coarse basis function information equal to zero, then the resulting system produces a solution that is influenced only by this particular coarse basis function. Via superposition we obtain a total solution by "turning on" each of these four coarse acceleration basis functions

(and the physical source) one at a time. (Note that the numerical Green's function or influence function idea is described in many of the papers by Arbogast. Specifically, a simple example illustrating this superposition principle is given in detail in [6].)

The subgrid acceleration in a single coarse cell is a linear combination of solutions to these five problems (from the physical source f and the four influence functions). This technique merely facilitates solution of the subgrid problems. The subgrid problem solutions are then used to solve for coarse acceleration in Step 2 of the algorithm. Pressure P is completely determined by (3.9)–(3.11).

3.2. The upscaled equation. Once we have obtained the solution of the subgrid problems, we restrict to the coarse-scale test functions in (2.6) and use decomposition (3.1) to obtain

$$(\mathbf{U}_H + \delta\mathbf{U}, \mathbf{v})_\Omega - (P, \nabla\cdot\mathbf{v})_\Omega = 0 \text{ for all } \mathbf{v} \in \mathbf{V}_H. \tag{3.12}$$

This equation results in a linear system for the coefficients β_n in the finite element representation of the coarse acceleration $\mathbf{U}_H$. Global index n takes values ranging from 1 to N_c, where N_c is the total number of coarse acceleration basis functions. In order to determine the structure of the system matrix, we use the coarse basis functions $\mathbf{v}_n$ as test functions and the finite element representation of $\mathbf{U}_H$ in (3.12). Since the hat basis function $\mathbf{v}_n$ is only supported on two coarse blocks E_c^l and E_c^r, the inner product of any two basis functions is zero, except for $(\mathbf{v}_n, \mathbf{v}_n)_{E_c^l \cup E_c^r}$, $(\mathbf{v}_{n-1}, \mathbf{v}_n)_{E_c^l}$, and $(\mathbf{v}_{n+1}, \mathbf{v}_n)_{E_c^r}$. Therefore, (3.12) reduces to

$$\sum_{s=-1}^{1} \beta_{n+s} \left(\mathbf{v}_{n+s}, \mathbf{v}_n\right) + \left(\delta\mathbf{U}^l + \delta\mathbf{U}^r, \mathbf{v}_n\right) - \left(P, \nabla\cdot\mathbf{v}_n\right) = 0, \tag{3.13}$$

where $\delta\mathbf{U}^l$ and $\delta\mathbf{U}^r$ denote the subgrid solutions internal to the coarse cells E_c^l and E_c^r, respectively. The subgrid solutions $\delta\mathbf{U}^l$ and $\delta\mathbf{U}^r$ involve the coarse coefficients β_{n-1}, β_n, and β_{n+1}, the solutions from the physical source ($\delta\mathbf{U}_0^l$ and $\delta\mathbf{U}_0^r$), and the numerical Green's functions ($\delta\mathbf{U}_n^l$ and $\delta\mathbf{U}_n^r$). Using this representation in (3.13), we obtain

$$\begin{aligned}\sum_{s=-1}^{1} \beta_{n+s} \left(\mathbf{v}_{n+s}, \mathbf{v}_n\right) + \sum_{s=-1}^{1} \beta_{n+s} \left(\delta\mathbf{U}_{n+s}^l + \delta\mathbf{U}_{n+s}^r, \mathbf{v}_n\right) \\ = \left(P, \nabla\cdot\mathbf{v}_n\right) + \left(\delta\mathbf{U}_0^l + \delta\mathbf{U}_0^r, \mathbf{v}_n\right)\end{aligned} \tag{3.14}$$

with inner products over the coarse cells E_c^l and E_c^r. The global numbering of coarse blocks within the numerical domain may differ depending on implementation. Here we assume that neighboring coarse blocks are numbered sequentially.

Let $\mathbf{b}$ denote the vector of unknown coarse coefficients β_n, M the tridiagonal stiffness matrix coming from the basis function inner products in (3.14), and $\mathbf{q}$ the right-hand side load vector from (3.14). In linear algebra notation, (3.14) becomes

$$M\mathbf{b} = \mathbf{q}. \tag{3.15}$$

In a companion paper by Korostyshevskaya and Minkoff [22], the operator-based upscaling method is analyzed as a linear algebra problem. In that paper, we treat the more general case of the acoustic wave equation with variable density and sound velocity. As the input parameters all live on the fine scale, all quadrature used is fine-grid quadrature. We show that when fine-grid quadrature is used, the matrix system

is in fact diagonal due to simplification of some of the matrix elements. These diagonal entries contain weighted density values which are never zero. Thus the system will always be solvable. The method described in this paper is a special case of the more general variable density and velocity situation and will also be solvable.

We have determined a new coarse-grid operator for solving the upscaled problem, rather than a new coarse-grid parameter field. Fine-scale information is included in this new operator through the subgrid solutions.

4. Numerical implementation and parallel algorithm.

4.1. Serial implementation. Operator-based upscaling consists of two stages. To solve the wave equation using this technique we need to solve the subgrid problems internal to each coarse block and then solve the modified (upscaled) problem defined on the coarse grid at each time level. We use an explicit finite-difference scheme to solve each of the subgrid problems. During the subgrid stage, the numerical grid is staggered in space with pressure unknowns defined at the centers of the fine cells and the horizontal and vertical components of the acceleration defined at the edges of the fine cells. The subgrid solutions are defined in terms of the coarse solutions (and vice versa). In order to begin the algorithm, we effectively eliminate the coarse unknowns from the subgrid problems via numerical Green's functions.

The use of numerical Green's functions at this stage also allows us to perform a significant amount of the subgrid computation outside the time-step loop. Recall that we break each of the subgrid problems into two subproblems. The first subproblem (equations (3.9)–(3.11)) generates the component of the subgrid solution which comes from the influence of the physical source. Equations (3.9)–(3.11) look similar to the original wave equation and are, therefore, time dependent. In contrast, the components of the subgrid solution from the numerical Green's function sources are completely independent of time and can be computed once outside the time-step loop. Finally, since these components are the same for each coarse block, it is sufficient to compute the numerical Green's function solutions for a single coarse block.

For the coarse-grid problem we note that the system matrix M is also time independent. From the left-hand side of (3.14) we see that the only contribution to the system matrix are the solutions from the numerical Green's functions and coefficients coming from the coarse-grid inner products $(\mathbf{v}_{n+s}, \mathbf{v}_n)$ for $s = -1, 0, 1$. Since the basis functions $\mathbf{v}_n$ are functions of spatial location only, the system matrix can be assembled outside the time-step loop. On the other hand, the right-hand side of (3.14) depends on time and must be updated at each time level. The algorithm can be summarized as follows:

1. *Preprocessing:*
   (i) Solve the subgrid problems to obtain the solutions from the numerical Green's functions.
   (ii) Assemble the system matrix for the coarse-grid linear system.
2. *For each time level:*
   (i) Loop over all the coarse blocks to obtain the subgrid solutions from the physical source f using the second-order in time and space staggered finite-difference scheme.
   (ii) Solve the coarse-grid linear system for the coefficients in the finite element representation of coarse acceleration using a tridiagonal linear solver.

4.2. Cost of serial algorithm. Since the total cost of the algorithm is dominated by the time-step loop, we restrict our cost estimates to these calculations. If we

denote the size of the coarse problem by N_c, then the cost of solving the coarse-grid linear system using tridiagonal Gaussian elimination is $O(N_c)$. Let N_δ be the size of the subgrid problem on a single coarse block. The total number of operations for a single time level is

$$(N_p + 2N_u)N_\delta\, N_c + O(N_c), \tag{4.1}$$

where N_p and N_u are the number of flops required to compute the subgrid pressure and acceleration unknowns at one point of the grid. Since the size of the coarse problem is small compared to the total size of the subgrid problems, the majority of the work is concentrated at the subgrid stage of the algorithm. However, the zero Neumann boundary conditions imposed on each coarse block ensure that the subgrid problems can be solved independently. In the following sections we develop the parallel algorithm and discuss accuracy and performance.

4.3. Parallel implementation. The parallel algorithm is based on two key features of the operator-based upscaling method. First, no communication is needed between the subgrid problems. (They are embarrassingly parallel.) Second, the size of the coarse problem is small compared to the size of the subgrid problems. We make use of these two facts and parallelize only the subgrid stage of the method (step 2(i) in the serial algorithm). The physical domain is decomposed into coarse blocks and split among the processors so that each processor is assigned an equal number of subgrid problems. The coarse-grid linear system is assembled and solved in serial. Although the coarse problem could also be parallelized, the cost savings do not appear to warrant the work. The system matrix is time independent and tridiagonal. Therefore, the linear system is cheap to assemble and fast to solve. The coarse-grid system matrix is the only global matrix stored on any single processor. The original velocity field $c(x, y)$ is used only at the subgrid stage. Therefore, only the portion of the earth model corresponding to a particular processor's subdomain is stored on that machine.

Parallelization of the algorithm reduces the cost of solving the subgrid problems by a factor p, with p the number of processors. Thus, the total cost (in number of operations) on a single time level is

$$\frac{(N_p + 2N_u)N_\delta\, N_c}{p} + O(N_c). \tag{4.2}$$

The only communication occurs during the transition between solving the subgrid problems and solving the coarse problem (and vice versa). After the subgrid problems are solved, we collect the data required to form the right-hand side vector for the coarse-grid linear system onto a single processor. Similarly, once the coarse problem is solved, the solution is broadcast to all processors in preparation for solving the subgrid problems at the next time level.

This parallelization strategy is decidedly different from standard spatial parallelism. In spatial or data parallelism, in order for each processor to compute the solution for its subdomain independently of the other processors, one must allocate additional arrays of memory (ghost cells) to store data from neighboring processors along the edges of the subdomain. Our approach avoids both allocation of additional memory and the passage of ghost cell information between neighboring processors.

4.4. Discussion of stability, dispersion, and convergence. In the paper by Arbogast [5] an error analysis of operator-based upscaling for elliptic problems

was given for the combined solution. In our case the combined solution consists of $\mathbf{U} \in \mathbf{V}_{H,h} = \mathbf{V}_H \oplus \delta\mathbf{V}_h$ and $P \in W_h$. We consider solving the system

$$(\mathbf{U}, \mathbf{v}) - (P, \nabla \cdot \mathbf{v}) = 0, \quad \mathbf{v} \in \mathbf{V}_{H,h}, \tag{4.3}$$

$$\left(\frac{1}{c^2}\frac{\partial P^2}{\partial t^2}, w\right) + (\nabla \cdot \mathbf{U}, w) = 0, \quad w \in W_h. \tag{4.4}$$

The augmented acceleration $\mathbf{U} = \mathbf{U}_H + \delta\mathbf{U}$ can be recovered easily from the coarse and subgrid solutions generated in the upscaling process. An alternative approach would be to study only the solution to the coarse problem (equation (3.12)). However, analysis of this problem is complicated by the fact that the acceleration and pressure solutions live on different grids (pressure on the fine scale only, upscaled acceleration on the coarse scale). In addition, the analysis of the coarse problem gives little insight into the stability of the method since this problem does not depend on time explicitly. The time derivative is associated with pressure and, therefore, appears only at the subgrid stage.

Stability analysis of the method is based on the principle of energy conservation [29], [12], [19], [20], which for system (4.3)–(4.4) can be written as follows:

$$\frac{d}{dt}\mathcal{E}(t) = \frac{d}{dt}\left(\left\|\frac{1}{c}\frac{\partial P}{\partial t}\right\|_{L^2}^2 + \|\mathbf{U}\|_{L^2}^2\right) = 0. \tag{4.5}$$

Rather than studying stability of the finite element equations (4.3)–(4.4), we turn our attention to a fully discrete difference scheme that corresponds to this method. The motivation for the discrete scheme is given in detail in a related paper by Korostyshevskaya and Minkoff [22]. We first establish the discrete equivalent of the conservation property (equation (4.5)):

$$E^{k-\frac{1}{2}} = E^{k+\frac{1}{2}}, \quad E_h^{k+\frac{1}{2}} = \frac{1}{2}\left\|c^{-1}\frac{P^{k+1}-P^k}{\Delta t}\right\|_\Delta^2 + \frac{1}{2}\left(\mathbf{U}^{k+1}, \mathbf{U}^k\right)_\Delta, \tag{4.6}$$

where $\|\cdot\|_\Delta$ is generated by the discrete inner product [29]. (The subscript Δ is omitted if there is no confusion with the standard L^2 norm.) Stability follows if positivity of the discrete energy functional can be established. To ensure positivity, we modify (4.6) using the polarization identity [12], [20]:

$$E_h^{k+\frac{1}{2}} = \frac{1}{2}\left\|c^{-1}\frac{P^{k+1}-P^k}{\Delta t}\right\|^2 + \frac{1}{2}\left\|\mathbf{U}^{k+\frac{1}{2}}\right\|^2 - \frac{\Delta t^2}{8}\left\|\frac{\mathbf{U}^{k+1}-\mathbf{U}^k}{\Delta t}\right\|^2, \tag{4.7}$$

where $\mathbf{U}^{k+\frac{1}{2}} = \frac{\mathbf{U}^{k+1}+\mathbf{U}^k}{2}$. The discrete equations from [22] and standard algebraic inequalities used to estimate the last term in (4.7) allow us to obtain a lower bound on the discrete energy:

$$E_h^{k+\frac{1}{2}} \geq \frac{1}{2}\left(1 - c_0^2\Delta t^2\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}\right)\right)\left\|c^{-1}\frac{P^{k+1}-P^k}{\Delta t}\right\|^2 + \frac{1}{2}\left\|U^{k+\frac{1}{2}}\right\|^2.$$

Here we have denoted the length of a single fine cell in the x- and y-directions by Δx and Δy, respectively. The fine-grid time step is denoted by Δt, and c_0 is a lower bound on the sound velocity.

We observe that if

$$c_0^2 \Delta t^2 \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right) < 1, \tag{4.8}$$

then the discrete energy is positive. This implies that temporal iterates are bounded by the initial data:

$$\left(1 - c_0^2 \Delta t^2 \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right)\right) \left\| c^{-1} \frac{P^{k+1} - P^k}{\Delta t} \right\|^2 + \left\| U^{k+\frac{1}{2}} \right\|^2$$
$$\leq E_h^{k+\frac{1}{2}} = \cdots = E_h^{\frac{1}{2}} \leq \left\| c^{-1} \frac{P^1 - P^0}{\Delta t} \right\|^2 + \left\| U^{\frac{1}{2}} \right\|^2.$$

Therefore, the discrete scheme is stable. (See Korostyshevskaya and Minkoff [22] and Vdovina and Minkoff [31] for a complete convergence analysis of the upscaling method in the context of the wave equation.)

The stability condition (equation (4.8)) is exactly the standard CFL condition that relates the time step to the parameter field and space step. The importance of this result for upscaling follows from the fact that condition (4.8) gives a restriction on the time step in terms of the parameter field and subgrid step only. It does not depend on the coarse grid. This result confirms our intuitive expectations since both the sound velocity field and time derivative explicitly appear only at the subgrid stage. Note that since we have not chosen to upscale both pressure and acceleration, we pay for this choice in the size of time step allowed. If pressure were also upscaled, it is possible that a considerably bigger time step could be taken.

To prove convergence, one also must show that the upscaling scheme is accurate (consistency). The discussion of accuracy is given in detail in the companion paper by Korostyshevskaya and Minkoff [22].

We study the dispersion properties of the upscaling algorithm numerically by running the program with a homogeneous background velocity of 2500 m/s. Our goal is to determine the number of fine- and coarse-grid nodes required per wavelength to minimize the effect of numerical dispersion. We consider three numerical experiments. In all experiments we assume a square domain of size 36 km $\times$ 36 km discretized into 3600 $\times$ 3600 fine grid blocks and use a Gaussian source located at the center of the domain. (See section 5 for the functional form and graph of the source.) The source parameter σ controls the source wavelength. We choose the time step so that the stability condition (equation (4.8)) is close to optimal and run the simulations for 800 time steps. The standard second-order in space and time numerical scheme requires at least 10 grid points per wavelength [12]. Using this result as a starting point, we assume 100 fine-grid and 10 coarse-grid nodes per wavelength in our first experiment. We control the number of coarse- and fine-grid nodes per wavelength by varying the source parameter σ and the coarse spatial step. The fine spatial step remains fixed (10 m) for all the experiments. In the first dispersion experiment, σ equals 1000.

Figure 4 shows the upscaled horizontal acceleration for this experiment. In order to highlight the small features of the amplitude, we plot the solution on the scale of $1 \cdot 10^{-10}$. We can see that solution shows no sign of dispersion. In our second experiment, we violate the dispersion condition on the fine grid by modifying the source parameter so that we have only 5 fine grid nodes per wavelength ($\sigma = 50$ in this experiment). As expected, the solution in Figure 5 is corrupted by grid dispersion. The

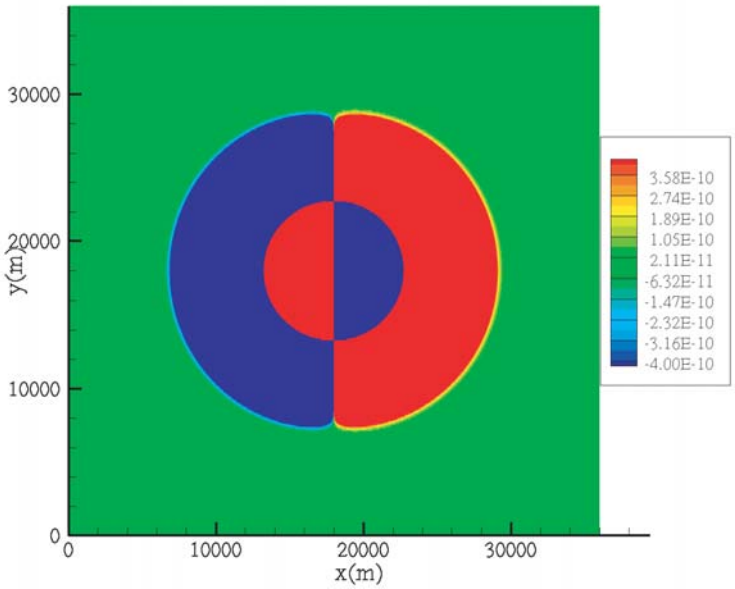


FIG. 4. *The upscaled horizontal acceleration for the first dispersion experiment. The numerical grid contains* 3600×3600 *fine grid blocks and* 360×360 *coarse grid blocks. Source parameter* σ *equals* 1000. *We use* 100 *fine-grid and* 10 *coarse-grid points per wavelength and take* 800 *total time steps.*

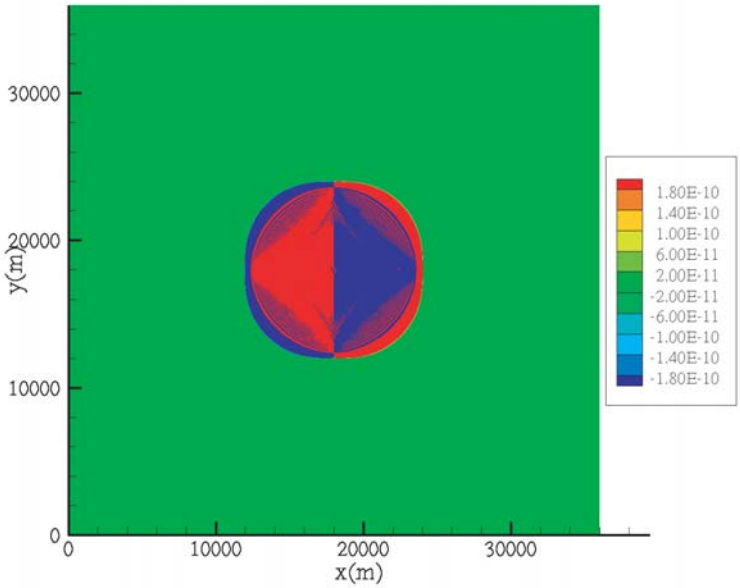


FIG. 5. *The upscaled horizontal acceleration for the second dispersion experiment. The numerical grid contains* 3600×3600 *fine grid blocks and* 900×900 *coarse grid blocks. The source parameter* σ *equals* 50. *We used* 5 *fine-grid and* 1.25 *coarse-grid points per wavelength and took* 800 *total time steps.*

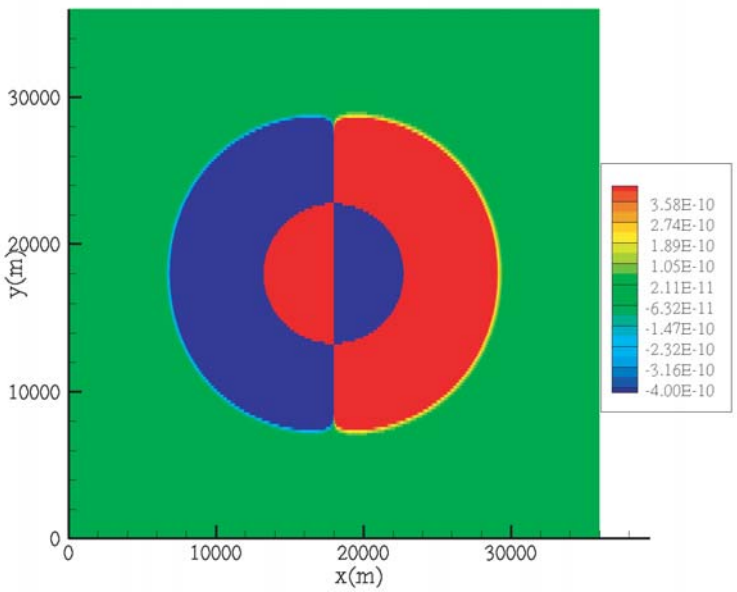


FIG. 6. *The upscaled horizontal acceleration for the third dispersion experiment. The numerical grid contains* 3600×3600 *fine grid blocks and* 180×180 *coarse grid blocks. Source parameter* σ *equals* 1000. *We used* 100 *fine-grid and* 5 *coarse-grid points per wavelength and took* 800 *total time steps.*

amplitude spikes in this figure correspond to the oscillations generated by parasitic waves. Note, however, that whenever we violate the dispersion condition on the fine grid, it is automatically broken on the coarse grid. Therefore, this experiment does not explain which grid causes the dispersion. To clarify this point, we ensure that the dispersion condition is satisfied on the fine grid and violated only on the coarse grid. In the third dispersion experiment, we take 100 fine-grid points and 5 coarse-grid points per wavelength. Figure 6 shows the solution for the third dispersion experiment. We see no sign of dispersion in this experiment. We conclude that (1) we need 10 fine-grid nodes per wavelength to minimize the effect of dispersion, and (2) numerical dispersion seems to be independent of the number of coarse-grid points chosen per wavelength.

5. Numerical experiments. In this section we discuss accuracy of the method and performance of the parallel algorithm. In section 5.1 we compare upscaled solutions to full finite-difference solutions of the wave equation for three different velocity fields. In experiments 1 and 2, the wave propagates through a medium with two velocities that alternate in a periodic checkerboard. Periodic media are standard input fields for homogenization studies, so they seemed worthy of study with our method. Section 5.2 contains timing results and speedup plots for the checkerboard experiment. The velocity field used in experiment 3 is a finely layered medium (a common geophysical model that explains subsurface regions such as the Gulf of Mexico). Finally, experiment 4 uses an input velocity field which is a single realization taken from a stochastic von Karman distribution of a two-component material mixture [15], [14]. This example in particular is of interest for one of our target applications—deep

crustal imaging. As discussed in [16], the lower crust contains interesting historical information about the earth's geology. However, this part of the subsurface is deep enough that it is not accessible by conventional means (i.e., seismic near offset and wide angle reflection profiles). Deep crustal data is characterized by short, discontinuous reflections which are best described stochastically rather than deterministically. Huge amounts of data are needed to do accurate modeling to these depths, and exact targets are not imaged reliably enough for deterministic modeling.

While flow in porous media parameters such as permeability can vary over a few orders of magnitude [11], seismic parameters such as velocity and density are much less variable. For instance, Dobrin and Savit [13] give a table which lists compressional and shear wave velocities in typical subsurface materials (rocks). These velocities are based on laboratory measurements of rock samples taken from various locations and include granite, basalt, sandstone, limestone, clay, and loose sand. Compressional wave velocities for these samples range from about 1500 m/s (clay, loose sand, and limestone) to 6000–8000 m/s for dunite, basalt, and granite. As stated in [13], loosely consolidated rock types have velocities which depend quite a bit on the size of the pore space and on the fluids filling the pores. (See Minkoff et al. [24] for an interesting case study of how mechanical changes in rock materials which resulted in changes in the pore space and fluid composition filling this space impact seismic wave velocities.)

One might suppose that changes in velocity due to rock depth would be significant. However, Dobrin and Savit [13] state that a pressure change from one atmosphere to 10000 bars (corresponding to 40000 m depth) resulted in only a 15% increase in compressional wave velocity for granite. Although sedimentary rocks such as sandstone may show more of a change due to consolidation, the range of velocity values stated earlier (1500–8000 m/s) still brackets the wave velocities of most subsurface rocks.

Nonetheless, being able to solve seismic modeling problems on a coarser grid is of much interest. For one thing, the mathematical tool of inversion (used for oil and gas exploration, near surface cleanup, and deep crustal imaging) requires repeated solution of the wave equation (forward problem). Further, realistic seismic modeling in three dimensions commonly employs terabytes of data [25], [26]. So although the range of values for the parameters of interest in wave propagation is narrower than for flow in porous media, the need for an improvement in speed and memory still exists.

The experiments discussed in this paper were chosen primarily for two reasons:

- They contain subwavelength-scale heterogeneities (heterogeneities that are less than a single coarse grid block).
- They contain simple features which may be components of subsurface regions such as repeated structures, thin layers of varying sediments, and heterogeneity characterized by statistical distribution.

All the numerical experiments described in this section were performed on a heterogeneous Linux cluster with 12 processors (6 dual-processor machines). Eight of the processors run at a maximum speed of 2 GHz with 1 GB of memory. The other four nodes use 3.06 GHz Intel chips and have 2 GB of RAM per processor. The dual-processor machines are connected by 100 Mbits/sec ethernet cables. For simplicity we distributed the work uniformly among the processors. (We did not load balance to make full use of the four faster processors, since these were only acquired recently.)

5.1. Accuracy. To demonstrate the accuracy of the upscaling method, we compare the upscaled solution to the solution generated by a standard second-order in space and time, staggered, finite-difference scheme applied to the full fine-grid problem. For the variable-velocity experiments shown in this paper, no analytic solutions

exist. However, standard finite-difference schemes which satisfy dispersion and CFL conditions give well-accepted convergent solutions and form the basis for our comparison here. These full finite-difference solutions are exactly the correct comparison to make for an upscaling algorithm applied to realistic problems. The upscaled solution shown in this section is the augmented or reconstructed solution defined on the fine grid and given by the following formula:

$$\mathbf{U} = \mathbf{U}_H + \delta\mathbf{U},$$

where $\mathbf{U}_H$ and $\delta\mathbf{U}$ are the coarse and subgrid components. We make use of the fact that the subgrid component $\delta\mathbf{U}$ is automatically generated in the upscaling process and can be used to compute the reconstructed solution after the time-step loop is complete. (In other words, it requires no more than a single postprocessing *gather* at the end of the parallel algorithm.) In all the numerical experiments we show the reconstructed (upscaled) solution. In the most realistic experiment (experiment 4) we also show the coarse-scale solution ($\mathbf{U}_H$ alone).

In all the experiments, we used a Gaussian source function described by

$$f(x,y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-x_0)^2 + (y-y_0)^2}{2\sigma^2}\right),$$

where σ controls the wavelength. The source is centered at grid location (x_0, y_0), which for these experiments is chosen to be the center of the domain (see Figure 7).

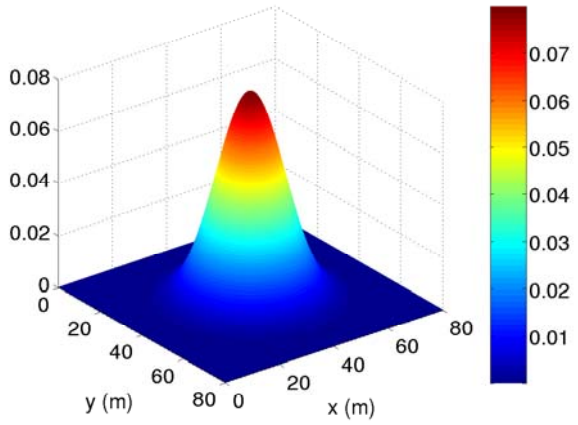
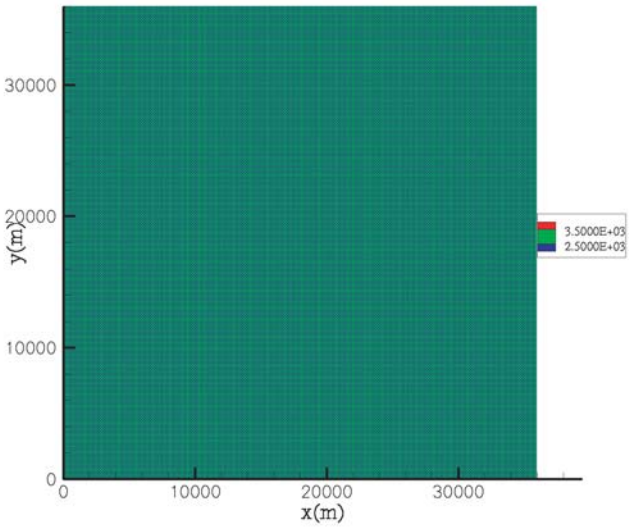


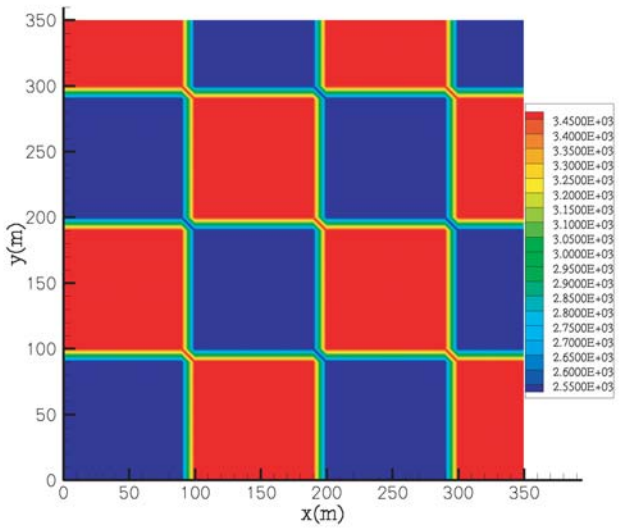
FIG. 7. *Gaussian source with* $\sigma = 10$.

For experiments 1 and 2 we assume a square domain of size 36 km by 36 km. The domain is discretized into 3600×3600 fine grid blocks. The coarse grid consists of 180×180 blocks in experiment 1 and 90×90 blocks in experiment 2. Both simulations ran for 400 time steps. Source parameter σ equals 4000.

The sound velocity in experiments 1 and 2 changes periodically on the fine scale. The earth model consists of a checkerboard medium with cells having velocities that alternate between 2500 m/s and 3500 m/s (see Figure 8). The period (the size of each checkerboard cell) is 10×10 fine grid blocks. Note that in experiment 1 each coarse block consists of 20×20 fine blocks, and in experiment 2 each coarse block contains 40×40 fine grid blocks. Therefore, each coarse block is two times larger than the



(a) Earth model for experiments 1 and 2.



(b) A subset (1/100) of the earth model for experiments 1 and 2.

FIG. 8. *Each checkerboard cell is size* 10×10 *fine grid blocks. Each cell has sound velocity of either* 2500 *m/s or* 3500 *m/s. The domain is discretized into* 3600×3600 *fine grid blocks. In experiment* 1 *the coarse grid consists of* 180×180 *blocks with* 20×20 *fine blocks in each coarse block. In experiment* 2 *the coarse grid has* 90×90 *blocks with* 40×40 *fine grid blocks in each coarse block.*

fine-scale structural period in experiment 1 and four times larger than the period in experiment 2. Figures 9, 10, and 11 show top views of the full finite-difference and upscaled solutions for experiment 1. Figure 12 shows the full finite-difference and upscaled vertical accelerations for experiment 2.

We see from the pictures that the upscaled solution captures the fine-scale periodic structure of the medium. The horizontal and vertical accelerations have regions of low and high amplitude that match cells of low and high velocities in Figure 8. Comparing the vertical accelerations in experiments 1 and 2, we see that as we increase the size of the coarse blocks, the upscaled solution loses some of the fine-scale details internal to the main feature, but the results are still relatively good.

For experiments 3 and 4 we consider a square domain of size 10 km by 10 km. The numerical grids contain 1000×1000 fine blocks and 100×100 coarse blocks. We ran the simulation for 200 time steps in experiment 3 and 350 time steps in experiment 4. We take $\sigma = 1000$ in both experiments.

The finely layered velocity field for experiment 3 contains velocity strips ranging from 3000 m/s to 7500 m/s (see Figure 13). The values increase monotonically from the top of the domain to the bottom, except for two high velocity strips of 7500 m/s located at 5250 m and 5750 m in y. Figures 14, 15, and 16 compare the full finite-difference solution and the upscaled solution for this experiment. Since the domain is homogeneous in the x-direction, the horizontal acceleration is not affected by the strips of varying sound velocity. Figure 16 shows that vertical acceleration captures the fine-scale fluctuations of sound velocity. We see the regions of high amplitude generated by velocity strips located between 3500 m and 7000 m in y.

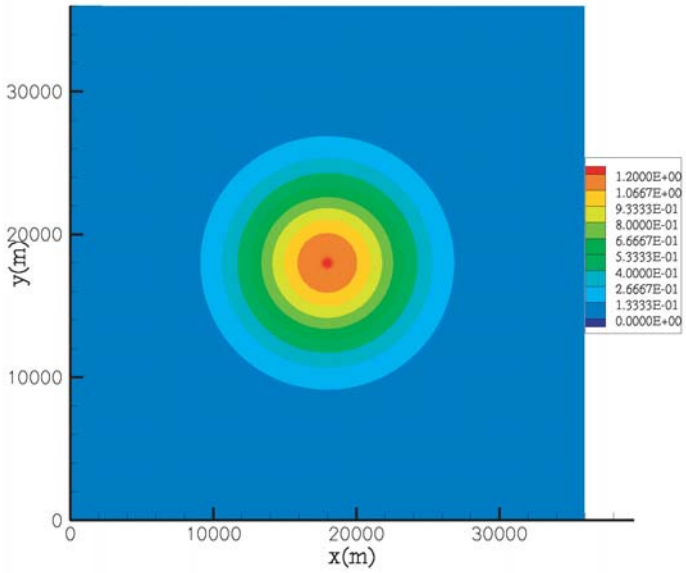
The velocity field for experiment 4 (see Figure 17) is a single realization from a stochastic distribution which models a two-component mixture of materials that vary based on input correlation lengths in x and y, a roughness parameter (or Hurst number), and a percentage distribution of the two materials. For this experiment, the materials are distributed evenly (a 50-50 mixture), and the correlation lengths were chosen to be 40 m for each material in each direction. This correlation length guaranteed fine-scale heterogeneity below the coarse grid block size of 100 m $\times$ 100 m. The two materials have velocity values of 3500 m/s and 4500 m/s.

Figures 18, 19, 20, and 21 show the full-finite difference and upscaled solutions for experiment 4. In this experiment, in addition to the reconstructed solution defined on the fine grid, we show the coarse component of acceleration (Figures 19 and 21). We see from the pictures that the upscaled solution appears to be in a good agreement with the full finite-difference solution. The reconstructed solution resolves the fine-scale information very accurately. The coarse components of the horizontal and vertical accelerations shown in Figures 19 and 21 lose some fine-scale details but still do an amazingly good job of capturing the essential structure of the earth model.

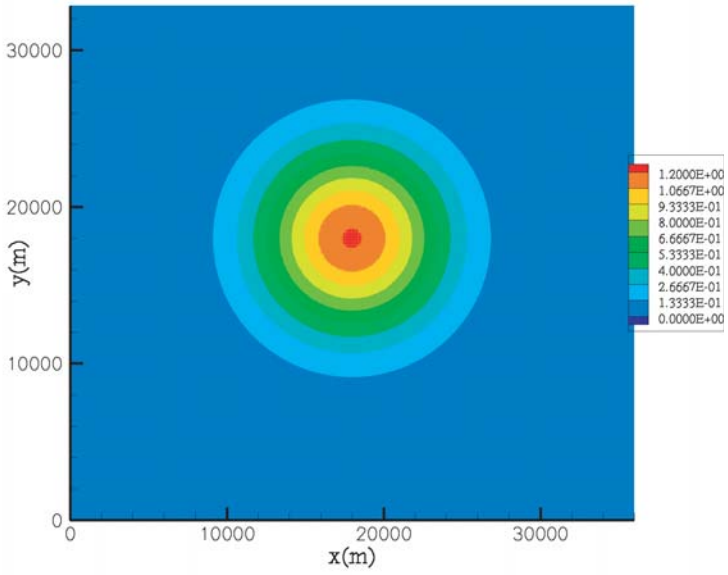
5.2. Speedup and efficiency. To understand the parallel performance of the algorithm, we analyze speedup. Speedup is a measure of the length of time it takes to run a code on p processors versus a single processor:

$$S(n,p) = \frac{T_1(n)}{T_p(n)},$$

where $T_1(n)$ and $T_p(n)$ are the runtimes on 1 and p processors, respectively, and n denotes the size of the problem. The program is said to have linear or optimal speedup if $S(n,p) = p$ [27].

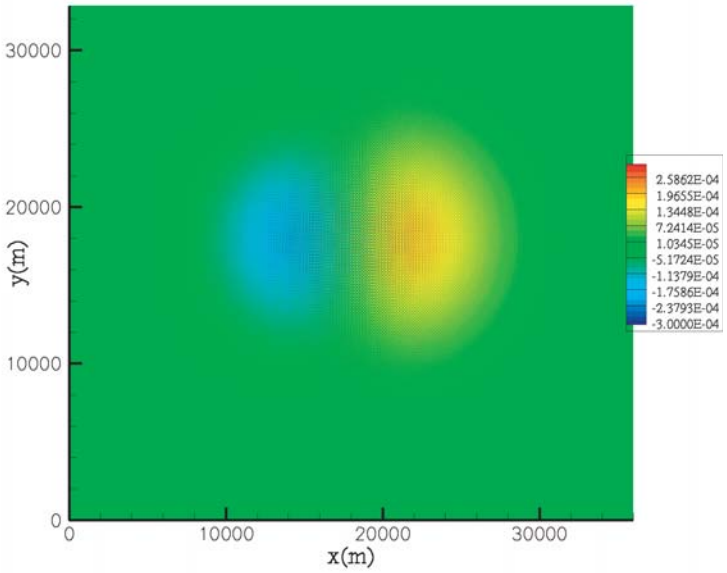


(a) Finite-difference pressure.

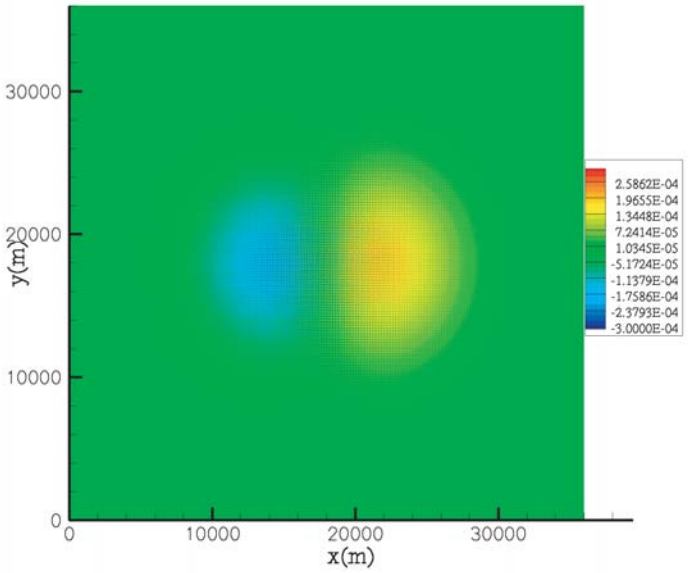


(b) Subgrid pressure.

FIG. 9. *Comparison of the finite-difference solution (top) and the upscaled solution (bottom) for experiment* 1. *The numerical grids contain* 3600×3600 *fine grid blocks and* 180×180 *coarse grid blocks. The velocity field is shown in Figure* 8.

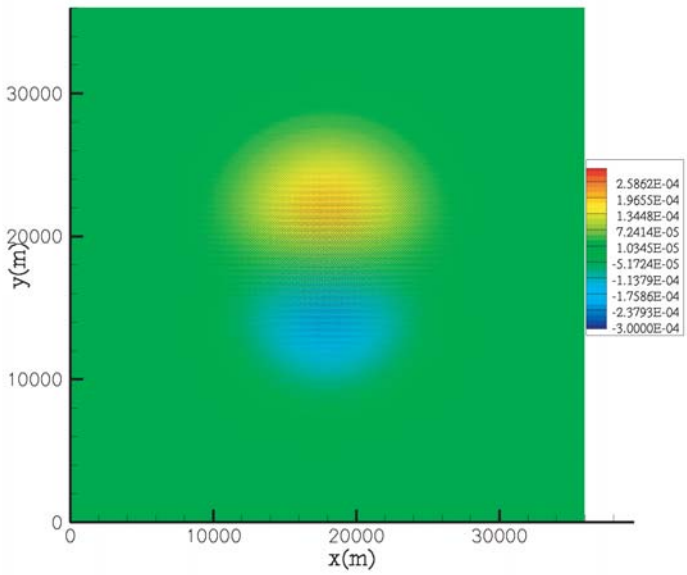


(a) Finite-difference horizontal acceleration.

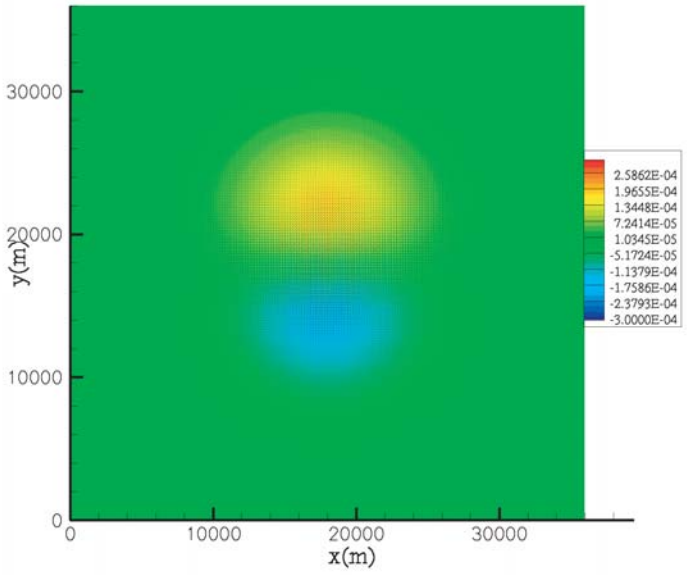


(b) Upscaled horizontal acceleration.

FIG. 10. *Comparison of the finite-difference solution (top) and the upscaled solution (bottom) for experiment* 1. *The numerical grids contain* 3600×3600 *fine grid blocks and* 180×180 *coarse grid blocks. The velocity field is shown in Figure* 8.

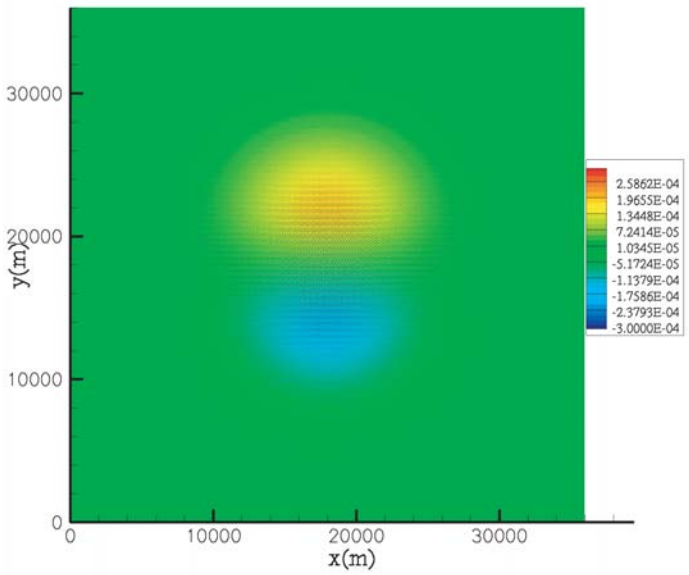


(a) Finite-difference vertical acceleration.

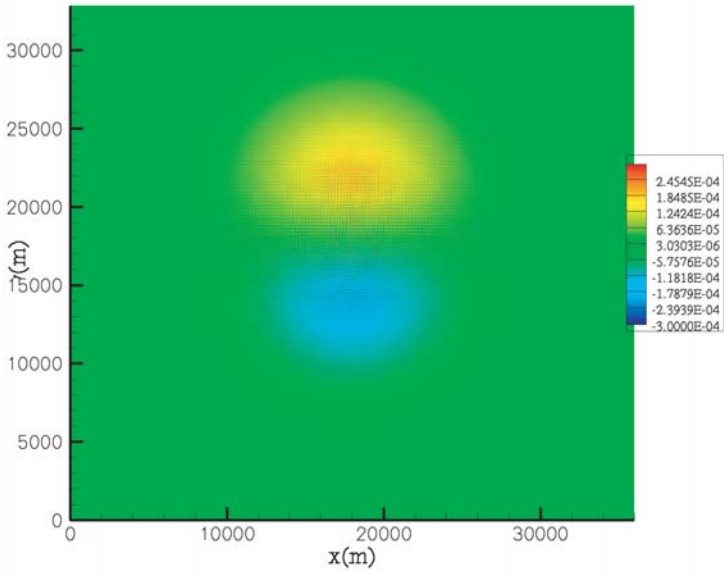


(b) Upscaled vertical acceleration.

FIG. 11. *Comparison of the finite-difference solution (top) and the upscaled solution (bottom) for experiment* 1. *The numerical grids contain* 3600×3600 *fine grid blocks and* 180×180 *coarse grid blocks. The velocity field is shown in Figure* 8.



(a) Finite-difference vertical acceleration.



(b) Upscaled vertical acceleration.

FIG. 12. *Comparison of the finite-difference solution (top) and the upscaled solution (bottom) for experiment* 2. *The numerical grids contain* 3600×3600 *fine grid blocks and* 90×90 *coarse grid blocks. The velocity field is shown in Figure* 8.

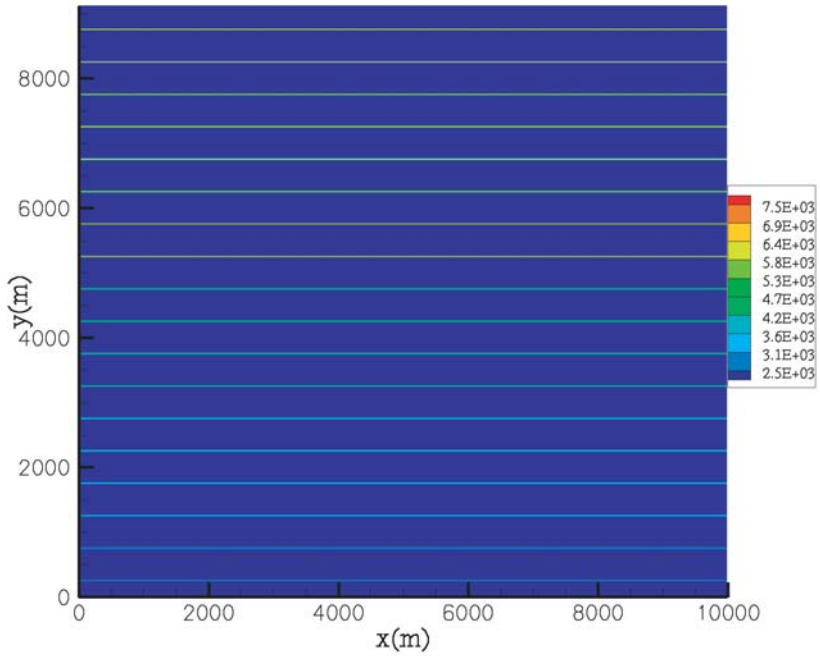
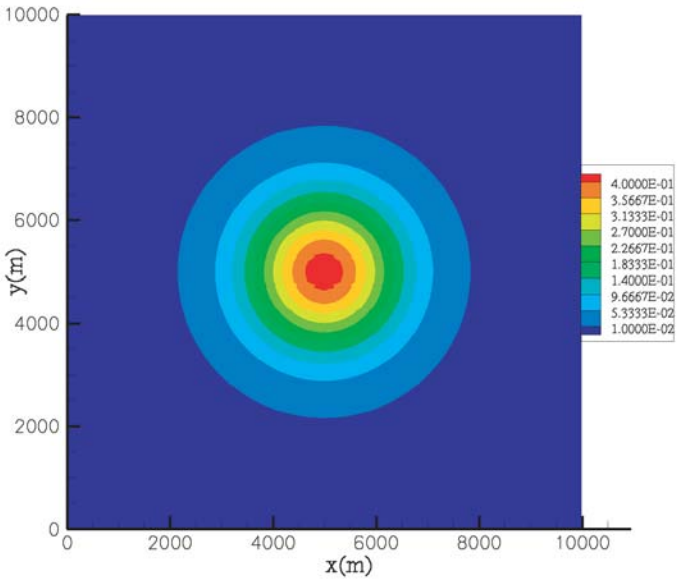


FIG. 13. *Earth model for experiment* 3. *Sound velocity is given in units of m/s. Each strip is two fine grid blocks wide. The domain is discretized into* 100×100 *coarse blocks with* 10×10 *fine blocks in each coarse block.*

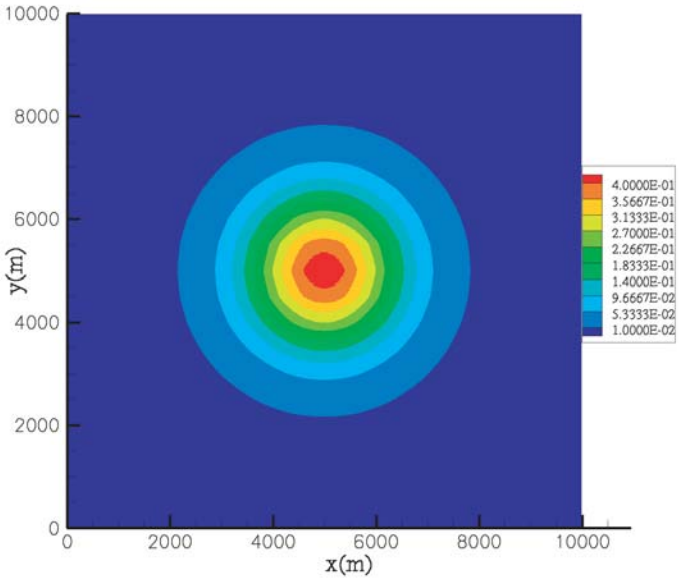
Since the subgrid problems are independent from one coarse block to the next, we expect to see optimal speedup in the parallelization of the subgrid stage. The coarse problem is solved serially, so we do not expect to achieve optimal speedup for the algorithm as a whole.

Tables 1 and 2 and Figure 22 summarize the observed timing results (in seconds) for a nonhomogeneous checkerboard medium of size 360 m $\times$ 360 m discretized into 3600×3600 fine grid blocks. Figure 22 shows the speedup plots for different sizes of coarse blocks and different numbers of processors. Table 2 contains the observed timing results corresponding to the speedup plots in Figure 22. In addition to the timing results for the upscaling code, Table 2 shows the time required to solve the problem by a standard serial full finite-difference code. In both the second column of Table 1 and in all columns of Table 2 "total time" refers to the time taken within the time-step loop to solve both the subgrid and coarse problems. Table 1 also contains the time required to solve the subgrid and coarse problems individually and the time taken for postprocessing. Postprocessing refers to the communication of data between the subgrid problems and the coarse problem and the updating of subgrid variables at the end of each time step.

As expected, the third column of Table 1 demonstrates that the time required to solve the subgrid problems is reduced by a factor of 2 when we double the number of processors. The fourth column in the table indicates that the time required to solve the coarse problem is significantly less than the time required to solve the subgrid problems.

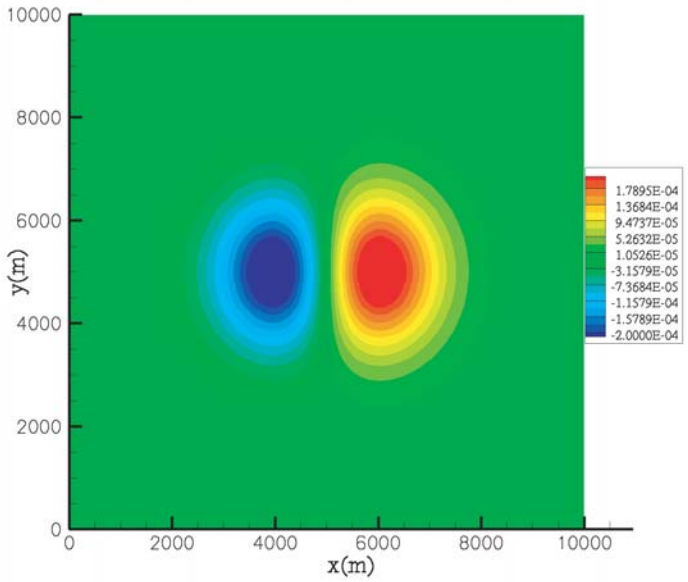


(a) Finite-difference pressure.

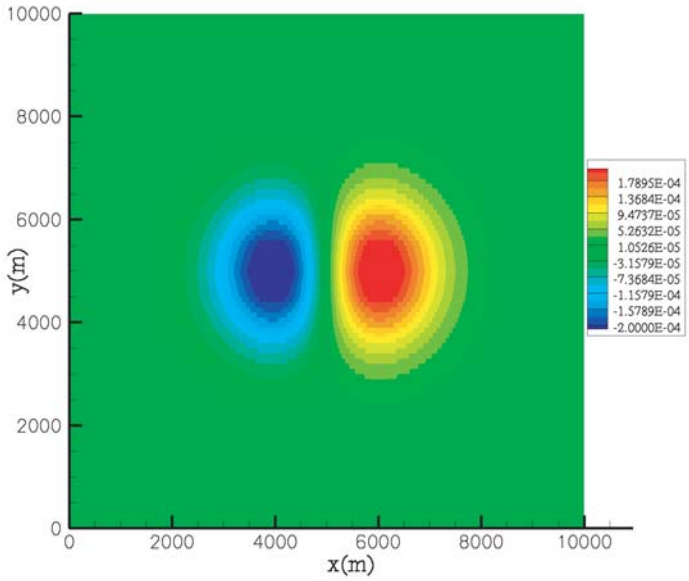


(b) Subgrid pressure.

FIG. 14. *Comparison of the finite-difference solution (top) and the upscaled solution (bottom) for experiment* 3. *The numerical grids contain* 1000×1000 *fine grid blocks and* 100×100 *coarse grid blocks. The velocity field is shown in Figure* 13.

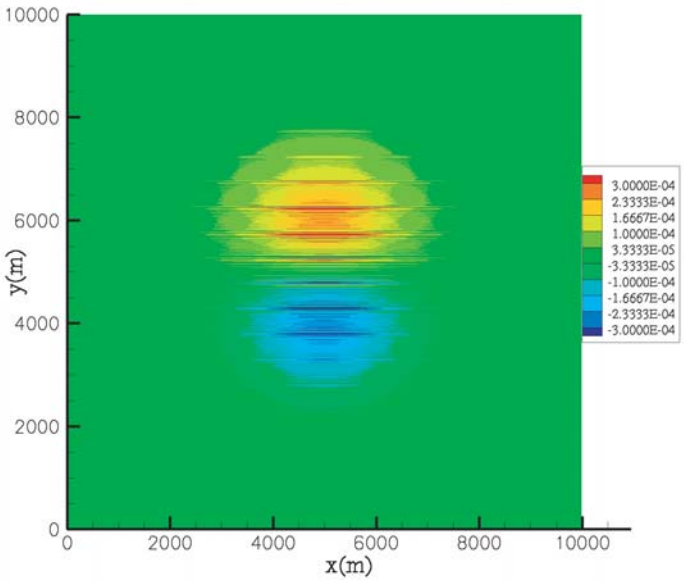


(a) Finite-difference horizontal acceleration.

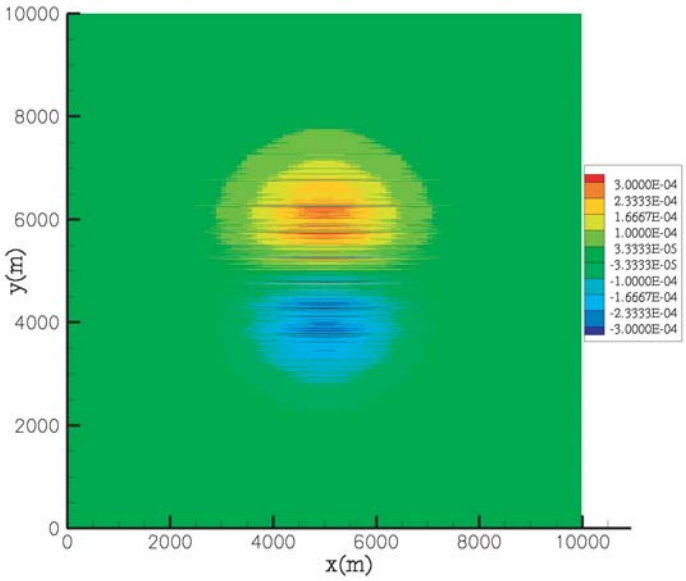


(b) Upscaled horizontal acceleration.

FIG. 15. *Comparison of the finite-difference solution (top) and the upscaled solution (bottom) for experiment* 3. *The numerical grids contain* 1000×1000 *fine grid blocks and* 100×100 *coarse grid blocks. The velocity field is shown in Figure* 13.



(a) Finite-difference vertical acceleration.



(b) Upscaled vertical acceleration.

FIG. 16. *Comparison of the finite-difference solution (top) and the upscaled solution (bottom) for experiment* 3. *The numerical grids contain* 1000×1000 *fine grid blocks and* 100×100 *coarse grid blocks. The velocity field is shown in Figure* 13.

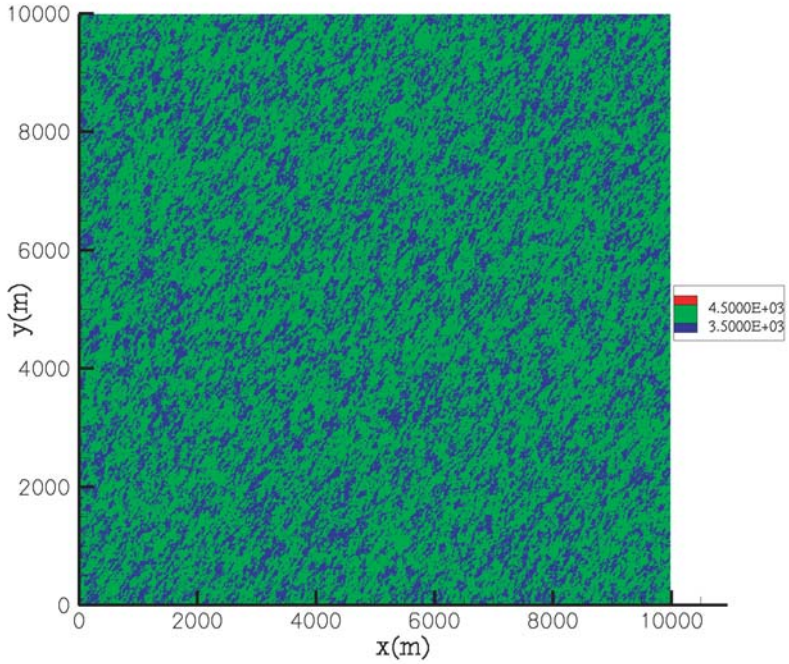
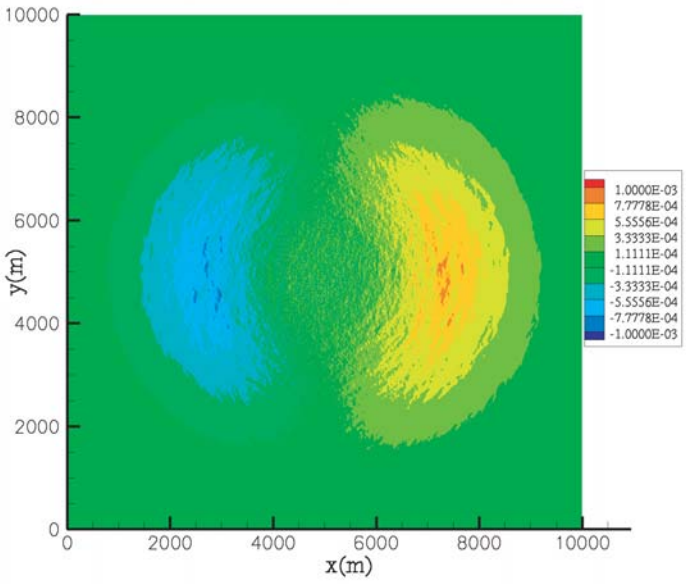


FIG. 17. *Earth model for experiment* 4. *Sound velocity is given in units of m/s. The domain is discretized into* 100×100 *coarse blocks with* 10×10 *fine blocks in each coarse block.*

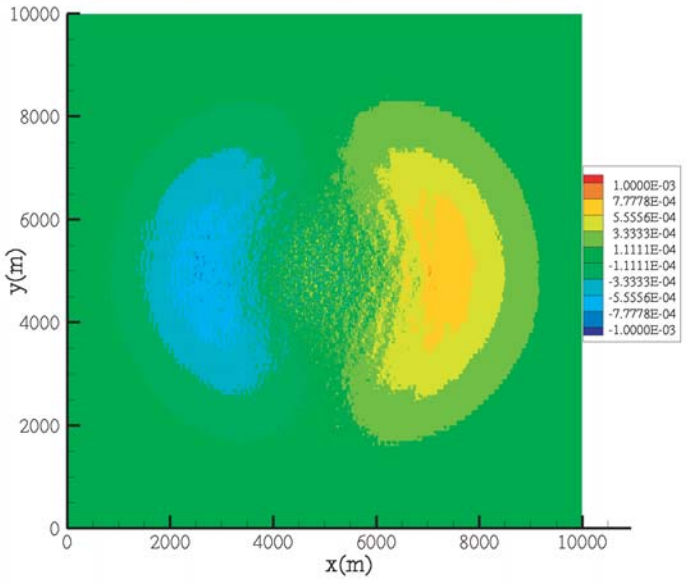
We see from Table 2 that, as expected due to the complexity of the upscaling algorithm, on a single processor the finite-difference algorithm is faster than operator upscaling. On multiple processors, however, the parallel upscaling algorithm shows much better performance. More importantly, no communication is required, so the method would appear to be faster than standard data parallelism (although we did not have the ability to make that comparison at this time). Table 2 and Figure 22 demonstrate that the algorithm achieves close to optimal speedup for up to 6 processors. We observe a reduction in speedup from 6 to 12 processors, as expected, since postprocessing becomes prohibitive for this size problem as the number of processors increases. We also see that the speedup depends on the size of the coarse blocks. The speedup improves as the size of the coarse block increases.

6. Conclusions. Modeling of wave propagation in a heterogeneous medium often involves many different spatial scales. Operator-based upscaling uses a two-scale decomposition of the unknowns to capture local information about the solution while solving the full problem only on the coarse grid. We have adapted the two-stage operator-based upscaling method to the constant density, variable sound velocity acoustic wave equation. At the first stage of the method we solve for the subgrid information internal to each coarse block using staggered finite differences. The subgrid solutions from the first stage are used to augment a new coarse-grid problem which is solved at the second stage.

The main difference between operator-based upscaling and other upscaling techniques is that operator-based upscaling uses the original input parameter field in both

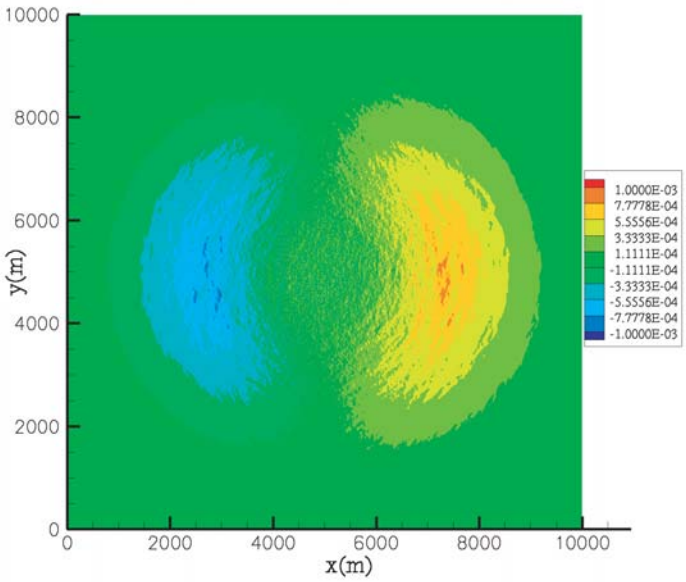


(a) Finite-difference horizontal acceleration.

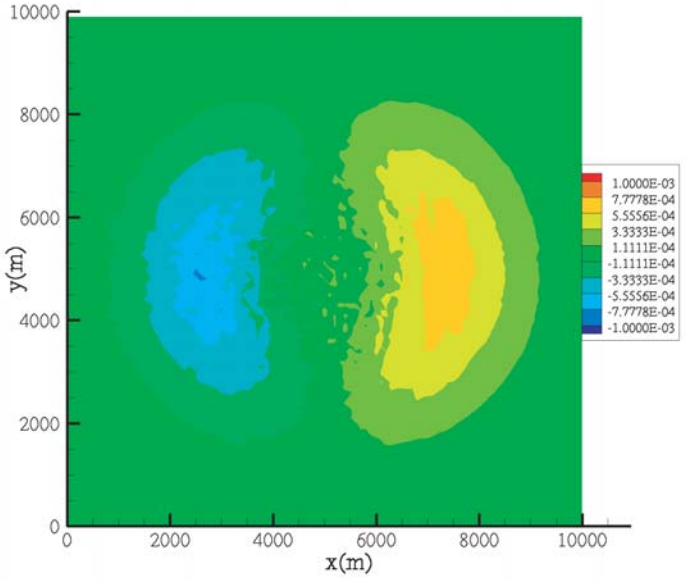


(b) Upscaled horizontal acceleration.

FIG. 18. *Comparison of the finite-difference solution (top) and the upscaled solution (bottom) for experiment* 4. *The numerical grids contain* 1000×1000 *fine grid blocks and* 100×100 *coarse grid blocks. The velocity field is shown in Figure* 17.

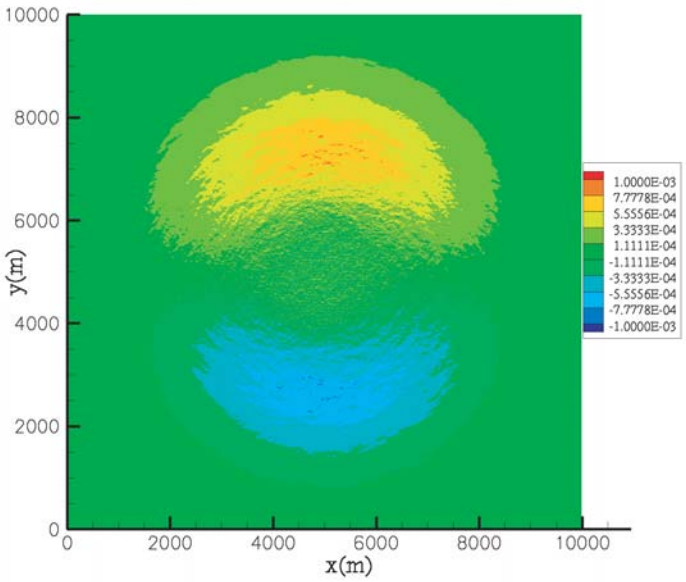


(a) Finite-difference horizontal acceleration.

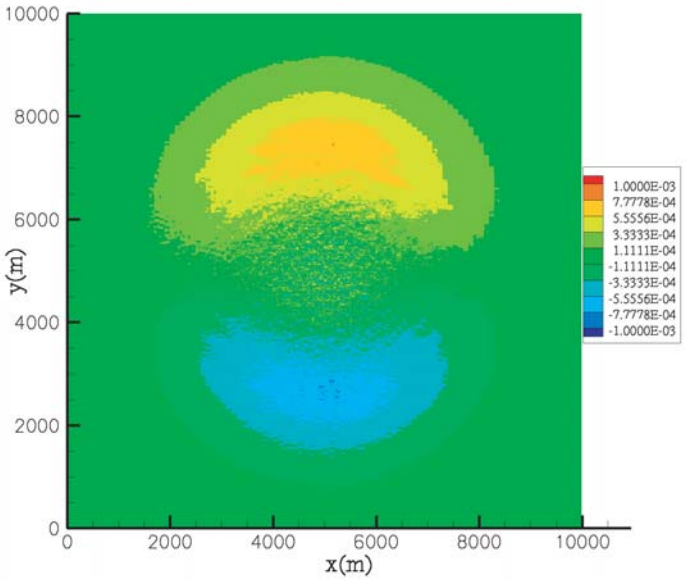


(b) Coarse component of horizontal acceleration.

FIG. 19. *Comparison of the finite-difference solution (top) and the coarse component of the upscaled solution (bottom) for experiment* 4. *The numerical grids contain* 1000×1000 *fine grid blocks and* 100×100 *coarse grid blocks. The velocity field is shown in Figure* 17.

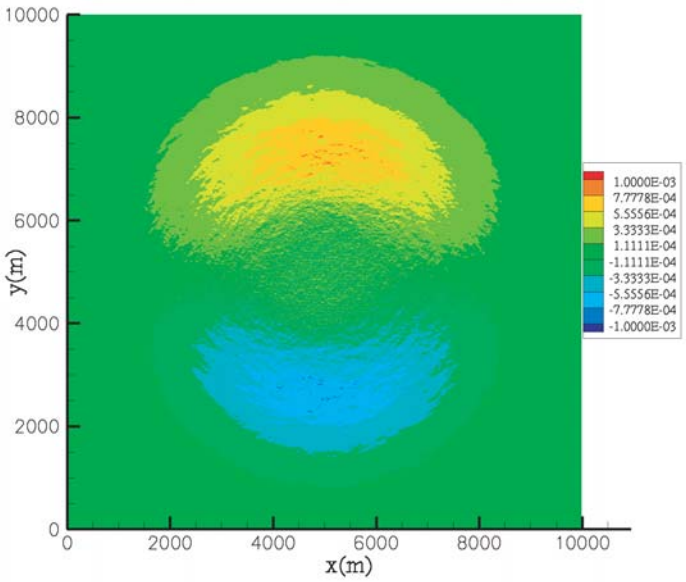


(a) Finite-difference vertical acceleration.

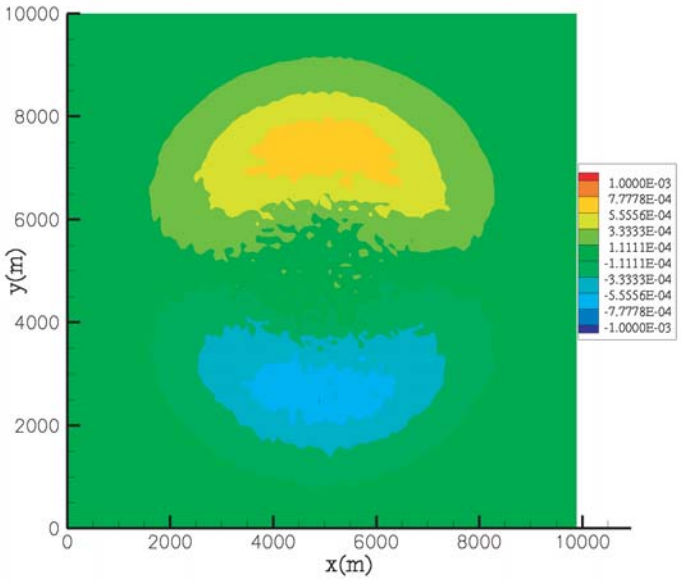


(b) Upscaled vertical acceleration.

FIG. 20. *Comparison of the finite-difference solution (top) and the upscaled solution (bottom) for experiment* 4. *The numerical grids contain* 1000×1000 *fine grid blocks and* 100×100 *coarse grid blocks. The velocity field is shown in Figure* 17.



(a) Finite-difference vertical acceleration.



(b) Coarse component of vertical acceleration.

FIG. 21. *Comparison of the finite-difference solution (top) and the coarse component of the upscaled solution (bottom) for experiment* 4. *The numerical grids contain* 1000×1000 *fine grid blocks and* 100×100 *coarse grid blocks. The velocity field is shown in Figure* 17.

TABLE 1

Observed timing (in seconds) for different numbers of processors. The numerical grids contain 3600×3600 *fine grid blocks and* 36×36 *coarse grid blocks. Twenty time steps were taken in these experiments. The total time (column* 2*) is the sum of columns* 3–5.

Number of processors	Total time	Subgrid problems	Coarse problem	Post-processing
1	67.01	61.36	0.007724	5.64
2	34.20	31.24	0.007438	2.83
4	17.19	15.60	0.007290	1.43
6	11.57	10.49	0.007320	0.95
8	10.69	9.01	0.006828	1.51
12	7.39	5.96	0.007347	1.00

TABLE 2

Observed time (in seconds) taken by the full finite-difference and upscaling codes (time-step loop) for different numbers of processors and different groupings of fine grid blocks. Twenty time steps were taken in these experiments. The fine numerical grid contains 3600×3600 *grid blocks.*

Number of processors	Finite-difference code	Upscaling code number of fine blocks per coarse block			
		100×100	60×60	50×50	40×40
1	40.27	67.01	68.34	66.88	68.46
2	–	34.20	34.07	34.19	34.73
4	–	17.19	17.43	17.66	18.19
6	–	11.57	11.96	12.16	12.72
8	–	10.69	10.96	11.99	12.40
12	–	7.39	8.11	8.97	11.50

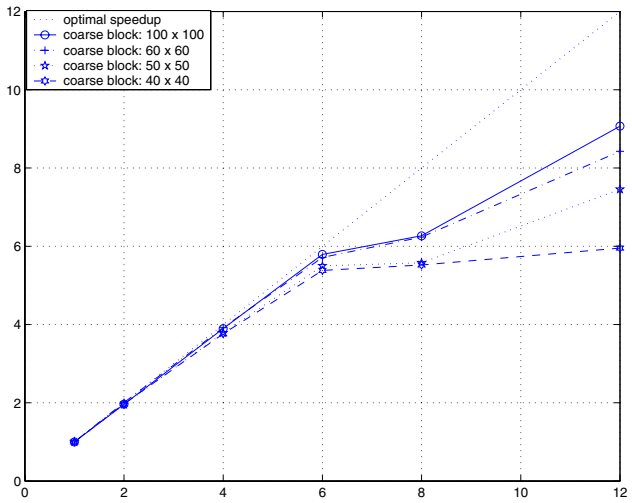


FIG. 22. *Plot of observed speedup. The fine numerical grid contains* 3600×3600 *grid blocks.*

stages of the algorithm. The output of the method is not the calculation of effective parameters but rather an *effective solution*. Since this method makes no assumptions about physical properties of the model, it can be applied to an extremely large class of problems. Parallelization of the subgrid stage of the method reduces the cost of solving the problem and allows us to solve significantly larger problems in less time. The main advantage of the parallel version of operator upscaling over standard data parallel strategies is that no communication is required between neighboring processors.

Timing studies show that the parallel algorithm has near optimal speedup. Parallel efficiency improves as more fine grid blocks are added to each coarse block. Four numerical experiments with three variable velocity earth models show excellent agreement between the upscaled and full finite-difference solutions. The upscaled solution captures the fine-scale fluctuations of the sound velocity extremely accurately.

Acknowledgments. We thank Bill Symes and Alan Levander of Rice University and Christian Poppeliers (formerly of Rice University, currently at San Angelo State) for helpful discussions and suggestions that enhanced this work.

REFERENCES

[1] G. Allaire, *Homogenization and two-scale convergence*, SIAM J. Math. Anal., 23 (1992), pp. 1482–1518.
[2] T. Arbogast, *Numerical Subgrid Upscaling of Two-Phase Flow in Porous Media*, Lecture Notes in Phys. 552, Springer-Verlag, Berlin, 2000.
[3] T. Arbogast, *Implementation of a locally conservative numerical subgrid upscaling scheme for two-phase flow*, Comput. Geosci., 6 (2002), pp. 453–481.
[4] T. Arbogast, *An overview of subgrid upscaling for elliptic problems in mixed form*, in Current Trends in Scientific Computing, Z. Chen, R. Glowinski, and K. Li, eds., AMS, Providence, RI, 2003, pp. 21–32.
[5] T. Arbogast, *Analysis of a two-scale, locally conservative subgrid upscaling for elliptic problems*, SIAM J. Numer. Anal., 42 (2004), pp. 576–598.
[6] T. Arbogast and S. Bryant, *A two-scale numerical subgrid technique for waterflood simulations*, SPE Journal, 27 (2002), pp. 446–457.
[7] T. Arbogast, S. Minkoff, and P. Keenan, *An operator-based approach to upscaling the pressure equation*, in Computational Methods in Water Resources XII, V. Burganos, G. Karatzas, A. Payatakes, C. Brebbia, W. Gray, and G. Pinder, eds., Computational Mechanics Publications, Southampton, UK, 1998, pp. 405–412.
[8] A. Bensoussan, J. Lions, and G. Papanicolaou, *Asymptotic Analysis for Periodic Structure*, North–Holland, New York, 1979.
[9] D. Bergman, J. Lions, G. Papanicolaou, F. Murat, L. Tartar, and E. Sanchez-Palencia, *Les méthodes de l'homogénéisation: théorie et applications en physique*, Editions Eyrolles, Paris, 1985.
[10] Z. Chen and T. Hou, *A mixed multiscale finite element method for elliptic problems with oscillating coefficients*, Math. Comp., 72 (2002), pp. 541–576.
[11] M. Christie, *Upscaling for reservoir simulation*, J. Pet. Tech., 48 (1996), pp. 1004–1010.
[12] G. Cohen, *Higher-Order Numerical Methods for Transient Wave Equations*, Springer-Verlag, New York, 2002.
[13] M. Dobrin and C. Savit, *Introduction to Geophysical Prospecting*, 4th ed., McGraw–Hill, New York, 1988.
[14] J. Goff, K. Holliger, and A. Levander, *Modal fields: A new method for characterization of random seismic velocity heterogeneity*, Geophys. Res. Lett., 21 (1994), pp. 493–496.
[15] J. Goff and T. Jordan, *Stochastic modeling of seafloor morphology: Inversion of sea beam data from second-order statistics*, J. Geophys. Res., 93 (1988), pp. 13589–13608.
[16] K. Holliger and A. Levander, *A stochastic view of lower crustal fabric based on evidence from the Ivrea Zone*, Geophys. Res. Lett., 19 (1992), pp. 1153–1156.
[17] T. Hou and X. Wu, *A multiscale finite element method for elliptic problems in composite materials and porous media*, J. Comput. Phys., 134 (1997), pp. 169–189.
[18] T. Hughes, *Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods*, Comput. Methods Appl. Mech. Engrg., 127 (1995), pp. 387–401.
[19] E. W. Jenkins, B. Rivière, and M. F. Wheeler, *A priori error estimates for mixed finite element approximations of the acoustic wave equation*, SIAM J. Numer. Anal., 40 (2002), pp. 1698–1715.
[20] P. Joly, *Variational Methods for Time-Dependent Wave Propagation Problems*, Lect. Notes Comput. Sci. Eng. 31, Springer-Verlag, Berlin, 2003.
[21] P. King and J. Williams, *Upscaling permeability: Mathematics of renormalization*, Transp. Porous Media, 23 (1996), pp. 337–354.

[22] O. Korostyshevskaya and S. Minkoff, *A matrix analysis of operator-based upscaling for the wave equation*, SIAM J. Numer. Anal., to appear.
[23] J. Lions, *Some Method in Mathematical Analysis of Systems and Their Control*, Gordon and Breach, New York, 1981.
[24] S. Minkoff, C. Stone, S. Bryant, and M. Peszynska, *Coupled geomechanics and flow simulation for time-lapse seismic modeling*, Geophysics, 69 (2004), pp. 200–211.
[25] C. Ober, R. Gjertsen, S. Minkoff, and D. Womble, *3d finite-difference seismic migration with parallel computers*, in Mathematical and Numerical Aspects of Wave Propagation, J. DeSanto, ed., SIAM, Philadelphia, 1998, pp. 381–385.
[26] C. Ober, R. Oldfield, D. Womble, and J. VanDyke, *Seismic imaging on the Intel Paragon*, J. Computers & Math. with Applic., 35 (1998), pp. 65–72.
[27] P. Pacheco, *Parallel Programming with MPI*, Morgan Kaufmann, San Francisco, 1997.
[28] M. Peszynska, M. Wheeler, and I. Yotov, *Mortar upscaling for multiphase flow in multiblock domains*, Comput. Geosci., 6 (2002), pp. 315–332.
[29] R. Richtmyer and K. Morton, *Difference Methods for Initial-Value Problems*, 2nd ed., Interscience, New York, 1967.
[30] T. Russell and M. Wheeler, *The Mathematics of Reservoir Simulation*, SIAM, Philadelphia, 1983.
[31] T. Vdovina and S. Minkoff, *Convergence of Operator-Based Upscaling for the Wave Equation*, manuscript.
[32] X. Wu, Y. Efendiev, and T. Hou, *Analysis of upscaling absolute permeability*, Discrete Contin. Dyn. Syst. Ser. B, 2 (2002), pp. 185–204.
[33] I. Yotov, *Mortar mixed finite element methods on irregular multiblock domains*, IMACS Series in Comp. Appl. Math., 4 (1998), pp. 239–244.