

HIGH RESOLUTION FORWARD AND INVERSE EARTHQUAKE MODELING ON TERASCALE COMPUTERS*

VOLKAN AKÇELİK[†], JACOBO BIELAK[†], GEORGE BIROS[‡], IOANNIS EPANOMERITAKIS[†], ANTONIO FERNÁNDEZ[†], OMAR GHATTAS[§], EUI JOONG KIM[†], JULIO LÓPEZ[¶], DAVID O'HALLARON^{||},
TIANKAI TU^{**}, AND JOHN URBANIC^{††}

Abstract. For earthquake simulations to play an important role in the reduction of seismic risk, they must be capable of high resolution and high fidelity. We have developed algorithms and tools for earthquake simulation based on multiresolution hexahedral meshes. We have used this capability to carry out 1 Hz simulations of the 1994 Northridge earthquake in the LA Basin using 100 million grid points. Our wave propagation solver sustains 1.21 teraflop/s for 4 hours on 3000 AlphaServer processors at 80% parallel efficiency. Because of uncertainties in characterizing earthquake source and basin material properties, a critical remaining challenge is to invert for source and material parameter fields for complex 3D basins from records of past earthquakes. Towards this end, we present results for material and source inversion of high-resolution models of basins undergoing antiplane motion using parallel scalable inversion algorithms that overcome many of the difficulties particular to inverse heterogeneous wave propagation problems.

1. Introduction. The main objective of our research is to develop the capability for generating realistic inversion-based models of complex basin geology and earthquake sources, and to use this capability to model and forecast strong ground motion during earthquakes in such large basins as Los Angeles. This problem is of great importance to hazard mitigation, because assessing the ground motion to which structures will be exposed during their lifetimes is an essential first step in designing earthquake-resistant facilities and retrofitting existing structures. Thus, ground motion modeling and forecasting are necessary precursors of the design process. The Los Angeles region is a particularly critical and appropriate basin to study, because it is the most highly populated seismic region in the U.S., it has well-characterized geological structures (including a varied fault system), and extensive records of past earthquakes are available.

Modeling and forecasting earthquake ground motion in large basins is a challenging and complex task. The complexity arises from several sources. First, multiple spatial scales characterize the earthquake source and basin response: the shortest wavelengths are measured in

*This work was supported by the National Science Foundation's Knowledge and Distributed Intelligence (KDI) and Information Technology Research (ITR) programs (through grants CMS-9980063, ACI-0121667, and ITR-0122464), the Department of Energy's Scientific Discovery through Advanced Computation (SciDAC) program through the Terascale Optimal PDE Simulations (TOPS) Center, the Computer Science Research Institute at Sandia National Laboratories, and a grant from the Intel Corporation. Computing resources on the HP AlphaCluster system at the Pittsburgh Supercomputing Center are NSF/AAB/PSC award BCS020001P.

[†]Mechanics, Algorithms, and Computing Laboratory, Department of Civil & Environmental Engineering, Carnegie Mellon University, Pittsburgh, Pennsylvania, 15213, USA.

[‡]Courant Institute, for Mathematical Sciences, New York University, New York, NY, 10012, USA.

[§]Mechanics, Algorithms, and Computing Laboratory, Departments of Biomedical Engineering and Civil & Environmental Engineering, Carnegie Mellon University, Pittsburgh, Pennsylvania, 15213, USA.

[¶]Electrical and Computer Engineering Department, Carnegie Mellon University, Pittsburgh, Pennsylvania, 15213, USA.

^{||}Computer Science Department and Electrical and Computer Engineering Department, Carnegie Mellon University, Pittsburgh, Pennsylvania, 15213, USA.

^{**}Computer Science Department, Carnegie Mellon University, Pittsburgh, Pennsylvania, 15213, USA.

^{††}Pittsburgh Supercomputing Center, Pittsburgh, Pennsylvania, 15213, USA.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SC'03, November 15-21, 2003, Phoenix, Arizona, USA Copyright 2003 ACM 1-58113-695-1/03/0011...\$5.00

tens of meters, whereas the longest measure in kilometers; basin dimensions are on the order of tens of kilometers, and earthquake sources up to hundreds of kilometers. Second, temporal scales vary from the hundredths of a second necessary to resolve the highest frequencies of the earthquake source up to as much as several minutes of shaking within the basin. Third, many basins have highly irregular geometry. Fourth, the soils' material properties are highly heterogeneous. And fifth, geology and source parameters are observable only indirectly, and thus introduce uncertainty into the modeling process. Because of its modeling and computational complexity and its importance to hazard mitigation, earthquake simulation has been recognized in several U.S. Federal agency reports as one of the computational Grand Challenges.

In this paper we present recent work that extends our earlier capabilities for large-scale forward modeling of earthquake ground motion [7] to larger basins, with softer soils, and for higher resolved frequencies, all of which add significant computational complexity, yet are crucial for practical applications. The key components that have enabled these capabilities include an octree-based out-of-core mesh generator that can generate unstructured wavelength-tailored hexahedral meshes of sizes limited only by available disk space, and an unstructured hexahedral mesh parallel elastic wave propagation solver that uses very little memory and scales to thousands of processors with high parallel efficiency and good node performance, despite the highly irregular meshes needed to capture efficiently the wide range of spatial and temporal scales that characterize heterogeneous basin response.

We have used these tools to simulate the 1994 Northridge earthquake in the Greater LA Basin at 1 Hz maximum frequency resolution and 100 m/s minimum shear wave velocity. The resulting unstructured mesh contains over 100 million grid points and 80 million hexahedral finite elements, which places it among the largest unstructured mesh simulations ever conducted.¹ These are the most highly resolved simulations of the Northridge earthquake carried out to date; they are made possible by the multiresolution octree-based meshes we use (a uniform grid would have required over 1000 times more grid points to resolve the same frequencies), as well as the low memory required by the hexahedral data structures. The simulation code exhibits nearly 90% parallel efficiency in scaling from 1 to 2048 processors on LeMieux, the HP AlphaServer system at the Pittsburgh Supercomputing Center; it sustains nearly a teraflop/s over 12 hours in solving the 300 million wave propagation ODEs that result upon spatial discretization; and executes at 25% of the peak floating point rate on the 2 Gflops/s Alpha processors—excellent figures considering the highly irregular, multiresolution meshes that we use.²

These levels of performance are due in part to effective use of LeMieux's fast Quadrics network, and in part due to the design of our hexahedral code's data structures and algorithms so that they eschew sparse matrix-vector products in favor of more cache-friendly local dense matrix-vector products. Most importantly, we now have in place a mesh generation and wave propagation framework that will enable us to scale efficiently up to the 2–4 Hz frequencies that are of critical interest to design engineers.³ Results from a 2 Hz simulation—which involves 1.2 billion unstructured grid points—will be presented at SC2003. Because of the low memory required by our hexahedral solver, there is sufficient capacity on the current AlphaServer system to accommodate the 2 Hz simulation. Because of the finer granularity,

¹Model elasticity problem with up to 700 million grid points have been solved on the Earth Simulator, but although the code employs an unstructured finite element data structure, the tests were conducted on regular cubic grids [28, 29].

²The 25% of peak figure compares favorably with single processor efficiencies of about 33% that are beginning to be reported for the Earth Simulator supercomputer for *unstructured* meshes [28, 29].

³Each doubling of frequency leads to a factor of 8 increase in grid size and factor of 16 increase in work, for a given material model.

we expect that parallel performance will be even better than that reported here for the 1 Hz simulations.

Such high resolution simulations have been able to reproduce observed ground motion from past earthquakes at some locations that could not be captured by lower resolution simulations. Observations at other sites, however, have not been reproduced well by high resolution ground motion simulations. This discrepancy is likely the result of uncertainties in both the source and the geological model. Thus, we are led to an *inverse problem*: we wish to estimate the soil property distribution that results in a predicted response that most closely matches observed records of past earthquakes. This inverse problem requires knowledge of the earthquake source, which means we must invert for the source model in the process of inverting for the material model.

We have developed a capability for earthquake inversion for both material and source properties. In this paper we describe the methodology and provide typical results for a 2D earthquake model; results from 3D inversion will be presented at SC2003. The inverse problem is significantly more difficult to solve than the associated forward wave propagation problem. Even when the forward problem is well-posed, possesses a unique and continuous solution, can be evolved in time to obtain a solution, and is characterized by sparse operators—the inverse problem is ill-posed and characterized by multiple solutions that are discontinuous, and has an operator that is dense and couples the entire time-history of response. Specialized algorithms are therefore required, and these are described below.

The capability to invert for the source and for the crustal and basin structures permits us to generate improved earthquake source models and improved basin material models. This, in turn, permits us to model an ensemble of potential rupture scenarios, which improves our ability to forecast strong ground motion during future earthquakes, an essential first step in assessing the earthquake hazard and reducing the seismic risk. The ingredients necessary to achieve this goal are a forward earthquake simulation capability that scales to highly-resolved geologic models and frequencies of engineering interest on terascale supercomputers; and an inversion capability that addresses all of the fundamental challenges of inverse wave propagation while also scaling to large problem sizes, high resolution, and large numbers of processors.

Below we describe our efforts to create these capabilities. Section 2 describes the earthquake elastic wave propagation model, finite element approximation, and explicit wave propagation solution, and presents verification, performance, and scalability data. The inverse model and algorithm are presented in Section 3, and used to solve an inverse shear wave propagation problem for unknown earthquake source and basin structure.

2. Forward earthquake modeling. The Quake group has been working on modeling earthquakes in large basins on parallel supercomputers for over a decade. Over the years our forward earthquake modeling codes have run on the TMC CM-2, Intel iWarp and Paragon, SGI Origin, Cray T3D and T3E, and most recently the HP AlphaServer cluster. Each generation of architecture has prompted evolutionary changes to our algorithms and software implementations. Our simulations are based on multiresolution mesh algorithms, which overcome many of the obstacles related to the wide range of length and time scales characterizing basin earthquake response. We have pursued these methods despite the challenges they pose for obtaining good node performance and parallel scalability on highly parallel cache-based systems. In heterogeneous geological structures such as sedimentary basins, where material properties vary significantly throughout the domain, multiresolution meshes allow a tremendous reduction in the number of grid points (compared to uniform meshes), because element sizes can adapt locally to the highly-variable wavelengths of propagating seismic waves. Furthermore, a mesh tailored to local wavelengths permits much longer time steps

without suffering instability, since the Courant stability limit is on the order of that required for accuracy. Details on our computational methodology and underlying algorithms may be found in [6, 7, 10–12, 25, 32, 40]. Our codes have been used to model earthquake ground motion in the San Fernando Valley of Southern California [7], the Osaka basin in Japan [22], the Kirovakan Valley in Armenia [12], and the Wellington Valley in New Zealand [2]; to model the response of dams during earthquakes [31]; to model nonlinear elastoplastic ground motion [39]; and to assess three-dimensional local site effects in sedimentary basins [9].

Our earlier earthquake codes were based on linear tetrahedral finite elements. Recently, we have designed a new code that includes octree-based trilinear hexahedral elements and local dense element-based data structures [24] with several important advantages:

- The hexahedra provide somewhat greater accuracy per grid point (the asymptotic convergence rate is unchanged, but the constant is typically improved over tetrahedral approximation).
- The element-based data structure produces much better cache utilization by relegating the work that requires indirect addressing (and is memory bandwidth-limited) to vector operations, and recasting the majority of the work of the matrix-vector product as local element-wise dense matrix computations. The result is a significant boost in performance.
- The hexahedral meshes stem from wavelength-adapted octrees, which are more easily generated than general unstructured tetrahedral meshes, particularly when the number of elements increases above 50 million. We have developed an efficient out-of-core octree-based hexahedral mesh generator [37] that can generate meshes of sizes that are limited only by available disk space. (Since each basin is meshed just once for a given resolution of interest—but subjected to many earthquake scenarios—mesh generation can be done off-line.)
- The hexahedra all have the same element stiffness matrices, modulo element size and material properties (which are stored as vectors), and thus no matrix storage is required at all. This results in a substantial decrease in required memory—about an order of magnitude, compared to our grid-point-based tetrahedral code.

These features permit earthquake simulations to substantially greater resolutions than heretofore possible. The subsections below describe the wave propagation solution and mesh generation methods, assess performance and scalability on PSC’s HP AlphaServer system, and provide some typical verification data.

2.1. Wave propagation model, discretization, and solver. We model seismic wave propagation in the earth via Navier’s equation of linear elastodynamics. Let \mathbf{u} represent the vector field of the three displacement components, λ and μ the Lamé moduli and ρ the density distribution, \mathbf{b} a time-dependent body force representing the seismic source, and \mathbf{L}^{AB} a linear differential operator that vanishes on the free surface, and applies an appropriate absorbing boundary condition on truncation boundaries. Let Ω be an open bounded domain in \mathbb{R}^3 . The initial-boundary value problem is then written as:

$$\begin{aligned} \rho \ddot{\mathbf{u}} - \nabla \cdot [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T) + \lambda (\nabla \cdot \mathbf{u}) \mathbf{I}] &= \mathbf{b} \text{ in } \Omega \times (0, T], \\ [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T) + \lambda (\nabla \cdot \mathbf{u}) \mathbf{I}] \mathbf{n} &= \mathbf{L}^{AB} \mathbf{u} \text{ on } \partial\Omega \times [0, T], \\ \mathbf{u} &= \mathbf{0} \text{ on } \Omega \times \{t = 0\}, \\ \dot{\mathbf{u}} &= \mathbf{0} \text{ on } \Omega \times \{t = 0\}, \end{aligned} \quad (2.1)$$

where \mathbf{n} represents the outward unit normal to the boundary. With this model, longitudinal waves propagate with velocity $v_p = \sqrt{(\lambda + 2\mu)/\rho}$, and shear waves with velocity $v_s = \sqrt{\mu/\rho}$. The continuous form above does not include material attenuation, which we introduce

at the discrete level via a Rayleigh damping model. The vector \mathbf{b} comprises a set of body forces that equilibrate an induced displacement dislocation on a fault plane, providing an effective representation of earthquake rupture on the plane. Explicit expressions for such a body force will be given below in the case of antiplane shear.

On a face with a unit normal \mathbf{n} and two tangential vectors $\boldsymbol{\tau}_1$ and $\boldsymbol{\tau}_2$, such that the three vectors form a right-handed orthogonal coordinate system, the absorbing boundary condition (Stacey's formulation) takes the form

$$\mathbf{S}\mathbf{n} = \begin{bmatrix} -d_1 \frac{\partial}{\partial t} & c_1 \frac{\partial}{\partial \tau_1} & c_1 \frac{\partial}{\partial \tau_2} \\ -c_1 \frac{\partial}{\partial \tau_1} & -d_2 \frac{\partial}{\partial t} & 0 \\ -c_1 \frac{\partial}{\partial \tau_2} & 0 & -d_2 \frac{\partial}{\partial t} \end{bmatrix} \begin{Bmatrix} u_n \\ u_{\tau_1} \\ u_{\tau_2} \end{Bmatrix} \equiv \mathbf{L}^{AB} \mathbf{u}$$

where \mathbf{S} is the stress tensor and

$$\begin{aligned} c_1 &= -2\mu + \sqrt{\mu(\lambda + 2\mu)}, \\ d_1 &= \sqrt{\rho(\lambda + 2\mu)}, \\ d_2 &= \sqrt{\rho\mu}. \end{aligned}$$

Even though Stacey's absorbing boundary is not exact, it is local in both space and time, which is particularly important for large-scale parallel implementation.

2.2. Spatial and temporal approximation. We apply standard Galerkin finite element approximation in space to the appropriate weak form of the initial-boundary value problem (2.1). Let \mathcal{U} be the space of admissible solutions (which depends on the regularity of \mathbf{b}), \mathcal{U}_h be a finite element subspace of \mathcal{U} , and \mathbf{v}_h be a test function from that subspace. Then the weak form is written as follows.

Find $\mathbf{u}_h \in \mathcal{U}_h$ such that

$$\int_{\Omega} \left\{ \rho \ddot{\mathbf{u}}_h \cdot \mathbf{v}_h + \frac{\mu}{2} (\nabla \mathbf{u}_h + \nabla \mathbf{u}_h^T) \cdot (\nabla \mathbf{v}_h + \nabla \mathbf{v}_h^T) + \lambda (\nabla \cdot \mathbf{u}_h) (\nabla \cdot \mathbf{v}_h) - \mathbf{b} \cdot \mathbf{v}_h \right\} d\Omega = \int_{\partial\Omega} (\mathbf{L}^{AB} \mathbf{u}_h) \cdot \mathbf{v}_h dA, \quad \forall \mathbf{v}_h \in \mathcal{U}_h. \quad (2.2)$$

Finite element approximation is effected via piecewise trilinear basis functions and associated trilinear hexahedral elements on an octree mesh. This strikes a balance between simplicity, low memory (since all element stiffness matrices are the same modulo scale factors), and reasonable accuracy.⁴

Upon spatial discretization, we obtain a system of ordinary differential equations of the form

$$\mathbf{M} \ddot{\mathbf{u}} + (\mathbf{C}^{AB} + \alpha \mathbf{M} + \beta \mathbf{K}) \dot{\mathbf{u}} + (\mathbf{K} + \mathbf{K}^{AB}) \mathbf{u} = \mathbf{b}, \quad (2.3)$$

where \mathbf{M} and \mathbf{K} are mass and stiffness matrices, arising from the terms involving ρ and (μ, λ) in (2.2), respectively; \mathbf{b} is a body force vector resulting from a discretization of the seismic source model; and damping matrices \mathbf{C}^{AB} and \mathbf{K}^{AB} are contributions of the absorbing boundaries to the mass and stiffness matrices, respectively. We have also introduced damping matrices in the form of the Rayleigh material model $\alpha \mathbf{M} + \beta \mathbf{K}$ to simulate the effect of energy dissipation and resulting wave attenuation due to anelastic material behavior. The constants α and β are determined locally (elementwise) so that the resulting damping ratio

⁴The output quantities of greatest interest are displacements and velocities, as opposed to stresses.

is as close as possible to a constant value dictated by the local soil type, over a band of frequencies. Since Rayleigh damping increases both linearly and inversely with frequency, we seek a least squares solution to this optimization problem over each element. This provides a reasonable damping model for many soils, although very low and very high frequencies are overdamped.

The time dimension is discretized using central differences. The algorithm is made explicit using a diagonalization scheme that lumps the mass matrix—and possibly \mathbf{C}^{AB} —and splits the diagonal and off-diagonal portions of the stiffness and absorbing boundary damping matrix. The resulting update for the displacement field at time step $k + 1$ is given by

$$\begin{aligned} \left[\left(1 + \alpha \frac{\Delta t}{2} \right) \mathbf{M} + \beta \frac{\Delta t}{2} \mathbf{K}_{diag} + \frac{\Delta t}{2} \mathbf{C}_{diag}^{AB} \right] \mathbf{u}_{k+1} = \\ \left[2\mathbf{M} - \Delta t^2 \left(\mathbf{K} + \mathbf{K}^{AB} \right) - \beta \frac{\Delta t}{2} \mathbf{K}_{off} - \frac{\Delta t}{2} \mathbf{C}_{off}^{AB} \right] \mathbf{u}_k \\ + \left[\left(\alpha \frac{\Delta t}{2} - 1 \right) \mathbf{M} + \beta \frac{\Delta t}{2} \mathbf{K} + \frac{\Delta t}{2} \mathbf{C}^{AB} \right] \mathbf{u}_{k-1} + \Delta t^2 \mathbf{b}_k. \end{aligned} \quad (2.4)$$

The time increment Δt must satisfy a local CFL condition for stability. Space is discretized over an octree mesh (each leaf corresponds to a hexahedral element) that resolves local seismic wavelengths: given a (typically highly-heterogeneous) material property distribution and highest resolved frequency of interest, a local mesh size is chosen to produce p grid points per shortest wavelength (we typically take $p = 10$ for trilinear hexahedra). This insures that the CFL-limited time step is of the order of that needed for accuracy, and that excessive dispersion errors do not arise due to over-refined meshes.

Spatial discretization via refinement of an octree produces a non-conforming mesh, resulting in a discontinuous displacement approximation. Whenever a refined hexahedron neighbors an unrefined one, “hanging” grid points, which belong to refined elements but not to unrefined neighbors, are produced. In such cases, we restore continuity of the displacement field by imposing algebraic constraints that require the displacements at hanging grid points to be consistent with non-hanging neighbors. For linear hexahedra and provided 2-to-1 refinement is enforced between neighbors, these constraints state simply that hanging midside values must be the average of the two non-hanging edge neighbors, and hanging mid-face values must be the average of the four non-hanging vertex neighbors. We can express these discrete continuity constraints in the form

$$\mathbf{u} = \mathbf{B}\bar{\mathbf{u}},$$

where $\bar{\mathbf{u}}$ denotes the displacements at the independent non-hanging grid points, and \mathbf{B} is a sparse constraint matrix. In particular, $B_{ij} = \frac{1}{4}$ if (dependent) hanging grid point i is a face neighbor of (independent) non-hanging point j and $\frac{1}{2}$ if it is an edge neighbor, $B_{ij} = 1$ simply identifies a non-hanging grid point, and $B_{ij} = 0$ otherwise. Rewriting the linear system (2.4) as

$$\mathbf{A}\mathbf{u}_{k+1} = \mathbf{b}(\mathbf{u}_k),$$

we can impose the continuity constraints via the projection

$$\mathbf{B}^\top \mathbf{A} \mathbf{B} \bar{\mathbf{u}} = \mathbf{B}^\top \mathbf{b}(\mathbf{u}_k). \quad (2.5)$$

The constrained update (2.5) remains explicit, since the projected matrix $\mathbf{B}^\top \mathbf{A} \mathbf{B}$ preserves the diagonality of \mathbf{A} . The work involved in enforcing the constraints is proportional to the

number of hanging grid points, which can be a sizeable fraction of the overall number of grid points for a highly irregular octree, but is at most of $\mathcal{O}(N)$. Therefore, the per-iteration complexity of the update (2.5) remains linear in the number of grid points.

The combination of an octree-based wavelength-adaptive mesh, piecewise trilinear Galerkin finite elements in space, explicit central differences in time, constraints that enforce continuity of the displacement approximation, and local-in-space-and-time absorbing boundaries yields a second-order-accurate in time and space method that is capable of scaling up to the very large problem sizes that are required for high resolution earthquake modeling. In the next subsection we discuss our techniques for generating large octree-based wavelength-adaptive hexahedral meshes, and in the section that follows we demonstrate parallel scalability of our implementation to large parallel systems and problem sizes. The final subsection provides sample evidence of the correctness and validity of our simulations.

2.3. Etree method for large octree-based hexahedral mesh generation. Generating very large meshes, of order 10^8 elements and up, remains a major challenge. We have developed a new database-oriented method called *etree* [37] to generate large octree-based hexahedral meshes out of core. With the *etree* method, the limit on the largest mesh size that can be generated is extended to the available disk space, instead of the size of the memory. Any desktop computer with enough disk storage can be used to generate large octree-based meshes required for LA basin models with hundreds of millions of hexahedra.

We use the well-known *linear octree* [19] technique to assign each octant a unique key that encodes its location and size, and we store the octants in a *B-tree* [8, 17, 20, 33], the most commonly used primary key indexing structure in database systems. In addition, we have developed two new techniques, which we refer to as *auto-navigation* and *local balancing*, that address the special needs of octree mesh generation.

All of these components (linear octrees, B-trees, auto-navigation, and local balancing) are implemented in a C library called the *etree library*. An application uses the functions defined in the *etree library* to manipulate an octree mesh stored on disk. The *etree library* automatically performs extensive optimization to improve running time and reduce disk I/O.

Figure 2.1 shows the process of generating a mesh using the *etree* method. In the *construct* step, an octree is constructed in the same way as in an in-core algorithm, except that it is built and stored on disk. The decompositions of the octants are dependent on the geometry or physics being modeled. For earthquake simulations in heterogeneous basins, the shear wave velocity and maximum resolved frequency determine the local element size.⁵ The result is an unbalanced octree. Next, the *balance* step recursively decomposes all the (large) octants that violate the 2-to-1 constraint until no illegal conditions exist. Finally, in the *transform* step, mesh-specific information such as the element-node relationship and the node coordinates are derived from the balanced octree and stored in two databases, one for the mesh elements, the other for the mesh nodes.

We use the linear octree technique to linearize an octree so that we can store the octants in a B-tree. In particular, we assign keys to octants using a variant of the *Morton code* [27]. A Morton code maps a d -dimensional point to a one-dimensional scalar. The mapping can be computed quickly by interleaving the bits of the binary representation of the coordinate of a d -dimensional point. To identify uniquely an octant, we construct its key by appending its level in the octree to the Morton code for its left-lower corner. The result is a binary bit string that can be used as a key for the B-tree indexing.

⁵Shear waves are the slowest waves and thus have the shortest wavelengths. Given a local shear wave velocity (v_s), highest response frequency we wish to resolve (f_{max}), and number of grid points per wavelength deemed sufficient (N_λ , usually 10), we determine a local element size from $h = v_s / (N_\lambda f_{max})$.

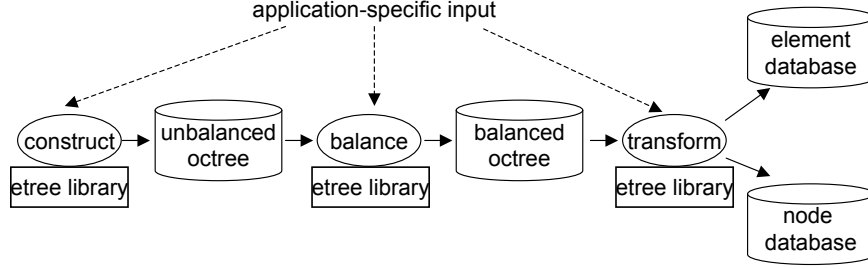


FIG. 2.1. The etree method of generating octree meshes.

The linear octree nicely solves the problem of how to address individual octants. However, it does not provide a programming model for octree construction. Although it is possible for an application to construct an octree by repeatedly inserting and deleting octants from the database, the application program would have to keep records of which octants have been decomposed and which have not. Worse, many insertions are in fact unnecessary because those octants are later decomposed and removed from the database.

To address this problem, we designed a higher-level abstraction to support automatic octree construction. The underlying technique is called *auto-navigation*. The idea of auto-navigation is based on a simple insight: since the ordering of expanding an octree under construction is independent of the correctness of the result, the octree traversal logic can be decoupled from the application's logic and incorporated into the etree library.

An octree obtained after the construction step must be balanced to conform to the 2-to-1 constraint. The etree library provides another high-level abstraction to isolate the applications from the details of enforcing the 2-to-1 constraint. To speed up the balance operation, we have designed a new technique called *local balancing*, which consists of three steps. First, we partition the whole domain into *equal-size blocks*. Next, we conduct *internal balancing* to enforce the 2-to-1 constraint within each block. Finally, we do *boundary balancing* to resolve interactions between adjacent blocks. Our results show that local balancing can achieve a speed-up ranging from 8 to 28 depending on the size of the meshes being balanced.

For example, a mesh of the LA Basin with 110,185,484 elements, 19,543,510 hanging grid points, and 129,019,552 total grid points required about 100 Gb of disk space and 42 hours of wall-clock time on an Intel PIII 1 GHz machine with 3 Gb of memory. This time is inconsequential, since the same mesh is used repeatedly to simulate numerous earthquakes corresponding to different fault rupture scenarios. Finally, at the time of writing, we have obtained preliminary data on the generation of a mesh with 1,184,517,967 elements and 1,219,193,420 grid points. This required an estimated 1 Tb of disk space and 9 days to generate, on an Intel Xeon 2.2 GHz with 1 Gb of memory.

In summary, the etree method is a useful and effective way of generating very large octree-based meshes that adapt to local seismic wavelengths. The maximum mesh size that can be generated is limited only by available disk space.

2.4. Performance of the wave propagation solver. In this section we provide data that demonstrate the excellent parallel performance and scalability of our octree-based multiresolution wave propagation code. We focus on *parallel efficiency* as measured by the degradation in megaflops/s sustained performance per processor as the number of processors and problem size increase. The other ingredient for scalability is *algorithmic efficiency*, which in general requires a work-optimal algorithm. This is typically achieved by an algorithm whose

TABLE 2.1

*Parallel scalability of unstructured octree-based earthquake code on PSC HP AlphaServer. **PEs** is number of processors; **model** identifies the LA Basin model and associated minimum period resolved; **grid pts** is the number of grid points in the model; **pts/PE** is grid points per processor; **Gflops** is the total gigaflops/s sustained over the duration of the earthquake simulation, exclusive of I/O; **Mflops/PE** is the corresponding megaflops/s per processor; and **efficiency** is the parallel efficiency measured by degradation in **Mflops/PE** relative to a single processor.*

PEs	model	grids pts	pts/PE	Gflops	Mflops/PE	efficiency
1	LA10S	134,500	134,500	0.505	505	1.00
16	LA5S	618,672	38,667	7.85	491	0.972
128	LA2S	14,792,064	115,563	60.0	469	0.929
512	LA1HA	47,556,096	92,883	231	451	0.893
1024	LA1HB	101,940,152	99,551	460	450	0.891
2048	LA1HB	101,940,152	49,775	907	443	0.874
3000	LA1HB	101,940,152	33,980	1,210	403	0.800

per-iteration cost is linear in problem size, and iteration count is constant or polylogarithmic in problem size.

For 3D wave propagation problems in which the grid size is adapted to the local wavelength of propagating waves, an optimal solver has complexity $\mathcal{O}(N^{\frac{4}{3}})$, where $N^{\frac{1}{3}}$ is the number of grid point in each direction. This results from the fact that simply writing the solution requires $\mathcal{O}(N^{\frac{4}{3}})$ complexity, since $\mathcal{O}(N)$ grid points are required for accurate spatial resolution, and $\mathcal{O}(N^{\frac{1}{3}})$ time steps for accurate temporal resolution, which is of the order of that dictated by the CFL stability condition. Thus, the explicit wave propagation solver described in Section 2.1 is asymptotically optimal, and our only concern in the present section is to evaluate parallel efficiency.

Table 2.1 presents sample scalability results for a series of 1994 Northridge earthquake simulations in the LA Basin with highest resolved frequencies ranging from 0.1 Hz to 1 Hz, corresponding to a range of problem sizes from 134,500 to over 100 million grid points, on a number of processors ranging from 1 to 3000 on PSC's HP AlphaServer system. Due to the unstructured mesh (and target frequencies of interest) the number of grid points per processor varies from a low of 33,980 for the 3000 processor simulation to a high of 134,500 on a single processor⁶. Overall performance increases from 505 megaflops/s on one processor (25% of the Alpha EV68's peak) to 1.21 teraflops/s on 3000 processors. The code exhibits outstanding performance, particularly for such a highly irregular problem. It maintains 87% parallel efficiency while scaling from 1 to 2048 processors. The largest simulation, consisting of over 300 million ODEs discretized over 60,000 time steps, sustains 1.21 teraflops/s over 4 hours of wall-clock time for the wave propagation solution⁷ to simulate 60s of the Northridge earthquake. For this 3000 processor simulation, parallel efficiency drops to 80%, due in part to the lower granularity. To our knowledge, this represents the largest *unstructured mesh* PDE simulation ever conducted. Most importantly, had this problem been solved with a regular grid code, it would have required 2×10^{11} grid points, i.e. a factor of 2000 greater than that used by our multiresolution code. Thus, we could have improved our flop rate by using a regular grid data structure, but we would have needed at least 2 years of 3000-processor LeMieux time to solve the same problem—assuming it would still fit in memory (actually it would not).

In conclusion, our hexahedral code is more cache-friendly and faster, a factor of ten more

⁶This range of granularity is typical for our production simulations.

⁷We exclude time for I/O, which has not been optimized.

memory-efficient, and more scalable to large problem sizes (due to the octree-based mesh generator) than our tetrahedral code. Indeed, this version combines the best of structured and unstructured mesh methods: the low memory per grid point characteristic of regular grid codes, along with the local adaptivity and multiresolution capability of unstructured codes.

2.5. Verification. Our hexahedral forward earthquake simulation code has been verified against closed-form solutions and against our earlier tetrahedral code in ground motion simulations in the Los Angeles region. Figure 2.2 shows volume renderings of velocities from a simulation for which such a closed-form solution is available. The figure depicts snapshots at different times of the wave propagation in a layer over a halfspace due to an idealized earthquake generated over an extended strike-slip fault. Agreement between the finite element simulation and the Green’s function solution is excellent.

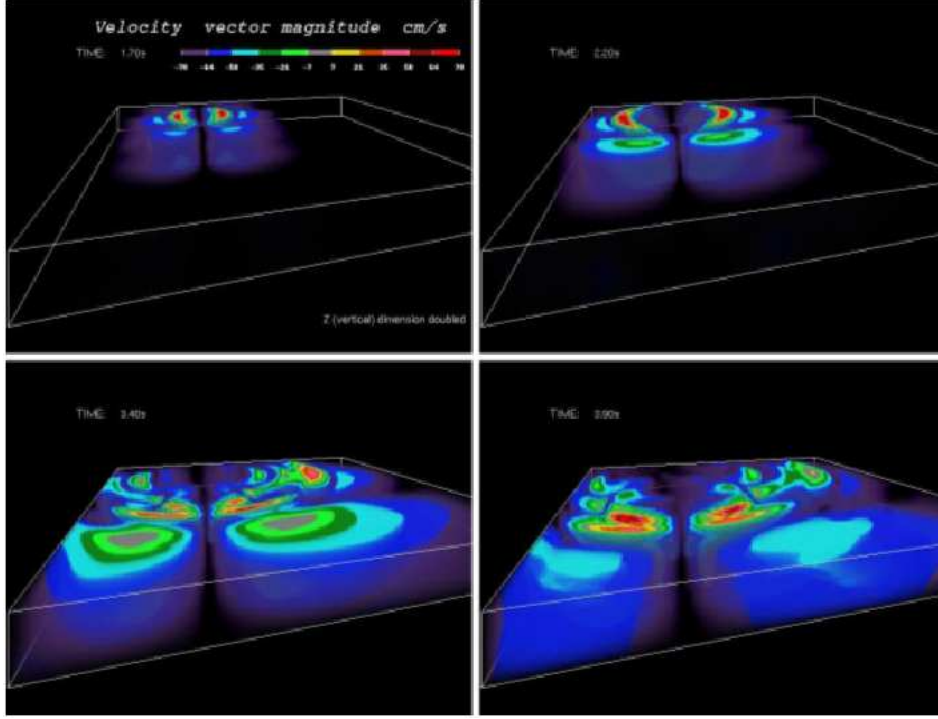


FIG. 2.2. Visualization of wave propagation in a layer over a halfspace due to an earthquake generated over an extended vertical strike-slip fault. Figure shows volume-rendered distribution of the fault parallel component of the horizontal velocity at different times following the onset of the excitation.

In order to assess the accuracy of our hexahedral code for problems for which closed-form solutions do not exist, we compare against ground motion computed using our tetrahedral wave propagation code. The tetrahedral code has been verified against four different finite difference codes from the earthquake modeling community, and thus serves as a good benchmark. That comparison showed that all five codes produce similar results [18]. To carry out the simulation, we use the same Southern California Earthquake Center (SCEC) Community Velocity Model and an idealized model of the 1994 Northridge earthquake that was used in the study. Figure 2.3a shows a plan view and a cross-section of the shear wave velocity distribution in the LA Basin model. Figure 2.3b shows an unstructured hexahedral mesh used to resolve frequencies up to 0.2 Hz, while 2.3c displays a closeup view of a portion

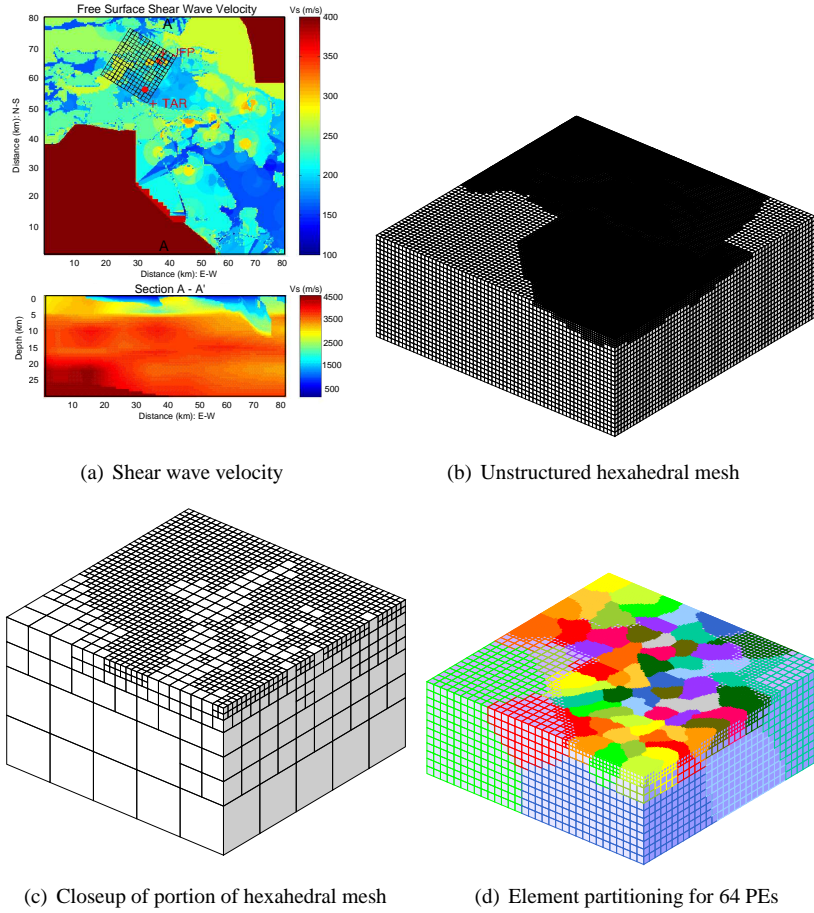


FIG. 2.3. LA Basin model. (a) Plan view and cross-section of distribution of shear wave velocity, along with projection of fault plane on free surface. (b) Corresponding unstructured octree-based hexahedral mesh that resolves seismic wavelengths to 0.2 Hz maximum frequency for illustrative purposes. (c) Portion of hexahedral mesh near free surface, illustrating octree mesh structure and 2-to-1 refinement constraints. (d) Element partition for 64 processors.

of the mesh near the free surface, illustrating the 2-to-1 refinement constraint on the underlying octree. Finally, Figure 2.3d depicts a 64-processor partitioning. We use the parallel graph partitioning code ParMETIS [23].

Figure 2.4 shows a comparison between our tetrahedral and hexahedral codes for two different frequencies. This hexahedral simulation was performed for a maximum frequency of 1.0 Hz, with a 100 million-grid point finite element mesh on 1024 processors of the AlphaServer system. The benchmark tetrahedral simulation was limited to 0.5 Hz maximum frequency due to difficulties in generating the required half-billion-tetrahedra mesh. Our hexahedral-based approach has made it possible, for the first time, to perform an earthquake simulation to 1 Hz maximum frequency for realistic properties of the entire Los Angeles Basin. Until now, the maximum frequency considered for the LA Basin had been 0.5 Hz. The results show very good agreement for the lower frequency. As expected, significant differences are observed for 1.0 Hz because our tetrahedral model cannot represent the ground motion at this higher frequency. Notice the higher amplitude of the ground motion at 1.0 Hz and the higher frequency content of the synthetics, which is not captured by the coarser

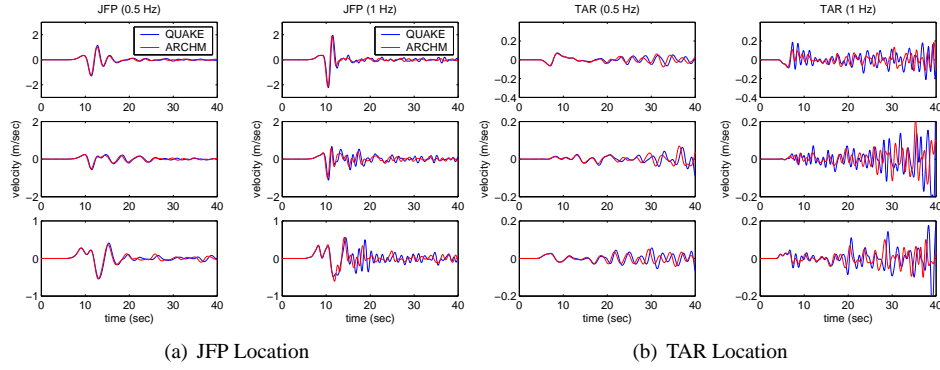


FIG. 2.4. Comparison between the results of our tetrahedral (red) and hexahedral (blue) codes for the ground motion velocity in two horizontal directions and the vertical directions at two locations, identified on the upper left frame; one set of results has been low pass filtered to 0.5 Hz, and the other to 1.0 Hz. The lower frequency corresponds to the maximum frequency modeled by our tetrahedral simulation; the higher, to that of our hexahedral simulation.

tetrahedral model. For design purposes, structural engineers are interested in even higher frequencies, up to 2–4 Hz. Even with an adaptive mesh, a 4 Hz simulation will require about 10 billion elements—underscoring the critical importance of scalability of meshing and wave propagation algorithms.

To illustrate the spatial variation of the ground motion, Figure 2.5 presents snapshots at different times of an animation of the wave propagation throughout the basin. Notice the directivity of the ground motion along strike from the epicenter and the concentration of motion near the fault corners. This pattern was also observed during the actual earthquake.⁸ Strong ground motion was also observed in the southeast portion of the San Fernando Valley (Sherman Oaks), in the Santa Monica area, and La Cienega area (in the middle of the Los Angeles Basin proper). These observations are not reproduced in the simulations depicted in the figure. This discrepancy is due, in part, to the simplified nature of the source, but also because the current SCEC model does not yet provide the necessary fidelity. This underscores the importance of developing the capability to invert for both source and material models from records of past earthquakes.

3. Inverse earthquake modeling. As noted in the previous section, while current simulations (ours and others’) provide much useful information, in many cases they are not capable of adequately reproducing observed seismograms. The reason for this is that these models are based on a number of restrictive assumptions. Some, such as limiting the highest resolved frequency and the lowest value of the shallow soil layers are made largely to reduce the computational effort, while others are necessitated by our limited knowledge of the earthquake source and of the material properties of the geological structure on the length scale of a seismic wavelength. All of these have a profound impact on our ability to predict earthquake ground motion. Thus there is a critical need to perform seismic inversion of the seismic source and the material properties of the geological model using high resolution 3D elastodynamic wave propagation models.

The recent advances described in the previous section on forward simulation of ground motion by octree-based multiresolution mesh methods provide an excellent opportunity for developing more accurate inversion-based models of the entire earthquake process, from its

⁸see USGS map at <http://pasadena.wr.usgs.gov/north/>.

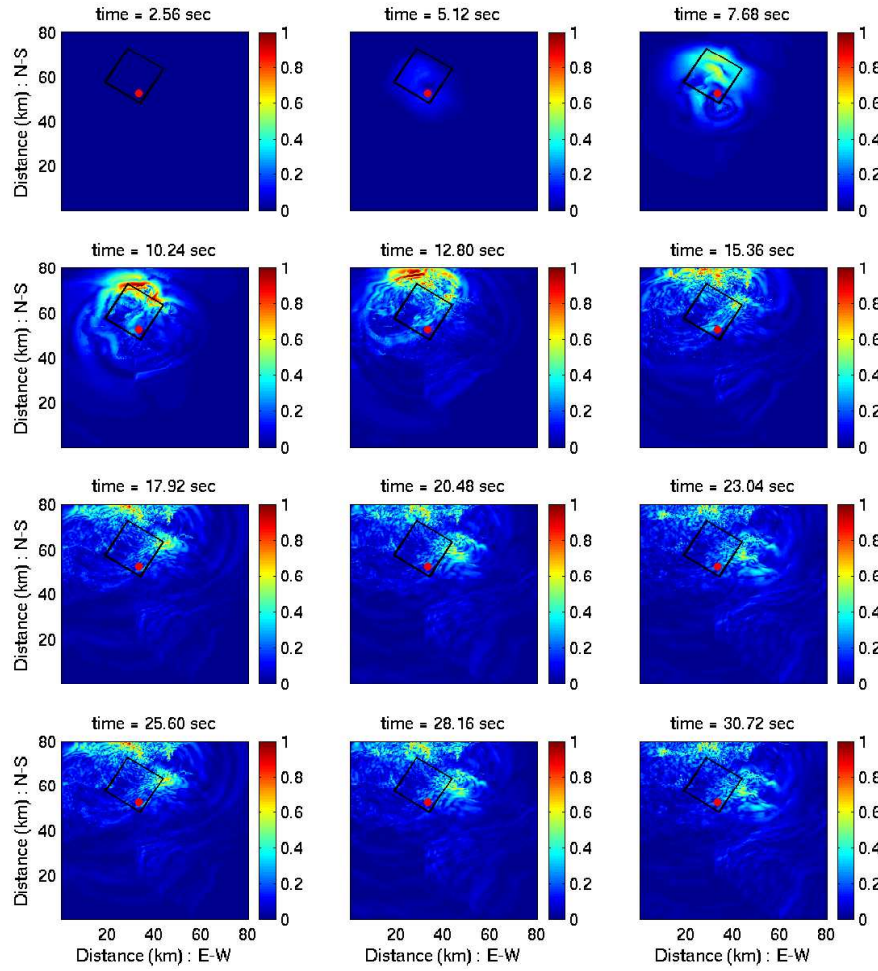


FIG. 2.5. Snapshots of propagating waves from simulation of 1994 Northridge earthquake.

inception at the source, through the wave propagation path, to basin and local site effects. What makes such an effort even more timely and warranted are recent cooperative seismic networks such as TriNet and the USArray of Earthscope, which feature hundreds of broadband and strong-motion sensors. They will add immensely to the collection of earthquake records, and will make it possible for inverse methods to yield accurate results.

However, solution of inverse wave propagation problems presents numerous difficulties, including severe nonlinearity, ill-posedness, multiple solutions, space-time coupling, discontinuous solutions, and dense ill-conditioned operators—despite the fact that the forward wave propagation problem is linear, well-posed, has a unique solution whose computation can be time-marched, and possesses a sparse well-conditioned operator. We have previously developed a parallel scalable method that addresses these difficulties, and demonstrated it for inverse acoustic wave propagation problems [3]. This section presents its application to the solution of seismic inverse problems. We report here on results for separate material and source inversion for a 2D sedimentary basin undergoing antiplane motion.

3.1. Nonlinear optimization formulation and inversion algorithm. In this section we state a nonlinear optimization formulation of the inverse wave propagation problem and develop first order necessary conditions for optimality [36]. For simplicity of the presentation, we consider a simplification of the 3D elastodynamic wave propagation model of the previous section to a shear wave propagation model. Let Ω be an open bounded domain in \mathbb{R}^2 , representing a vertical section through a basin, and let Σ represent a fault plane within Ω . The free surface is denoted as Γ_{FS} , and absorbing boundary conditions are imposed on the remaining boundaries, denoted Γ_{AB} , to limit reflections from outgoing waves. For illustrative purposes, here we employ a simple first-order absorbing boundary condition. Let $u(\mathbf{x}, t)$ represent the transverse displacement, and $\rho(\mathbf{x})$ and $\mu(\mathbf{x})$ the density and shear modulus of the soil. The seismic source appears in strong form as a dipole along the fault; it is parameterized by a delay time $T(\Sigma)$, a rise time $t_0(\Sigma)$, and a dislocation magnitude $u_0(\Sigma)$. The normal to the fault is taken as \mathbf{n}_Σ . Every point along the fault has a dislocation function $g(t)$, that depends on the delay time, rise time, and dislocation magnitude, as shown in Figure 3.1. The dislocation function is such that its time derivative is a hat function, as depicted in the figure.

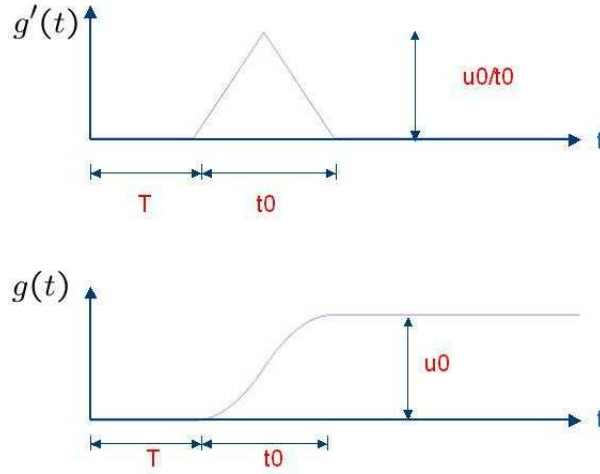


FIG. 3.1. *Seismic source model.*

The nonlinear least squares formulation of the inverse problem can be posed as follows: find the material property distribution $\mu(\mathbf{x})$ and earthquake source parameters $T(\Sigma)$, $t_0(\Sigma)$, and $u_0(\Sigma)$, so that the L_2 norm error between predicted and observed displacements is minimized, subject to satisfaction of the shear wave propagation equation. The error norm is continuous in time and pointwise in space, corresponding to the locations of the receivers. This problem is ill-posed in the sense that highly-oscillatory components of the material field are poorly determined by the observations; thus, some form of regularization to render the problem well-posed is required. In the following formulation, we employ a mixture of Tikhonov regularization for the source parameters, and total variation regularization [1, 38] for the material field:

Minimize w.r.t. μ, u_0, t_0, T

$$\begin{aligned} & \frac{1}{2} \sum_{j=1}^{N_r} \int_{\Omega} \int_0^T (u - u^*)^2 \delta(\mathbf{x} - \mathbf{x}_j) dt d\Omega \\ & + \frac{\beta_1}{2} \int_{\Omega} |\nabla \mu| d\Omega + \frac{\beta_2}{2} \int_{\Omega} |\nabla u_0|^2 d\Omega + \frac{\beta_3}{2} \int_{\Omega} t |\nabla t_0|^2 d\Omega + \frac{\beta_4}{2} \int_{\Omega} |\nabla T|^2 d\Omega \end{aligned}$$

subject to:

$$\begin{aligned} \rho \ddot{u} - \nabla \cdot \mu \nabla u &= -\nabla \cdot (\mu u_0 g(t_0, T) \delta(\Sigma) \mathbf{n}_{\Sigma}) \quad \text{in } \Omega \times (0, T] \\ \mu \nabla u \cdot \mathbf{n} &= 0 \quad \text{on } \Gamma_{FS} \times (0, T] \\ \mu \nabla u \cdot \mathbf{n} &= \sqrt{\rho \mu} \dot{u} \quad \text{on } \Gamma_{AB} \times (0, T] \\ u = \dot{u} &= 0 \quad \text{in } \Omega \times \{t = 0\} \end{aligned} \quad (3.1)$$

The Tikhonov regularization terms penalize oscillations (along the fault) in the source parameters, while the total variation regularizer also inhibits oscillations but in addition avoids smoothing of discontinuities in the material field, thereby preserving sharp interfaces prevalent in layered geologic media [3].

First-order necessary conditions can be derived from the optimization problem (3.1) in strong form, to yield the following.

State wave equation:

$$\begin{aligned} \rho \ddot{u} - \nabla \cdot \mu \nabla u &= -\nabla \cdot (\mu u_0 g(t_0, T) \delta(\Sigma) \mathbf{n}_{\Sigma}) \quad \text{in } \Omega \times (0, T] \\ \mu \nabla u \cdot \mathbf{n} &= 0 \quad \text{on } \Gamma_{FS} \times (0, T] \\ \mu \nabla u \cdot \mathbf{n} &= \sqrt{\rho \mu} \dot{u} \quad \text{on } \Gamma_{AB} \times (0, T] \\ u = \dot{u} &= 0 \quad \text{in } \Omega \times \{t = 0\} \end{aligned} \quad (3.2)$$

Adjoint wave equation:

$$\begin{aligned} \rho \ddot{\lambda} - \nabla \cdot \mu \nabla \lambda &= \sum_{j=1}^{N_r} (u^* - u) \delta(\mathbf{x} - \mathbf{x}_j) \quad \text{in } \Omega \times (0, T] \\ \mu \nabla \lambda \cdot \mathbf{n} &= 0 \quad \text{on } \Gamma_{FS} \times (0, T] \\ \mu \nabla \lambda \cdot \mathbf{n} &= -\sqrt{\rho \mu} \dot{\lambda} \quad \text{on } \Gamma_{AB} \times (0, T] \\ \lambda = \dot{\lambda} &= 0 \quad \text{in } \Omega \times \{t = T\} \end{aligned} \quad (3.3)$$

Material equation:

$$\begin{aligned} -\beta_1 \nabla \cdot \frac{\nabla \mu}{|\nabla \mu|} + \int_0^T (\nabla \lambda \cdot \nabla u + u_0 g(t_0, T) \delta(\Sigma) \nabla \lambda \cdot \mathbf{n}_{\Sigma}) dt &= 0 \quad \text{in } \Omega \\ \nabla \mu \cdot \mathbf{n} &= 0 \quad \text{on } \Gamma_{FS} \\ \frac{1}{|\nabla \mu|} \nabla \mu \cdot \mathbf{n} - \frac{1}{2} \int_0^T \sqrt{\frac{\rho}{\mu}} \lambda \dot{u} dt &= 0 \quad \text{on } \Gamma_{AB} \end{aligned} \quad (3.4)$$

Dislocation amplitude equation:

$$-\beta_2 \Delta u_0 + \int_0^T \mu g(t_0, T) \nabla \lambda \cdot \mathbf{n}_{\Sigma} dt = 0 \quad \text{on } \Sigma \quad (3.5)$$

Rise time equation:

$$-\beta_3 \Delta t_0 + \int_0^T \mu u_0 \frac{dg}{dt_0} \nabla \lambda \cdot \mathbf{n}_\Sigma dt = 0 \quad \text{on } \Sigma \quad (3.6)$$

Delay time equation:

$$-\beta_4 \Delta T + \int_0^T \mu u_0 \frac{dg}{dT} \nabla \lambda \cdot \mathbf{n}_\Sigma dt = 0 \quad \text{on } \Sigma \quad (3.7)$$

Equations (3.2)–(3.7) constitute a system of coupled nonlinear integro-partial differential equations including the initial-boundary value state wave equation (3.2), the final-boundary value adjoint wave equation (3.3), the material boundary value equation (3.4), and the dislocation amplitude, rise time, and delay time equations (3.5)–(3.7) that are posed on Σ . These are to be solved for the displacement u , the adjoint variable λ , the shear modulus μ , the dislocation amplitude u_0 , the rise time t_0 , and the delay time T .

We discretize the system (3.2)–(3.7) by Galerkin finite elements in space, and explicit central differences in time (in particular the discretizations for the state wave equation and adjoint wave equation are the same as described in the previous section). We solve the discretized system using the multiscale Gauss-Newton-conjugate gradient method described in [3] and briefly reviewed here. Our solver is built from components of the PETSc library [5]. At every CG iteration, the state wave equation is solved forward in time for given material and fault properties and their Newton increments, and the adjoint wave equation is solved backward in time using the computed states and state increments, with optional use of algorithmic checkpointing [21]. A CG iteration on the Gauss-Newton linearization of the reduced system of material, dislocation amplitude, rise time, and delay time equations then yields CG updates to the Newton increments. The coefficient matrix of the latter system is guaranteed to be positive definite with appropriate regularization, and an Armijo-type backtracking line search assures global convergence [30]. The Gauss-Newton approximation provides quadratic convergence for exact fit problems, and in general is rapid for small residual problems.

Since each CG iteration requires one forward and one adjoint wave propagation solution, it is essential to have a good preconditioner. We use the reduced Hessian preconditioner introduced in [13, 14] and implemented in the Veltisto package, which is based on a limited memory BFGS update [26] that has been initialized with several Frankel two-step stationary iterations [4] on the reduced system. For large material contrasts, occasionally the Newton method strays into negative material property territory, which destabilizes the wave propagation solution. This is prevented through a logarithmic barrier method to enforce non-negativity constraints [30] (which we know are non-active at the solution). One of the nefarious properties of the nonlinear optimization formulation of the inverse wave propagation problem is the existence of numerous local minima, possessing a radius of Newton convergence proportional to the wavelength of propagating waves. The algorithm described above is prone to entrapment in local minima [34], and several remedies have been proposed [16, 35]. Here we appeal to multiscale grid and frequency continuation [3, 15], which in our experience circumvents the problem by keeping successively finer scale inversion estimates within the radius of the ball of convergence.

Since the majority of the work in this algorithm involves repeated solution of forward or backward wave equations, and since parallel scalability of the forward wave propagation solver has already been demonstrated in Section 2.4, the crucial remaining issue is *algorithmic scalability* of the inversion algorithm. Our main interest is that the number of both linear and nonlinear iterations grows at most weakly with problem size. Table 3.1 provides evidence in the case of material field inversion that our inversion algorithm is scaling well. In

TABLE 3.1

Algorithmic scalability of inversion algorithm for scalar 3D wave equation case. Number of nonlinear iterations, total number of linear iterations, and average number of linear iterations per nonlinear iteration are shown as a function of the number of inversion parameters, which correspond to material grid points. The wave propagation grid is fixed at 274,625 unknowns, partitioned on 64 processors of the PSC HP AlphaServer, for all material grid sizes up to 274,625. For the 2,146,689 grid case, both wave propagation and material grids coincide, and are partitioned onto 256 processors. Results show essentially mesh independence of nonlinear and linear iterations.

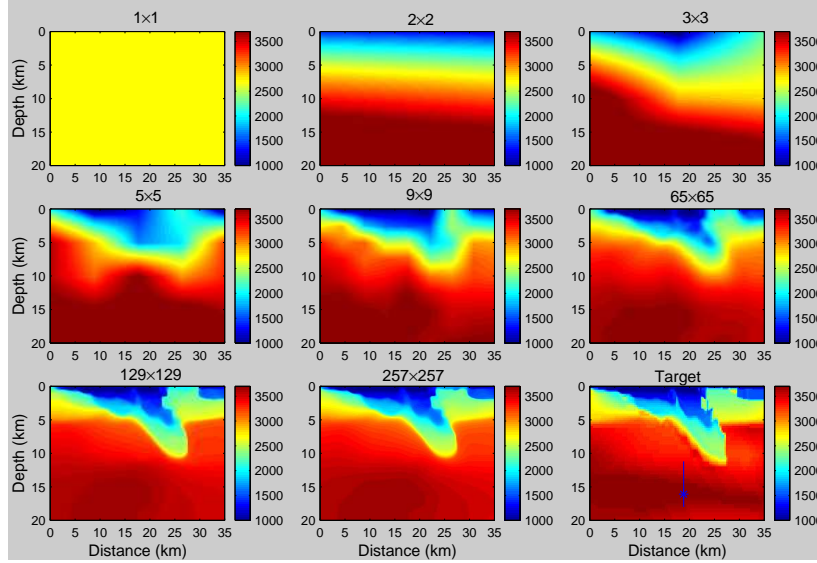
material grid	nonlinear iter	total linear iter	avg linear iter
125	17	144	8.5
729	12	249	21.0
4,913	12	396	33.0
35,937	25	439	17.6
274,625	19	370	19.5
2,146,689	22	436	19.8

fact the number of iterations is observed to be independent of problem size, once the mesh is sufficiently fine.

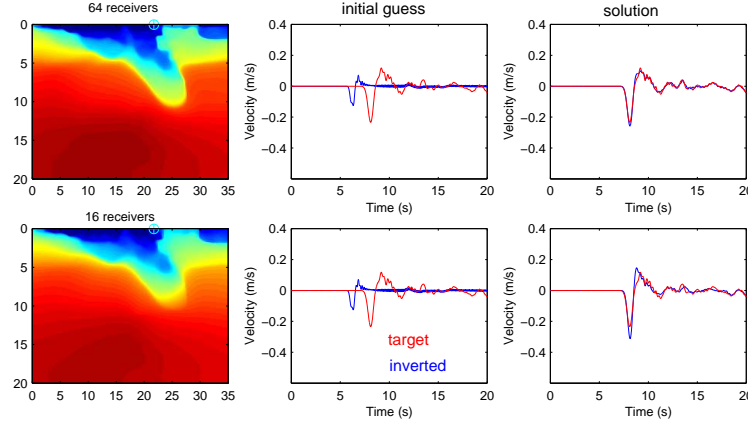
3.2. Application to antiplane ground motion inversion. In this section we illustrate the application of the inversion algorithm to a portion of the vertical cross-section of the Los Angeles basin in Figure 2.3, as shown at the bottom right frame in Figure 3.2a (the one labeled *Target*). The vertical line in the middle bottom of the target image represents the trace of a strike-slip fault running perpendicular to the valley, and the solid dot on it denotes the hypocenter of an idealized earthquake source. We regard the material properties of this cross-section and this source as the target model. In our numerical experiments, we use waveforms synthesized from the target model on the free surface as pseudo-observed data. First, we assume that the source is known and invert for the shear wave velocity distribution of the geological structure. We have assumed that the material is lossless and that the density distribution is known. The source is defined by its rupture velocity and slip function. The rupture velocity is taken to be constant (with T , the time it takes to arrive from the hypocenter to an arbitrary point on the fault), and the slip velocity is defined by an isosceles triangle, with duration t_0 , and amplitude u_0 . Five percent random noise has been added artificially to examine the effect of noise.

Figure 3.2a shows a sequence of inverted material models, corresponding to increasingly finer inversion grids, as dictated by our multiscale algorithm, and culminating with the 257×257 grid in the next-to-last image. The shear wave velocity is approximated on a piecewise bilinear mesh. This inversion was performed with 64 observers distributed uniformly on the free surface. The high fidelity of the results is noteworthy: even fine features, of the size of a fraction of a wavelength, are accurately resolved on the finest grid. Figure 3.2b compares results for 64 and 16 receivers. As expected, the resolution of the inverted model for 16 receivers is not as sharp as for 64 receivers, yet it too approximates closely the target model. The synthetics in Figure 3.2b show the velocity at one non-receiver location for the two models, both for the initial guess and for the inverted solution. The results show that even with just 16 receivers and for a comparison at a non-receiver location, the waveforms of the inverted material models remain quite close to the target.

Results of the source inversion are shown in Figure 3.3. The first, middle, and right columns show results for the initial guess, fifth iteration, and the converged solution (target values are in red). The latter essentially coincides with the target source. Inversion for *either* source or material parameters represents a major computational challenge when the forward problem is a 3D high-resolution earthquake simulation. When *both* are unknown (referred to



(a) Stages of multiscale inversion



(b) Effect of receiver density

FIG. 3.2. (a) Stages of the multiscale inversion, starting from a homogeneous guess. The last image shows the hypocenter and fault superposed on the target distribution. (b) Wave velocity profiles (left column) obtained using 64 receivers (top row) and 16 receivers (bottom row). Middle and right column: velocity history comparison at a non-receiver location (location of receiver is shown in the wave velocity profiles). Initial guess is shown in the middle column and the converged solution is in the right column. Red is target, blue is inverted result.

as the blind deconvolution problem), this problem is even more challenging.

4. Summary. The inverse problem of determining source, elastic, and attenuation parameters for complex 3D basins presents enormous computational challenges. Towards this end, we have developed the capability to mesh and solve forward earthquake simulation problems on multiresolution hexahedral meshes that will scale to billion grid point meshes of the LA Basin—which correspond to earthquake simulations to 2 Hz maximum frequency and 100 m/s lowest shear wave velocity—using present supercomputing resources. We have used this capability to carry out 1 Hz simulations of the 1994 Northridge earthquake with 100

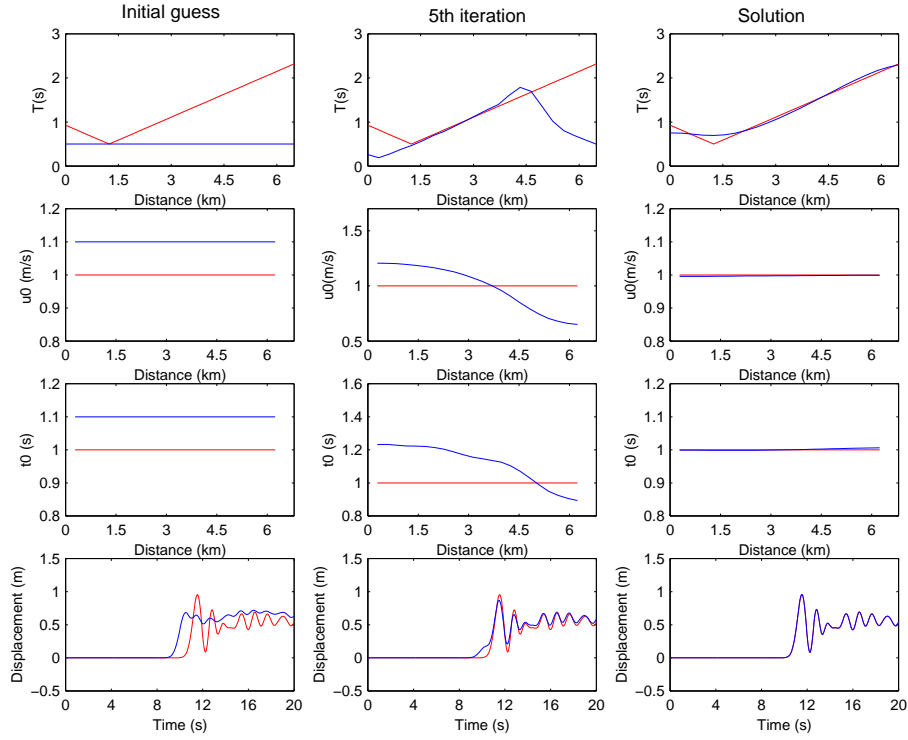


FIG. 3.3. Source inversion: left column, initial guess; middle column, 5th iteration; right column, converged solution. Red shows the target and blue shows the inverted results. Top row, function $T(x)$ represents the delay time; second row, function $u_0(x)$ represents the amplitude of the source; third row, $t_0(x)$ represents the ramp time of the source; fourth row, displacement history comparison at a receiver location.

million grid points; our wave propagation solver sustains 1.21 teraflop/s for 4 hours on 3000 AlphaServer processors at 80% parallel efficiency.

We have presented results for material and source inversion of high-resolution models of 2D basins undergoing antiplane motion using parallel scalable inversion algorithms that overcome many of the difficulties particular to inverse heterogeneous wave propagation problems. Our forward simulations are among the largest unstructured mesh computations reported to date, requiring multiple hours on thousands of processors. Yet these forward problems are but a mere “inner iteration” of the inverse problem, which invokes the (forward and backward) wave propagation simulations repeatedly. Even though the number of iterations taken by our inversion algorithm is shown to be independent of the problem size, it does require over 400 total iterations for inversion of a 2.2 million parameter problem—implying over 800 wave propagation simulations to solve the inverse problem. This places the goal of inversion of high resolution source and basin models from records of past earthquakes among the largest nonlinear inverse problems currently contemplated.

Acknowledgements. We thank Steve Day, Tom Jordan, Carl Kesselman, Phil Maechlin, Harold Magistrale, Kim Olsen, and our colleagues on the Southern California Earthquake Center (SCEC) Community Modeling Environment Project, for their suggestions and encouragement. Thanks to Natassa Ailamaki and Christos Faloutsos for helping us understand spatial databases.

REFERENCES

- [1] R. ACAR AND C. R. VOGEL, *Analysis of bounded variation penalty methods for ill-posed problems*, Inverse Problems, 10 (1994), pp. 1217–1229.
- [2] B. ADAMS, R. DAVIS, AND J. BERRILL, *Modeling site effects in the Lower Hutt Valley, New Zealand*, in 12th World Conference on Earthquake Engineering, Auckland, NZ, 2000.
- [3] V. AKÇELİK, G. BIROS, AND O. GHATTAS, *Parallel multiscale Gauss-Newton-Krylov methods for inverse wave propagation*, in Proceedings of IEEE/ACM SC2002 Conference, Baltimore, MD, Nov. 2002. SC2002 Best Technical Paper Award.
- [4] O. AXELSSON, *Iterative Solution Methods*, Cambridge Press, 1994.
- [5] S. BALAY, K. BUSCHELMAN, W. D. GROPP, D. KAUSHIK, M. KNEPLEY, L. C. MCINNES, B. F. SMITH, AND H. ZHANG, *PETSc home page*. <http://www.mcs.anl.gov/petsc>, 2001.
- [6] H. BAO, J. BIELAK, O. GHATTAS, L. F. KALLIVOKAS, D. R. O'HALLARON, J. R. SHEWCHUK, AND J. XU, *Earthquake ground motion modeling on parallel computers*, in Supercomputing '96, Pittsburgh, Pennsylvania, Nov. 1996.
- [7] ———, *Large-scale simulation of elastic wave propagation in heterogeneous basins on parallel computers*, Computer Methods in Applied Mechanics and Engineering, 152 (1998), pp. 85–102.
- [8] R. BAYER AND E. M. MCCREIGHT, *Organization and maintenance of large ordered indices*, Acta Informatica, 1 (1972), pp. 173–189.
- [9] J. BIELAK, Y. HISADA, H. BAO, AND O. GHATTAS, *One- vs two- or three-dimensional effects in sedimentary valleys*, in Proc. 12th World Conference on Earthquake Engineering, Auckland, New Zealand, 2000.
- [10] J. BIELAK, L. KALLIVOKAS, J. XU, AND R. MONOPOLI, *Finite element absorbing boundary for the wave equation in a halfplane with an application to engineering seismology*, in Proc. of the Third International Conference on Mathematical and Numerical Aspects of Wave Propagation, Mandelieu-la-Napule, France, Apr. 1995, INRIA-SIAM, pp. 489–498.
- [11] J. BIELAK, K. LOUKAKIS, Y. HISADA, AND C. YOSHIMURA, *Domain reduction method for three-dimensional earthquake modeling in localized regions, part I: Theory*, Bulletin of the Seismological Society of America, (2003). In press.
- [12] J. BIELAK, J. XU, AND O. GHATTAS, *Earthquake ground motion and structural response in alluvial valleys*, Journal of Geotechnical and Geoenvironmental Engineering, 125 (1999), pp. 413–423.
- [13] G. BIROS AND O. GHATTAS, *Parallel Lagrange-Newton-Krylov-Schur methods for PDE constrained optimization. Part I: The Krylov-Schur solver*, SIAM Journal on Scientific Computing, (2003). Under revision.
- [14] ———, *Parallel Lagrange-Newton-Krylov-Schur methods for PDE constrained optimization. Part II: The Lagrange-Newton solver*, SIAM Journal on Scientific Computing, (2003). Under revision.
- [15] C. BUNKS, F. M. SALECK, S. ZALESKI, AND G. CHAVENT, *Multiscale seismic waveform inversion*, Geophysics, 50 (1995).
- [16] G. CHAVENT, F. CLÉMENT, AND S. GÓMEZ, *Waveform inversion by MBTT formulation*, in Proceedings of the 3rd International Conference on Mathematical and Numerical Aspects of Wave Propagation, Philadelphia, 1995, SIAM, pp. 713–722.
- [17] D. COMER, *The ubiquitous B-Tree*, ACM Computing Surveys, 11 (1979), pp. 121–137.
- [18] S. M. DAY, *PEER/SCEC validation and verification study of ground motion simulation codes*, tech. report, SCEC Annual Meeting, Oxnard, California, September 2002.
- [19] I. GARGANTINI, *An effective way to represent quadtrees*, Communications of the ACM, 25 (1982), pp. 905–910.
- [20] J. GRAY AND A. REUTER, *Transaction Processing: Concepts and Techniques*, Morgan Kaufmann Publishers, Sep 1992, ch. 15.
- [21] A. GRIEWANK, *Achieving logarithmic growth of temporal and spatial complexity in reverse automatic differentiation*, Optimization Methods and Software, 1 (1992), pp. 35–54.
- [22] Y. HISADA, H. BAO, J. BIELAK, O. GHATTAS, AND D. O'HALLARON, *Simulations of long-period ground motions during the 1995 Hyogoken-Nanbu (Kobe) earthquake using 3D finite element method*, in 2nd International Symposium on Effect of Surface Geology on Seismic Motion, Special Volume on Simultaneous Simulation for Kobe, K. Irikura, H. Kawase, and T. Iwata, eds., Yokohama, Japan, Dec. 1998, pp. 59–66.
- [23] G. KARYPIS, K. SCHLOEGEL, AND V. KUMAR, *ParMETIS 3.0*. <http://www-users.cs.umn.edu/~karypis/metis/parmetis>.
- [24] E. KIM, J. BIELAK, AND O. GHATTAS, *Large-scale Northridge Earthquake simulation using octree-based multiresolution mesh method*, in Proceedings of the 16th ASCE Engineering Mechanics Conference, Seattle, Washington, July 2003.
- [25] E. KIM, J. BIELAK, O. GHATTAS, AND J. WANG, *Octree-based finite element method for large-scale earthquake ground motion modeling in heterogeneous basins*, Eos Transactions AGU, 83 (2002). Fall Meet.

- Suppl., Abstract S12B-1221.
- [26] J. L. MORALES AND J. NOCEDAL, *Automatic preconditioning by limited memory quasi-Newton updating*, SIAM Journal on Optimization, 10 (2000), pp. 1079–1096.
 - [27] G. M. MORTON, *A computer oriented geodetic data base and a new technique in file sequencing*, tech. report, IBM, Ottawa, Canada, 1966.
 - [28] K. NAKAJIMA, *Three-level hybrid vs. flat MPI on the Earth Simulator: Parallel iterative solvers for finite-element method*, Applied Numerical Mathematics, (2003). Submitted.
 - [29] K. NAKAJIMA AND H. OKUDA, *Parallel iterative solvers for finite-element methods using a hybrid programming model on SMP cluster architectures*, SIAM Journal on Scientific Computing, (2003). Submitted.
 - [30] J. NOCEDAL AND S. J. WRIGHT, *Numerical Optimization*, Springer, 1999.
 - [31] A. PAPALOU AND J. BIELAK, *Seismic elastic response of earth dams with canyon interaction*, Journal of Geotechnical Engineering, 127 (2001), pp. 446–453.
 - [32] F. J. SANCHEZ-SESMA, R. BENTES, AND J. BIELAK, *The assessment of strong ground motion: What lies ahead?*, in Proceedings of the 11th World Conference on Earthquake Engineering, Acapulco, Mexico, 1996. Paper No. 2014.
 - [33] A. SILBERSCHATZ, H. F. KORTH, AND S. SUDARSHAN, *Database system concepts*, McGraw Hill Companies, Inc., third ed., 1997, ch. 11.
 - [34] W. W. SYMES, *Velocity inversion: A case study in infinite-dimensional optimization*, Mathematical Programming, 48 (1990), pp. 71–102.
 - [35] W. W. SYMES AND J. J. CARAZZONE, *Velocity inversion by differential semblance optimization*, Geophysics, 56 (1991), pp. 654–663.
 - [36] A. TARANTOLA, *Inversion of seismic reflection data in the acoustic approximation*, Geophysics, 49 (1984), pp. 1259–1266.
 - [37] T. TU, D. O’HALLARON, AND J. LOPEZ, *Etree: A database-oriented method for generating large octree meshes*, in Proceedings of the Eleventh International Meshing Roundtable, Ithaca, NY, Sep 2002, pp. 127–138.
 - [38] C. R. VOGEL AND M. E. OMAN, *Iterative methods for total variation denoising*, SIAM Journal on Scientific Computing, 17 (1996), pp. 227–238.
 - [39] J. XU, J. BIELAK, O. GHATTAS, AND J. WANG, *Large-scale modeling of earthquake ground motion in elastoplastic basins*, Physics of the Earth and Planetary Interiors, (2003). In press.
 - [40] C. YOSHIMURA, J. BIELAK, Y. HISADA, AND A. FERNANDEZ, *Domain reduction method for three-dimensional earthquake modeling in localized regions, part II: Verification and applications*, Bulletin of the Seismological Society of America, (2003). In press.