Computers and Structures 100-101 (2012) 54–69

Contents lists available at SciVerse ScienceDirect

Computers and Structures

journal homepage: www.elsevier.com/locate/compstruc

Simulation of multistage excavation based on a 3D spectral-element method

Hom Nath Gharti [a,b,*], Volker Oye [a], Dimitri Komatitsch [c], Jeroen Tromp [d]

[a] NORSAR, Gunnar Randers vei 15, N-2007 Kjeller, Norway
[b] Department of Geosciences, Oslo University, Sem Sælands vei 1, N-0371 Oslo, Norway
[c] LMA, CNRS UPR 7051, Aix-Marseille University, Centrale Marseille, 13402 Marseille Cedex 20, France
[d] Department of Geosciences and Program in Applied & Computational Mathematics, Princeton University, Princeton, NJ, USA

ARTICLE INFO

Article history:
Received 15 June 2011
Accepted 12 March 2012

Keywords:
Spectral-element method
Multistage excavation
Mining
Elastoplasticity
Parallel processing

ABSTRACT

We implement a 3D spectral-element method for multistage excavation problems. To simulate excavation in elastoplastic soils, we employ a Mohr–Coulomb yield criterion using an initial strain method. We parallelize the software based on non-overlapping domain decomposition using MPI. We verify the uniqueness principle for multistage excavation in linear elastic materials. We validate our serial and parallel programs, and illustrate several examples of multistage excavation in elastoplastic materials. Finally, we apply our software to a model of the Pyhäsalmi ore mine in Finland. Strong-scaling performance tests involving multistage excavation show that the parallel program performs reasonably well for large-scale problems.

1. Introduction

Excavation is a common process in many geotechnical engineering constructions, for example, underground caverns, tunneling, mining, and road construction (e.g., [1]). During excavation, a certain portion of the model is removed in stages, thereby significantly changing the geometry. Accordingly, the domain, domain boundary, and boundary conditions of the problem change with each excavation stage. After excavation, newly generated surfaces become traction-free boundaries (e.g., [2]). Therefore, not only the geometry of the excavation but also the resulting stress redistribution have important consequences for the stability of the excavated region. Simulation of multistage excavation provides progressive information during the excavation process, which may be important for economic design and efficient risk mitigation.

The finite-element method (FEM) is usually used to simulate the excavation process. The FEM is a powerful tool for solving boundary value problems, and is widely used in many applications of solid and fluid mechanics (e.g., [3–5]). The original idea behind FEM-based simulations of multistage excavation was to nullify contributions of nodes of an excavated element by considering an infinitesimal stiffness (e.g., [6–8]). In a linear elastic medium, multistage excavation satisfies the uniqueness principle first postulated by Ishihara (e.g., [9,10]). The uniqueness principle is based on the principle of superposition; it states that the final solution is independent of the sequence of excavation. Based on the infinitesimal stiffness approach, it was not always possible to satisfy the uniqueness principle due to spurious contributions from excavated nodes. To overcome this problem, Desai and Sargand [11] proposed a hybrid method to compute nodal loads. Similarly, Borja et al. [12,13] developed a method employing infinitesimal stiffnesses and a variational formulation. Comodromos et al. [14] suggested a method based on the so-called ‘variable domain vector’, which basically tracks intact and excavated entities so that contributions of excavated nodes may be removed by static condensation of the resulting matrix system of the FEM during the solution procedure (e.g., [15,4]). Smith and Griffiths [16] used a similar technique, forming the global equations of the FEM only for intact elements. There are several other studies that deal with excavation based on the FEM, e.g., 2D excavation models [17,18], stability of vertical excavation in plane-strain and axisymmetric problems [19], and deep excavation [20]. There are also some studies involving 3D excavation, e.g., 3D modeling of tunneling [21], deep excavation in Shanghai [22], and simulations based on parallel processing [23]. Other related research includes transient analysis of excavation [24] and excavation in poroelastic media [25]. The discrete-element method (DEM; e.g., [26,27]) is also used to simulate excavation (e.g., [28]). The DEM is often suitable for discontinuous media to simulate brittle failure. However, for reliable simulations, a large number of discrete particles is required, which involves large computational costs. Therefore, particular types of FEMs, such as the extended finite-element method (XFEM; e.g., [29,30]), the particle discretization scheme finite-element

---

* Corresponding author. Address: Department of Geosciences, Princeton University, Princeton, NJ, USA. Tel.: +1 609 258 2393.

E-mail addresses: hgharti@princeton.edu (H.N. Gharti), volker@norsar.no (V. Oye), komatitsch@lma.cnrs-mrs.fr (D. Komatitsch), jtromp@princeton.edu (J. Tromp).

0045-7949/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.compstruc.2012.03.005

method (PDS-FEM; e.g., [31,32]), or the combined finite-discrete element method (e.g., [33]) may be of interest.

Most simulations of multistage excavation are limited to either 2D or simple 3D models. In this study, we develop a parallel 3D package based on a spectral-element method, which may be used to simulate excavation in complex 3D models of various scales.

The spectral-element method (SEM) is a higher-order FEM in which integration over an element is based on nodal quadrature, e.g., Gauss–Legendre–Lobatto quadrature. In nodal quadrature, interpolation and integration points coincide. In view of nodal quadrature, the SEM has two main advantages: (1) interpolating functions become orthogonal at quadrature points, resulting in a diagonal mass matrix, thereby greatly simplifying the time-marching algorithm because a fully explicit scheme may be used (e.g., [3,5]), and (2) interpolation is unnecessary to determine nodal quantities from quadrature-point quantities and vice versa, thus simplifying pre- and postprocessing. Therefore, higher-order elements are easier to implement. In 2D and 3D models, nodal quadrature was originally limited to specific types of elements, e.g., quadrilaterals and hexahedra. Hexahedral meshing is a challenging task and an area of active research (e.g., [34–36]). Only a few hexahedral meshing tools are currently available, e.g., CUBIT [37], TrueGrid [38], and Gmsh [39]. Meshing is usually not fully automated, and careful manual mesh design is often necessary. There have been successful implementations of the SEM with other types of elements, for instance, triangles in 2D and tetrahedra in 3D, using so-called Fekete points (e.g., [40–43]). Since nodal quadrature includes boundary points, the order of numerical integration may not always be sufficiently high [44]. Due to the high spatial accuracy, however, the influence of low-order integration on accuracy and convergence may not be critical, depending on the problem (e.g., [45,46]).

The SEM was originally applied to problems in computational fluid dynamics [47,48]. Detailed reviews may be found in, e.g., Cohen [49] and Deville et al. [50]. Recently, it has been widely used to simulate seismic wave propagation from local to global scales (e.g., [51–58]). Similarly, the nonlinear SEM has been used for 2D viscoplastic problems [59]. Gharti et al. [60] have developed a software package for 3D slope stability analysis based on the elastoplastic spectral-element method. The software has been parallelized for large-scale problems.

Several problems in geomechanics, e.g., excavation, slope failure, and mine or tunnel collapse, exhibit elastoplastic deformation, which is an inherently nonlinear process. Nonlinear problems are usually solved using a nonlinear finite-element method (e.g., [3,61,5,62]). Higher-order finite elements are often desirable to capture nonlinear behaviour efficiently (e.g., [63–66]). Due to nonlinearity and the use of higher-order elements, computation is often demanding for elastoplastic problems. Therefore, the SEM may be a suitable tool for these problems.

In this article, we present a 3D spectral-element implementation for multistage excavation problems. This work is an extension to the software package for 3D slope stability analysis [60]. In the excavation algorithm, excavated elements are used solely to compute surface tractions in intact regions. Processing is performed only in intact elements without ever computing stiffnesses for excavated elements. This approach leads to a robust and efficient algorithm. We use an element-by-element preconditioned conjugate-gradient solver for efficient storage (e.g., [67]). The software is parallelized based on non-overlapping domain decomposition using MPI (message passing interface; e.g., [68,69]). We verify the uniqueness principle for multistage excavation in linear elastic materials. We validate both serial and parallel programs with an example of multistage excavation in an elastoplastic material, and apply the technique to an underground ore mine in Finland. Finally, we present strong-scaling performance tests of the parallel code.

2. Formulation

2.1. Discretization of the governing equations

The governing equations for elastostatic problems may be written in index notation as

$$\sigma_{ij;j} + f_i = 0, \tag{1}$$

subject to the boundary conditions

$$\begin{aligned} u_i &= \hat{u}_i \quad \text{on} \quad \Gamma_u \\ t_i &= \sigma_{ij} n_j = \hat{t}_i \quad \text{on} \quad \Gamma_t, \end{aligned} \tag{2}$$

where $\sigma_{ij} = C_{ijk\ell}\varepsilon_{k\ell}$ represents the stress tensor, $\varepsilon_{k\ell} = \frac{1}{2}(u_{k;\ell} + u_{\ell;k})$ the strain tensor, $C_{ijk\ell}$ the fourth-order tensor relating stress and strain, and f_i the force term. The normal to the boundary is denoted by n_i, and $\hat{u}_i$ and $\hat{t}_i$ are the prescribed displacement and traction on boundaries Γ_u and Γ_t, respectively (Fig. 1). We use the summation convention for repeated indices, and a semicolon (;) denotes covariant differentiation.

The weak form of the governing Eq. (1) is

$$\int_\Omega w_{i;j}\sigma_{ij}\mathrm{d}\Omega = \int_\Omega w_i f_i \mathrm{d}\Omega + \int_{\Gamma_t} w_i \hat{t}_i \mathrm{d}\Gamma, \tag{3}$$

where w_i denotes a test function, and Ω and Γ the volume and boundary of the domain, respectively.

For multistage excavation problems, the domain, domain boundary, and boundary conditions change with each subsequent excavation stage. Therefore, Eqs. (1)–(3) are generalized to multistage excavation. More specifically, the stress tensor at excavation stage s depends on the previous excavation stage $s-1$ via

$$\sigma_{ij}^s = \sigma_{ij}^{s-1} + \Delta\sigma_{ij}^s, \tag{4}$$

where $\Delta\sigma_{ij}^s$ denotes the incremental stress due to the removal of certain portions at excavation stage s.

Spectral-element discretization and integration techniques are explained in detail in the literature (e.g., [48–50,53,56]). For completeness, we briefly summarize the procedure here. In the spectral-element method (SEM), the displacement field is discretized using interpolation functions defined over Gauss–Legendre–Lobatto (GLL) points (Fig. 2) via

$$u_i(\xi) = \sum_{\alpha=1}^{N} u_i^\alpha \phi^\alpha(\xi), \tag{5}$$

where u_i^α and ϕ^α denote nodal displacements and interpolation functions, respectively, and N denotes the total number of GLL

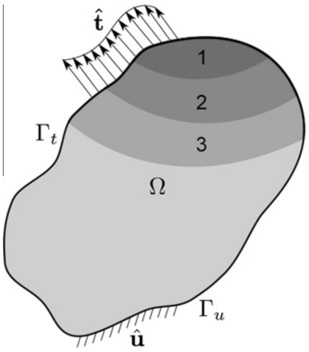
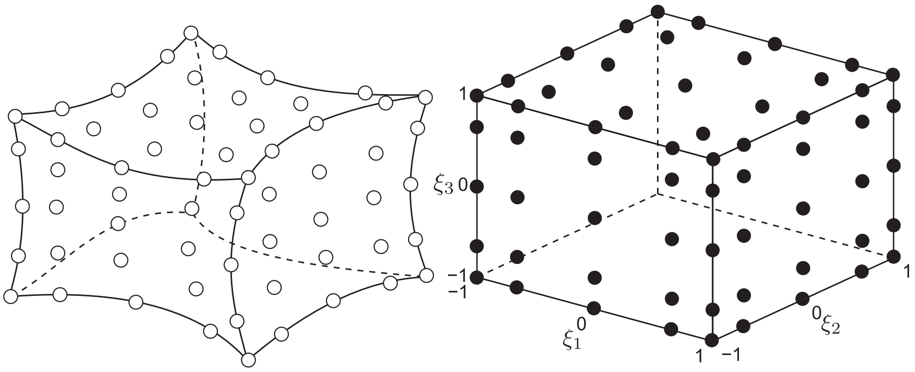


Fig. 1. Schematic diagram of a multistage excavation. The excavation stages are numbered.



(a) (b)

Fig. 2. (a) A typical spectral element with five nodes (open circles) in each dimension. (b) Spectral element mapped to its natural coordinates. The Gauss–Legendre–Lobatto points (solid black circles) are used for numerical integration. Only nodes on the three visible faces are shown here for clarity.

points in an element; it is given by the product of the number of GLL points in each dimension, i.e., $N = \prod_{j=1}^{3} N_j$. In the natural state with coordinates $\boldsymbol{\xi} = \{\xi_j\}$, the N_j GLL points are determined by the roots of the polynomial $(1-\xi^2)P'_n(\xi) = 0$, where P_n denotes the Legendre polynomial of degree $n = N_j - 1$. The interpolation functions ϕ^α in natural coordinates are determined by the tensor product of one-dimensional Lagrange polynomials

$$\phi_j^{\alpha_j}(\xi_j) = \prod_{\substack{\beta=1 \\ \beta \neq \alpha_j}}^{N_j} \frac{\left(\xi_j - \xi_j^\beta\right)}{\left(\xi_j^{\alpha_j} - \xi_j^\beta\right)}, \tag{6}$$

such that

$$\phi^\alpha(\boldsymbol{\xi}) = \prod_{j=1}^{3} \phi_j^{\alpha_j}(\xi_j), \tag{7}$$

where α is the index of a GLL point in a linear mapping of the 3D GLL points, corresponding to location $\{\alpha_1, \alpha_2, \alpha_3\}$.

For numerical integration, a point $\mathbf{x} = \{x_i\}$ in a deformed element is mapped to a point $\xi = \{\xi_j\}$ in the natural element, as illustrated in Fig. 2(a) and (b), using the transformation

$$\mathbf{x}(\boldsymbol{\xi}) = \sum_{\alpha=1}^{N_g} \mathbf{x}^\alpha \psi^\alpha(\boldsymbol{\xi}). \tag{8}$$

Here ψ^α denotes a shape function and N_g the number of geometrical nodes $\mathbf{x}^\alpha$ of an element. The Jacobian matrix of the transformation has elements determined by $J_{ij}(\xi) = \partial x_i(\xi)/\partial \xi_j$. The same GLL points are used as quadrature points for numerical integration.

Since the internal GLL points of a spectral element do not contribute to inter-element connectivity, these points can safely be excluded during interpolation of model geometry, i.e., N and N_g may differ. Depending on element type and numerical algorithm, fewer points may be sufficient to capture transformation (8). Therefore, we usually have $N > N_g$, and consequently the degree of the interpolating functions, ϕ^α, is greater than the degree of the shape functions, ψ^α, leading to a subparametric formulation.

The SEM is a continuous Galerkin method, in which the interpolation function ϕ^α is taken as the test function w_i. Upon substituting $w_i = \phi^\alpha$ and u_i, given by (5), in Eq. (3), we obtain a system of linear equations that may be written conveniently in the matrix–vector form

$$\mathbf{KU} = \mathbf{F}, \tag{9}$$

where $\mathbf{U}$ is the global displacement vector satisfying the continuity conditions. Similarly, $\mathbf{K} = \sum_e \mathbf{K}_e$ and $\mathbf{F} = \sum_e \mathbf{F}_e$, are known, respectively, as the global stiffness matrix and global force vector, where $\mathbf{K}_e$ and $\mathbf{F}_e$ are the elemental stiffness matrix and elemental force vector expanded to the same sizes as $\mathbf{K}$ and $\mathbf{F}$, respectively, but having entries only in locations corresponding to the eth element. Symbolically, we write

$$\begin{aligned} \mathbf{K}_e &= \int_{\Omega_e} \mathbf{B}^T \mathbf{C} \mathbf{B} \, \mathrm{d}\Omega, \\ \mathbf{F}_e &= \int_{\Omega_e} \boldsymbol{\Phi}^T \mathbf{f} \, \mathrm{d}\Omega + \int_{\Gamma_e} \boldsymbol{\Phi}^T \hat{\mathbf{t}} \, \mathrm{d}\Gamma, \end{aligned} \tag{10}$$

where a superscript T denotes the transpose, Ω_e the volume, and Γ_e the boundary of an element. The quantities $\boldsymbol{\Phi}$, $\mathbf{B}$, and $\mathbf{C}$ are known, respectively, as the interpolation function matrix, the strain–displacement matrix, and the elasticity matrix.

2.2. Excavation load

During excavation, a newly excavated surface becomes a traction-free surface. Therefore, a load equal and opposite to the load contributed by the self weight and stress state of the excavated region has to be applied on excavated surfaces. This excavation load is computed using only excavated elements, and it may be expressed as (e.g., [16])

$$\mathbf{F}_e^{\mathrm{ex}} = \int_{\Omega_e} \mathbf{B}^T \boldsymbol{\sigma}_0 \, \mathrm{d}\Omega - \int_{\Omega_e} \boldsymbol{\Phi}^T \mathbf{f}^{\mathrm{gr}} \, \mathrm{d}\Omega, \tag{11}$$

where $\mathbf{f}^{\mathrm{gr}}$ represents the force due to gravity given by $\{0, 0, -\gamma_e\}$, where γ_e denotes a unit weight of material on the eth element. Similarly, $\boldsymbol{\sigma}_0$ represents the initial stress in an element before an excavation stage. If the initial stress before excavation is unknown, it may be approximated using an overburden pressure determined by (e.g., [70])

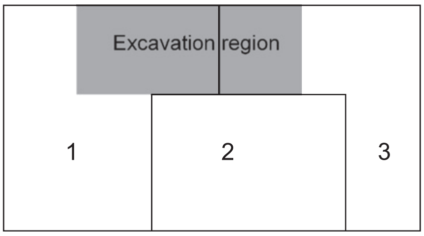


Fig. 3. Illustration of three partitions (numbered 1, 2, and 3) of an excavation model. Black lines mark boundaries of partitioned subdomains. The excavation region is indicated by the shaded rectangle and involves subdomains 1 and 3. The resulting load affects all three subdomains.

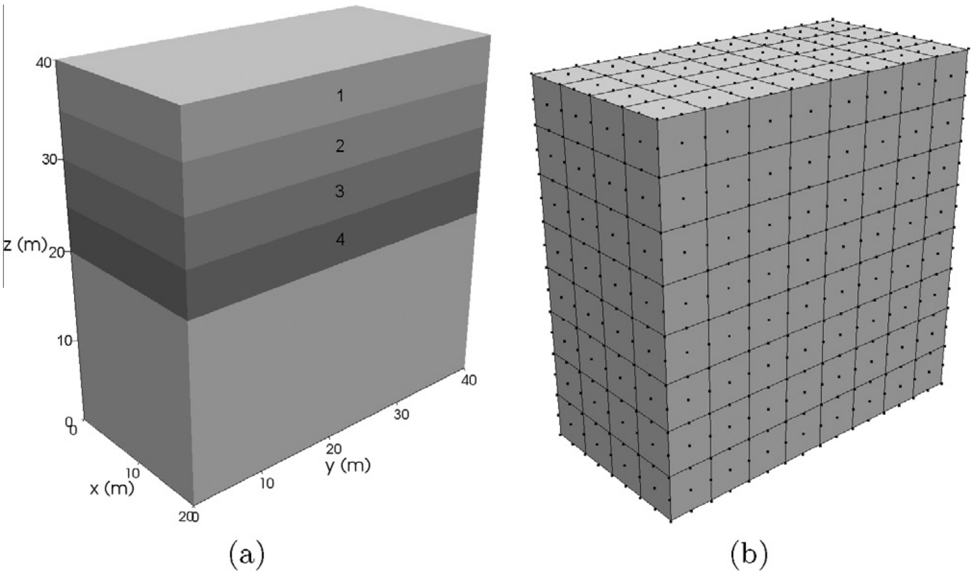


Fig. 4. (a) 1D excavation model with four excavation stages (numbered 1, 2, 3, and 4). (b) Spectral-element mesh of the model with three GLL points in each dimension. The total number of elements is 256.

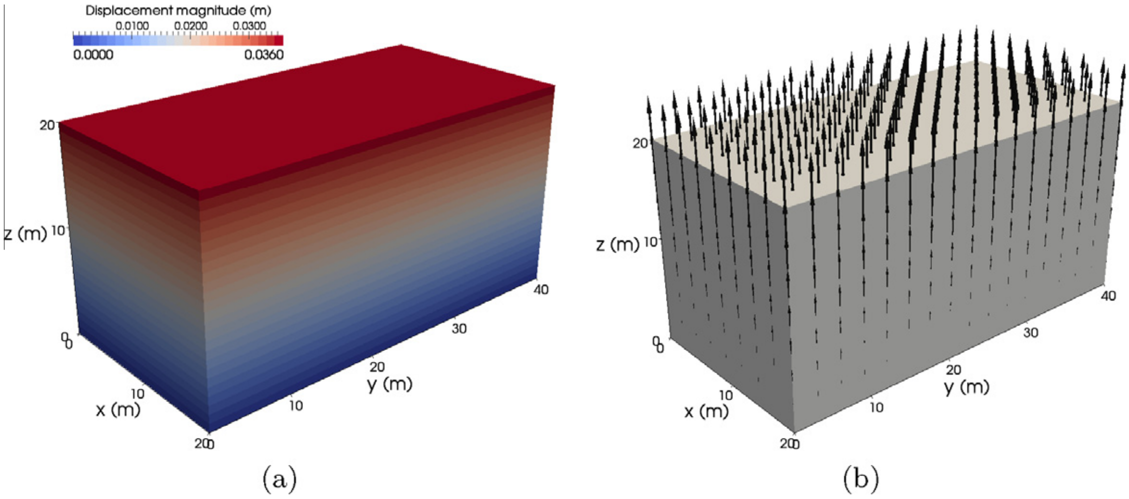


Fig. 5. (a) Displacement field and (b) Displacement vectors at the final stage. Results for single – and multistage simulations were identical.

$$\sigma_v(z_1) = \sigma_v(0) + \int_0^{z_1} \gamma(z)\mathrm{d}z, \tag{12}$$

where $\gamma(z)$ denotes a unit weight of material at depth z. Horizontal stress is often computed using the simple relation $\sigma_h = K_0\sigma_v$, where K_0 is the at-rest lateral earth pressure coefficient. For a simple model, overburden pressure can easily be computed using relation (12). However, it is difficult to use this relation for complex and heterogeneous models. In such cases, we may use the spectral-element method itself to compute the stress field before excavation, taking into account the appropriate boundary conditions. This enables computation of initial stress fields in complex models.

2.3. Elastoplastic failure

For elastoplastic materials, we implement a Mohr–Coulomb yield criterion with a non-associated flow rule. The Mohr–Coulomb yield criterion may be expressed as (e.g., [5])

$$F = \sigma_m \sin\phi + \bar{\sigma}\left(\cos\theta - \frac{1}{\sqrt{3}}\sin\phi\sin\theta\right) - c\cos\phi, \tag{13}$$

where σ_m, $\bar{\sigma}$, and θ are the stress invariants known as the mean stress, deviatoric stress, and Lode theta, respectively. The parameters c and ϕ represent the cohesive strength and internal friction angle of the material, which we assume to be elastic-perfectly plastic.

We solve the elastoplastic problem in an iterative manner using an initial strain method (e.g., [71]). In this method, the material is allowed to sustain stresses outside the failure envelope for a finite period, and plastic strain is computed using a concept of pseudo-viscosity. Hence the method is also referred to as the 'viscoplastic strain method' (e.g., [16]). The method attempts to satisfy non-linear behavior by successively correcting loads and solving linear system (9) using constant stiffness

$$\mathbf{K}\mathbf{U}^k = \mathbf{F} + (\mathbf{F}^p)^k. \tag{14}$$

The force term $(\mathbf{F}^p)^k$ at each iteration k depends on the force term at the previous iteration $k-1$ and incremental plastic strain $(\delta\boldsymbol{\varepsilon}^p)^k$, and is given by

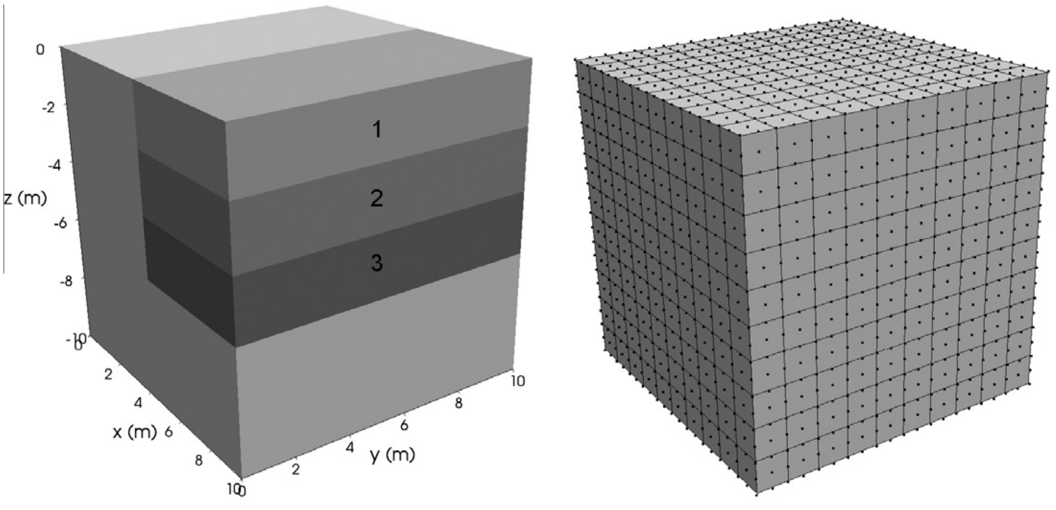
$$(\mathbf{F}^{\mathrm{p}})^k = (\mathbf{F}^{\mathrm{p}})^{k-1} + \sum_{\text{elements}} \int_{\Omega_e} \mathbf{B}^T \mathbf{C} (\delta \boldsymbol{\varepsilon}^{\mathrm{p}})^k \mathrm{d}\Omega. \tag{15}$$

The force contributed by the incremental plastic strain is self-equilibrating so that net loading remains the same. This load is accumulated in successive iterations until convergence is achieved. Convergence is measured as $\|\mathbf{U}^k - \mathbf{U}^{k-1}\| / \|\mathbf{U}^{k-1}\| \leqslant \epsilon$, where ϵ is a tolerance.

3. Numerical implementation and parallelization

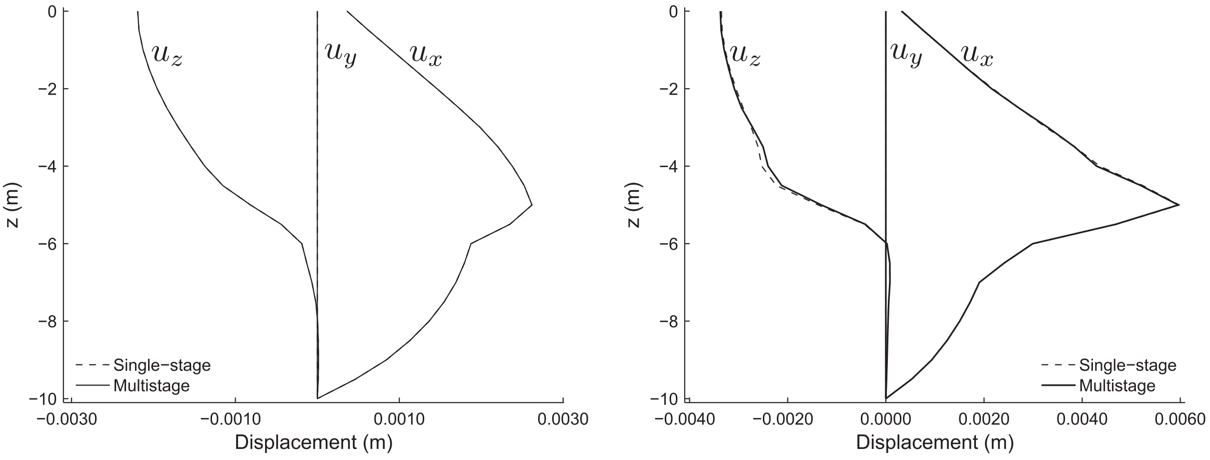
We implement an efficient and robust strategy for the simulation of multistage excavation. Surface traction due to excavated regions is computed solely based on excavated elements. The equations are formed only for active degrees of freedom corresponding to intact nodes, similar to the strategy of Comodromos et al. [14] and Smith and Griffiths [16]. The stiffness matrix is computed and stored only for intact elements. In order to solve the linear equations, we use an element-by-element preconditioned conjugate-gradient method, which is an iterative solver widely used in the classical FEM (e.g., [4]). Both conjugate gradient and nonlinear iteration loops only involve intact elements. Since the numbers of excavated and intact elements change at each excavation stage, we use dynamic-memory allocation to manage memory efficiently.

In order to handle large-scale problems, the software is parallelized based on a non-overlapping domain decomposition method (e.g., [72,73]). In this method, the mesh is divided into a number of subdomains. No subdomains share elements, and only nodes located on common subdomain interfaces are shared. We use the parallel library MPI to facilitate communication across processors. For efficient parallel processing, elements should be approximately equally distributed among subdomains, and additionally the number of nodes on subdomain interfaces should be minimal. Consequently, efficient mesh partitioning is required. There are some open-source tools for both serial and parallel graph partitioning, e.g., SCOTCH [74] and METIS [75]; in this study we use SCOTCH. To simulate multistage excavation, we implement a fixed-partition



(a) (b)

Fig. 6. (a) 2D excavation model with three excavation stages (numbered 1, 2, and 3). (b) Spectral-element mesh of the model with three GLL points in each dimension. The total number of elements is 1000.



(a) (b)

Fig. 7. (a) Displacement profile along a vertical line at $x = 5$ m and $y = 5$ m for an elastic material. (b) Same as (a) but for an elastoplastic material. The displacement profiles for the elastic material are identical for single – and multistage simulations.

strategy, i.e., the domain is partitioned once and for all before excavation begins. Since the excavated portion may involve any number of subdomains, the excavation load should be computed and distributed carefully. For example, Fig. 3 illustrates an excavation in a fixed-partition model with three subdomains. The excavation region lies in partitions 1 and 3, and hence excavation loads are computed only in these partitions. Since processing involves only intact elements, some of the excavation loads are unused by partitions 1 and 3 although these are necessary for partition 2. Therefore, these unused loads are distributed to neighboring active partitions, partition 2 in this example.

The fixed-partition strategy is relatively simple to implement. Once preprocessing is completed and communication topology for parallel processing is determined, modification of preprocessing and communication topology at subsequent excavation stages is straightforward. However, load-balancing may not always be perfect because the excavated region can involve any number of partitions, and the excavated portion in each partition may not be equivalent. Alternatively, we may repartition the intact domain at each excavation stage, but this requires preprocessing and determination of the communication topology at every stage. In this article, we use only the fixed-partition strategy for all examples dealing with parallel processing.

We use a parallel preconditioned conjugate-gradient solver to be consistent with the non-overlapping domain decomposition scheme. This solver has previously been implemented in both overlapping and non-overlapping domain decomposition methods (e.g., [76–80]). In the domain decomposition method, each processor solves its own part of the system, occasionally communicating across processors to assemble entities along common interfaces. Depending on the boundary conditions, load balancing during the solution procedure may not be ideal, even if the mesh is evenly decomposed. One can also repartition the degrees of freedom so that close to perfect load-balancing is achieved during the solution procedure (e.g., [81]). However, the communication topology for parallel processing may be complicated. In the future, other algorithms which provide more data locality, e.g., localized ILU preconditioning (e.g., [82]), could be of interest.

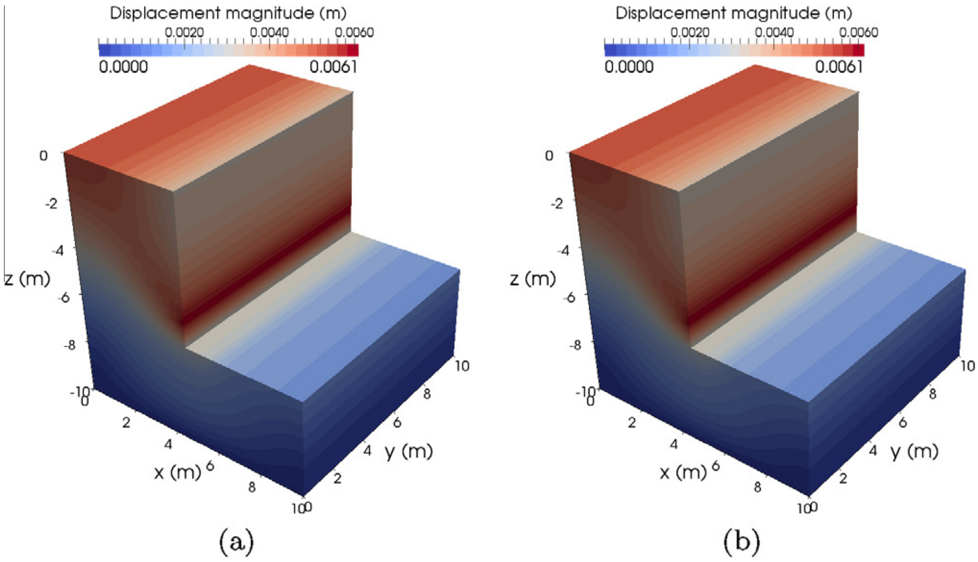


Fig. 8. Displacement field at the final stage for (a) three-stage excavation and (b) single-stage excavation. The displacement field in each figure is independently scaled to its displacement range.

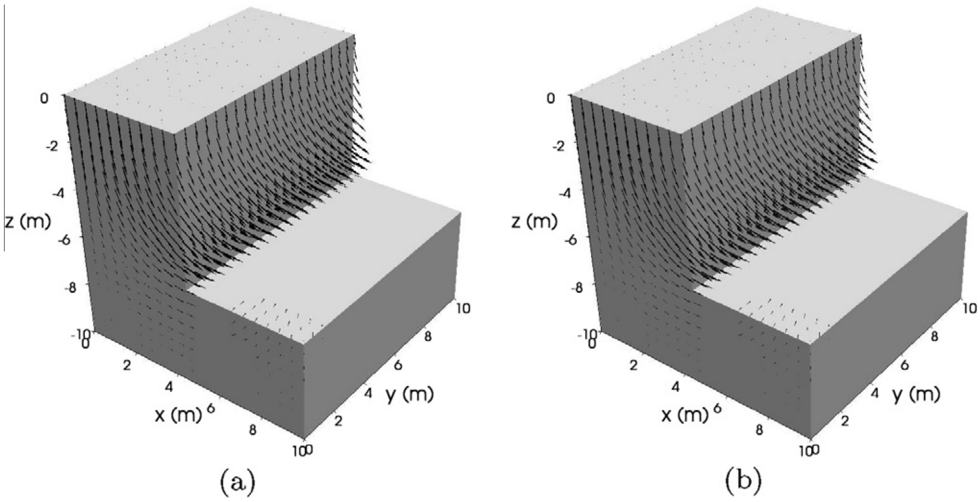


Fig. 9. Displacement vectors at the final stage for (a) three-stage excavation and (b) single-stage excavation. The vector field in each figure is independently scaled to its displacement range.

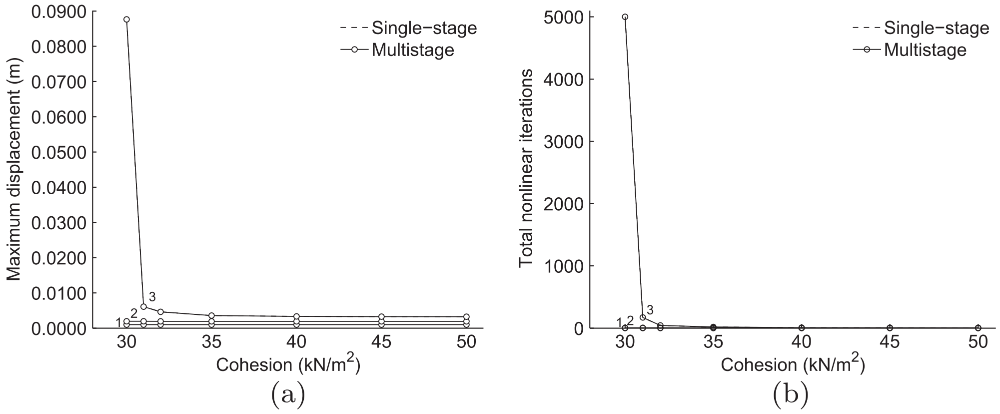


Fig. 10. (a) Maximum displacement and (b) total number of nonlinear iterations at three excavation stages for various cohesion values. The excavation stages are numbered 1, 2, and 3.

4. Numerical results

4.1. Example 1: 1D excavation in a linear elastic medium

In the first example, we take a simple four-stage excavation model with a total excavation height of 20 m (Fig. 4(a)). The model consists of a linear elastic material with Young's modulus $E = 10^4$ kN/m^2, unit weight $\gamma = 1$ kN/m^3, Poisson's ratio $\nu = 0.2$, and at-rest pressure coefficient $K_0 = 0.5$. For the boundary conditions, nodes located on the bottom surface are fixed in both directions, and nodes located on side surfaces are fixed only along the normals to the corresponding surfaces. Due to the boundary conditions and excavation geometry of this particular model, displacement depends only on the z dimension. Therefore, this example is equivalent to a 1D problem. This particular model has been used by several authors to validate their excavation algorithms (e.g., [7,8,14]). The analytical solution for this problem gives an upward displacement of 0.036 m on the excavated surface at the final stage (e.g., [14]).

Since the model has a simple geometry, meshing is straightforward. Here, we use the mesh generation tool kit CUBIT [37]. The generation of spectral elements with more than three GLL points in each dimension is currently not possible directly within CUBIT. Therefore, we first create an 8-node hexahedral mesh with CUBIT, and the mesh is subsequently converted to a spectral-element mesh with the desired number of GLL points within our main program. For this example, we use three GLL points in each dimension, resulting in a total of 27 nodes per element. We mesh the model with an element size of 5 m, resulting in a total of 256 elements (Fig. 4(b)). Based on this mesh, we perform two simulations using our serial SEM program. In the first simulation, four layers are excavated sequentially in four stages. In the second simulation, four layers are excavated in a single stage. We set a relative tolerance of 10^{-8} for conjugate-gradient iterations.

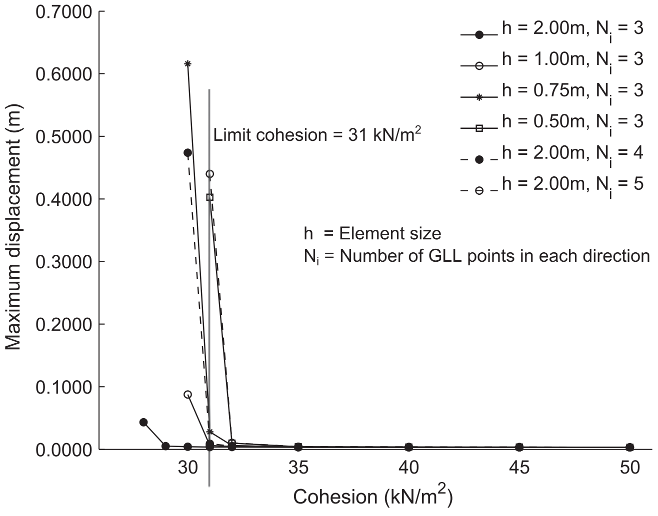


Fig. 11. Maximum displacement at the final stage of excavation for different degrees of h- and p-refinement computed for a range of cohesion values. The displacement for the lowest cohesion value for each curve represents the non-converged solution.

With both simulations, we obtain the same 0.036 m upward displacement on the excavated surface (Fig. 5(a) and (b)). This displacement is in perfect agreement with the analytical solution. As expected, we obtain the same displacement field and displacement vectors in the entire domain for single- and multistage excavations. Hence, the final solution is independent of the excavation sequence, thereby verifying the uniqueness principle for linear elastic materials.

4.2. Example 2: 2D excavation

In this example, we consider a three-stage excavation model with a total excavation height of 6 m (Fig. 6(a)). The material has unit weight 20 kN/m^3, Young's modulus 10^5 kN/m^3, cohesion

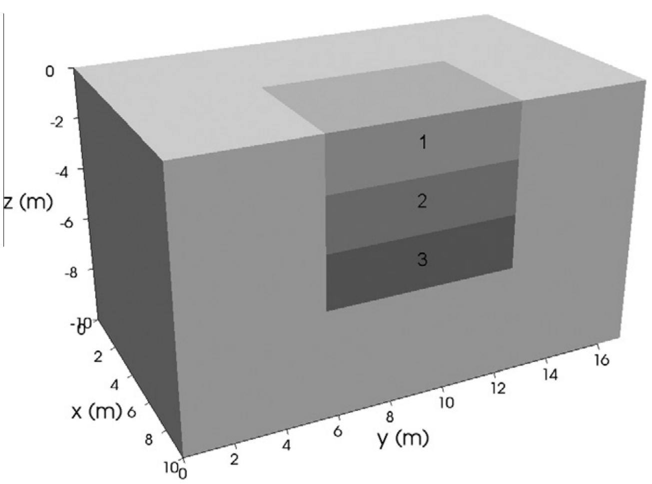


Fig. 12. Excavation model with three excavation stages (numbered 1, 2, and 3).

31 kN/m^2, friction angle 0°, and dilation angle 0°. For initial stress, we consider only the overburden pressure, which we compute during the initial stage of the simulation using the SEM itself. We use the same boundary conditions as in the previous example. Due to the boundary conditions and the excavation geometry of this particular model, the displacement depends only on the x and z coordinates. Therefore, this example is equivalent to a 2D plane strain problem.

We mesh the model with an element size of 1 m, resulting in a total of 1000 elements. For the elastoplastic material we need to solve nonlinear constitutive equations. Therefore, we use a finer mesh than in the previous example to capture nonlinear behavior of the model. We use three GLL points in each dimension. We set relative tolerances of 10^{-8} for conjugate-gradient iterations and 10^{-5} for nonlinear iterations. Similarly, we set the maximum number of nonlinear iterations to 5000. We perform four different simulations using the serial SEM program: (1) single-stage excavation in an elastic material, (2) multistage excavation in an elastic material, (3) single-stage excavation in an elastoplastic material, and (4) multistage excavation in an elastoplastic material. Fig. 7(a) and (b) show displacement profiles along a vertical line in the middle of the vertical excavated surface. As expected, profiles for single – and multistage simulations computed in the elastic medium are identical. We observe small discrepancies between the profiles of single – and multistage simulations computed in the elastoplastic medium. Although the displacement field and displacement vectors appear similar for single- and multistage simulations in an elastoplastic medium, we observe small numerical discrepancies of order $\sim 1.4 \times 10^{-4}$ m.

One purpose of the simulation of the excavation process is to assess the stability of structures. We can estimate the limiting strength of the material at which the model collapses during the excavation. In order to estimate this limiting strength (i.e., cohesion in our example), we compute displacement fields for a range of cohesion values. Fig. 10(a) and (b) show the resulting maximum displacement and required number of nonlinear iterations, respectively. We observe a small displacement for excavation stages 1 and 2. Relatively few iterations are required for convergence at these stages, and the model exhibits mostly elastic behaviour for all strengths. For excavation stage 3, we observe small displacements until the strength reaches approximately 32 kN/m^2, and the corresponding nonlinear iterations are relatively few. At strength 31 kN/m^2 the displacement begins to increase noticeably, requiring a larger number of nonlinear iterations. The displacement suddenly increases at 30 kN/m^2, and it does not converge within the given maximum number (i.e., 5000) of nonlinear itera-

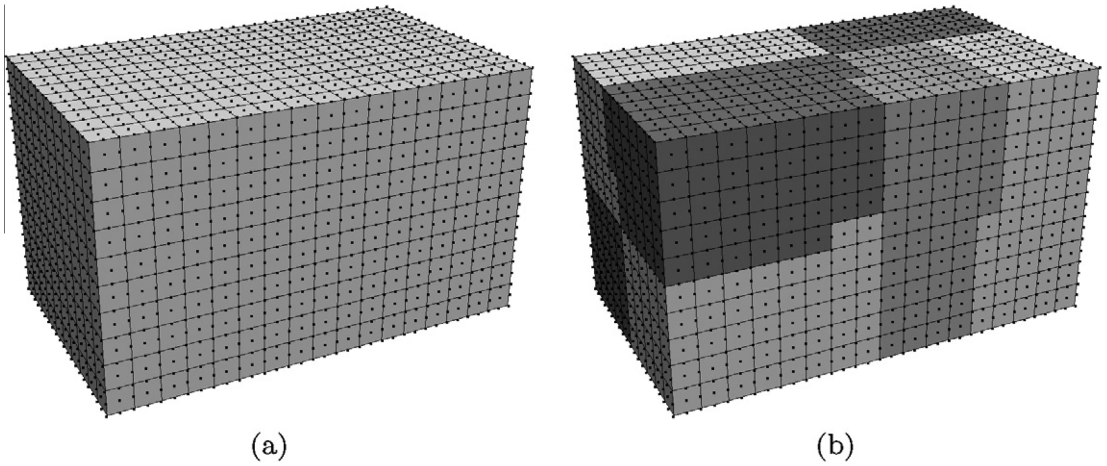

(a) (b)

Fig. 13. (a) Spectral-element mesh of a model with three GLL points in each dimension. (b) Same as (a), but partitioned into eight subdomains for parallel processing. The total number of elements is 1700.

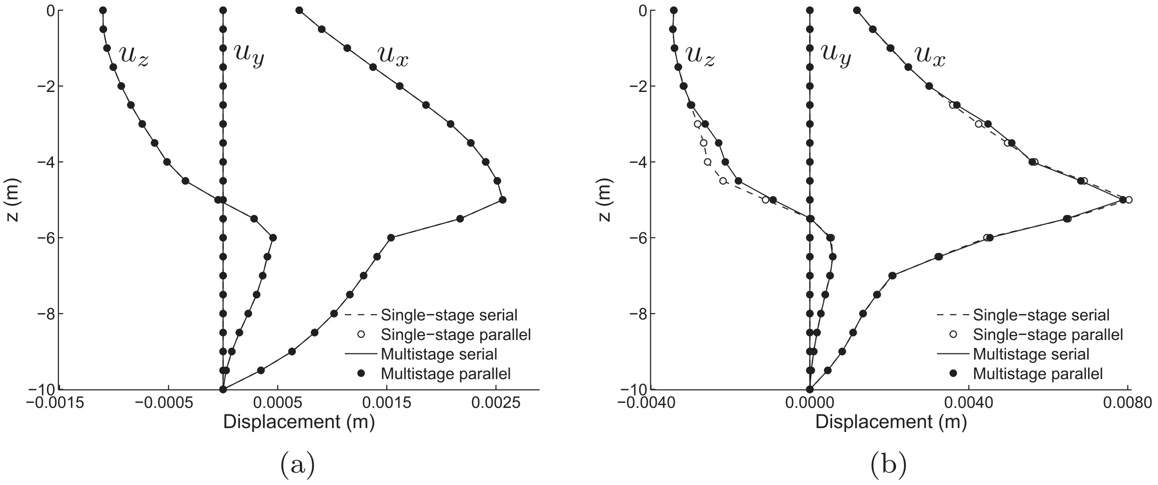


Fig. 14. (a) Displacement profile along a vertical line at x = 10 m and y = 5 m for an elastic material. (b) Same as (a) but for an elastoplastic material. The displacement profiles for the elastic material are identical for single – and multistage simulations.

tions. This indicates a possible collapse of the model, and hence the limiting strength of the model is ~30 kN/m^2. For the plane strain condition, the classical-limit plastic solution (e.g., [83,84]) gives a value of limiting cohesion of approximately 31 kN/m^2 for an excavation height of 6 m, which is in good agreement with the computed result. The maximum displacement and total number of nonlinear iterations appear similar for single – and multistage excavations with a maximum discrepancy of ~5.8×10^{-5} m in displacement magnitude (Fig. 10(a) and (b)). We observe a circular failure pattern after the final excavation stage, which is similar for single- and multistage excavations (Fig. 8(a) and (b)) (see Fig. 9).

Although it is difficult to compute accurate displacements during collapse, we conduct multistage simulations for elastoplastic

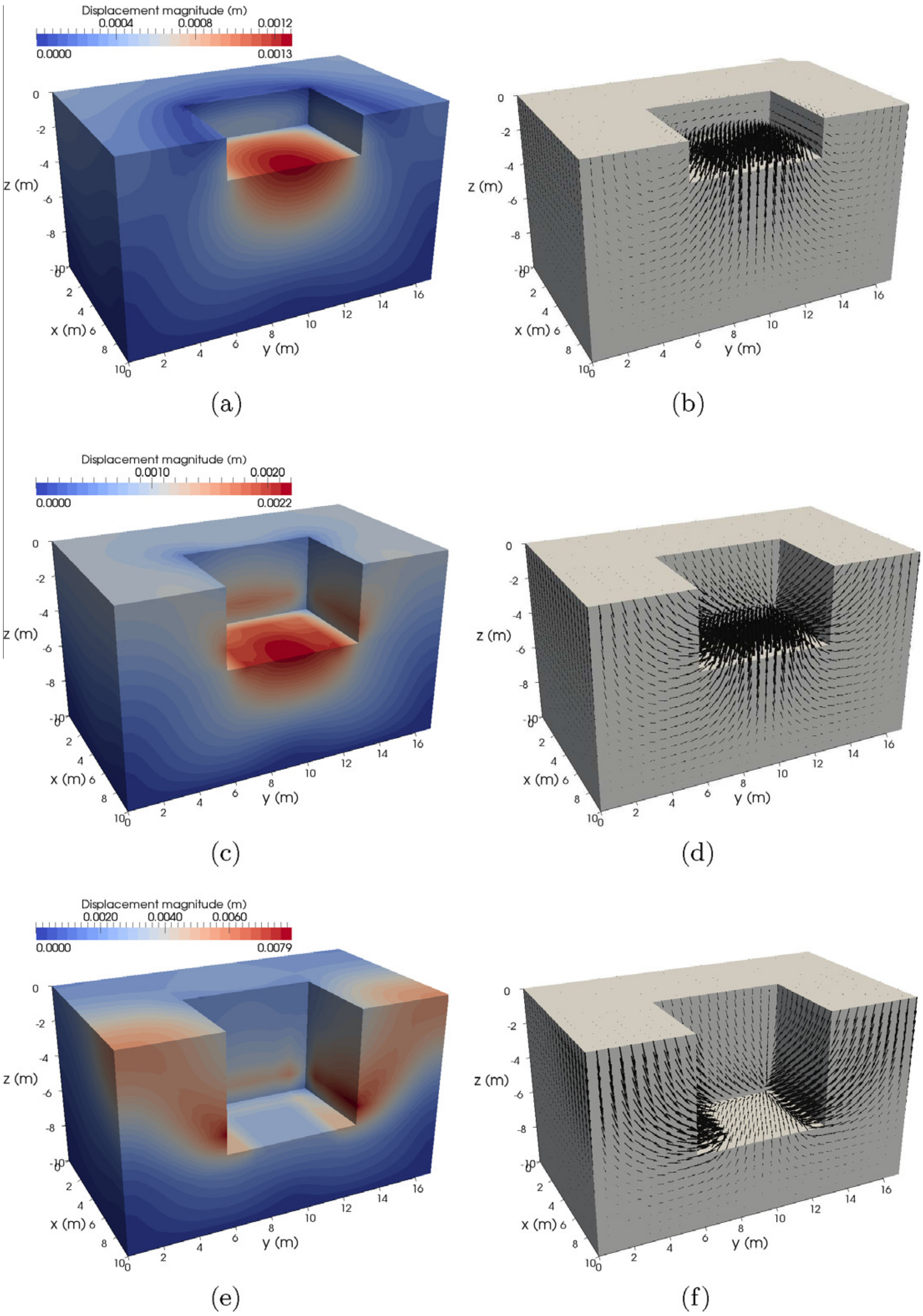


Fig. 15. Displacement fields (left column) and displacement vectors (right column) computed at three excavation stages (top to bottom: stages 1, 2 and 3). The displacement and vector fields in each figure are independently scaled to their range. The results obtained by the serial and parallel programs are identical.

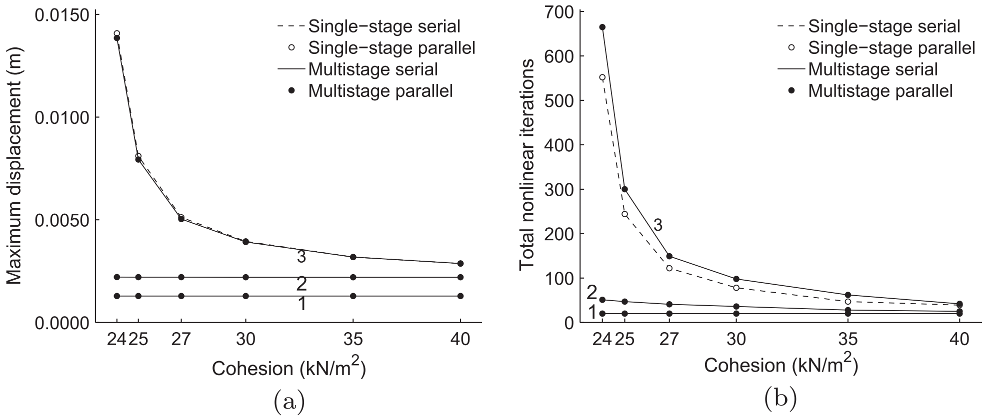


Fig. 16. (a) Maximum displacement and (b) total number of nonlinear iterations at three excavation stages (numbered 1, 2, and 3) for various cohesion values.

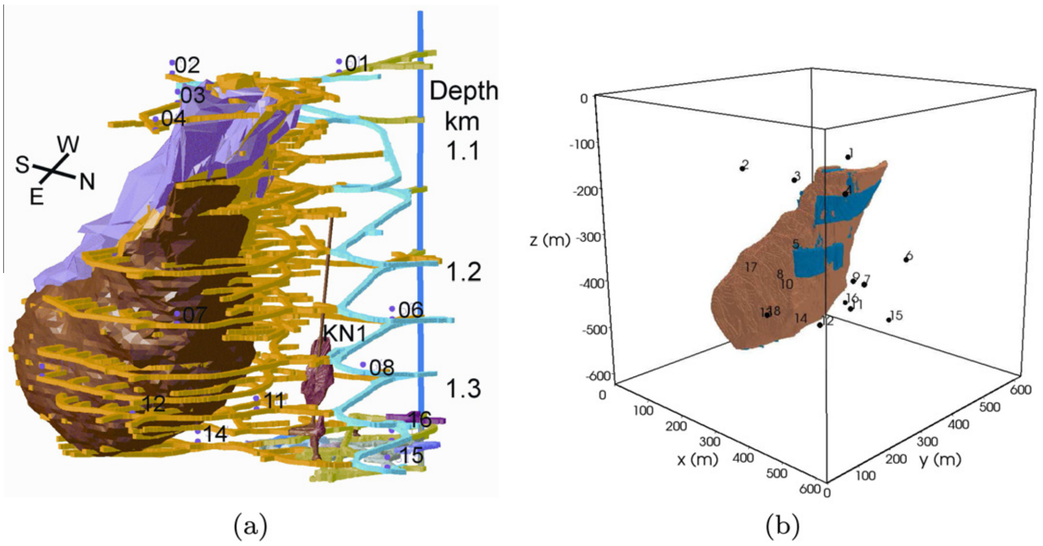


Fig. 17. (a) Pyhäsalmi mine with surrounding infrastructure: copper/zinc ore body (brown/pink), access tunnels (yellow), elevator shaft (dark blue), and seismic stations (numbered). The passage for quarried ore is marked by KN1. (b) 3D wave-speed model (see Table 1) of the Pyhäsalmi mine: stopes, i.e., mined-out cavities (blue) and ore body (brown). The remainder is host rock. Geophones are indicated by black dots and numbered. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

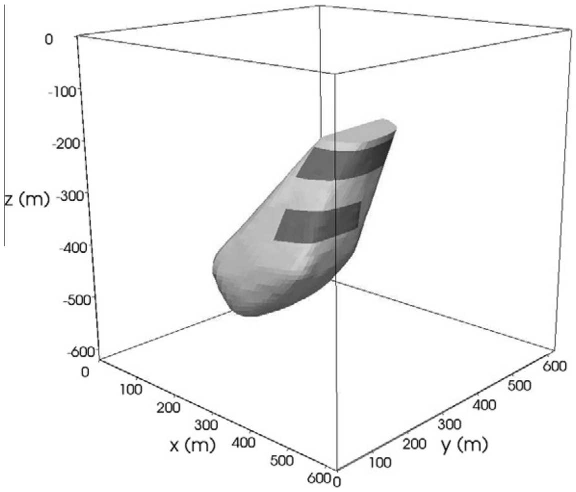


Fig. 18. Simplified 3D model of the Pyhäsalmi mine, including only the ore body (solid) and two major stopes (black).

media considering different degrees of h – and p-refinements of the mesh and monitor how the SEM results behave. We discretize the model with four different element sizes of approximately 2 m, 1 m, 0.75 m, and 0.5 m, resulting in a total number of elements of 150, 1000, 2548, and 8000, respectively. First, we perform four complementary simulations of the multistage excavation based on these four meshes, using three GLL points in each dimension. Second, we perform two complementary simulations with the coarsest mesh (element size 2 m) using, respectively, four and five GLL points in each dimension. In each simulation, we compute the displacement field for a range of cohesion values, successively

Table 1
Material properties of the Pyhäsalmi mine model.

	Host rock	Ore body
Mass density (kg/m^3)	2000	4400
P-wave speed (m/s)	6000	6300
S-wave speed (m/s)	3460	3700
Cohesion (MPa)	14	14
Friction angle (°)	39	39
Dilation angle (°)	0	0

decreasing its value until the solution fails to converge within the fixed maximum number of nonlinear iterations. Fig. 11 summarizes the results of these simulations. With the coarsest mesh, using three GLL points, we observe a sudden increase in maximum displacement at a cohesion of 28 kN/m^2, and the solution does not converge thereafter. This gives an approximate limiting cohesion of 28 kN/m^2, which is ~10% less than the correct limiting strength of 31 kN/m^2. With a finer mesh of element size 1 m, we obtain an approximate limiting cohesion of 30 kN/m^2, which is ~3% less than the actual value. As we further refine the mesh, the limiting cohesion converges to the correct value. On the other hand, with the coarsest mesh and four GLL points, we obtain an approximate limiting cohesion of 31 kN/m^2, in agreement with the actual value. The results involving the coarsest mesh and four or five GLL points are similar to those obtained based on finer meshes with element sizes of 0.75 m and 0.5 m, respectively.

4.3. Example 3: 3D excavation

In the third example, we simulate single- and multistage excavation in the model shown in Fig. 12. We use the same material properties as in previous cases except for the value of cohesion, which is 25 kN/m^2. We again consider only overburden pressure as initial stress and use the same boundary conditions as in the previous examples. Unlike in previous examples, the displacement in this model depends on all three (i.e., x, y, and z) coordinates. The model is meshed using an average element size of 1 m, resulting in a total of 1700 elements (Fig. 13(a)). In order to validate our parallel program, we also partition this mesh into eight subdomains using the graph partitioning tool SCOTCH [74] for parallel processing (Fig. 13(b)).

We perform four different simulations with each of the serial and parallel SEM programs: (1) single-stage excavation in an elastic material, (2) multistage excavation in an elastic material, (3) single-stage excavation in an elastoplastic material, and (4) multistage excavation in an elastoplastic material. We use three GLL points in each dimension and set the same relative tolerance for conjugate gradient and nonlinear iterations, and the same maximum number of nonlinear iterations as in previous cases. For the elastoplastic case, we apply the excavation load in ten increments. Fig. 14(a) and (b) show displacement profiles along a vertical line on the front-left vertical excavated corner. As expected, profiles for single- and multistage simulations computed in the elastic medium are identical. We observe small discrepancies between

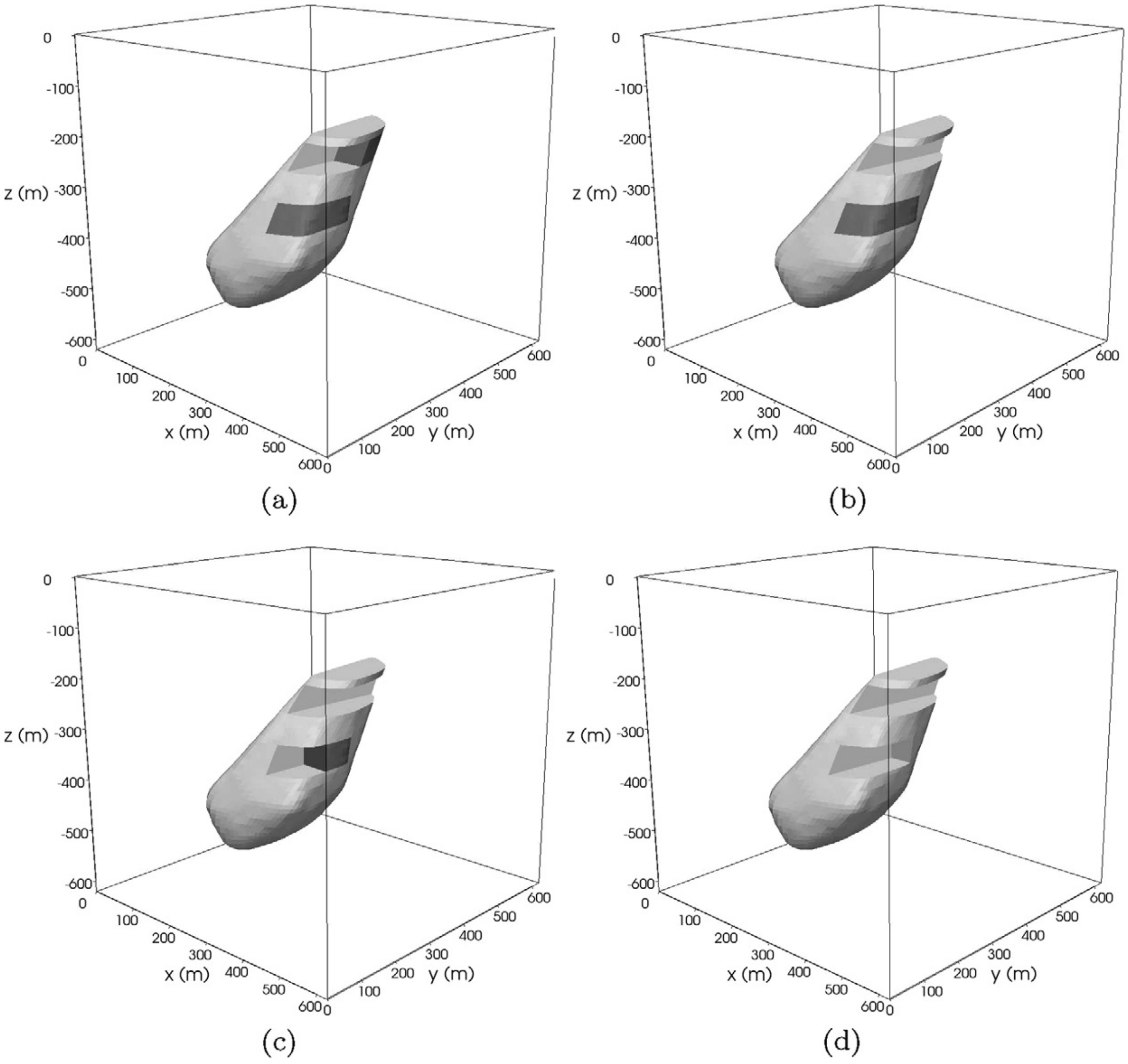


Fig. 19. Four excavation stages in the Pyhäsalmi mine model. (a) Stage 1. (b) Stage 2. (c) Stage 3. (d) Stage 4.

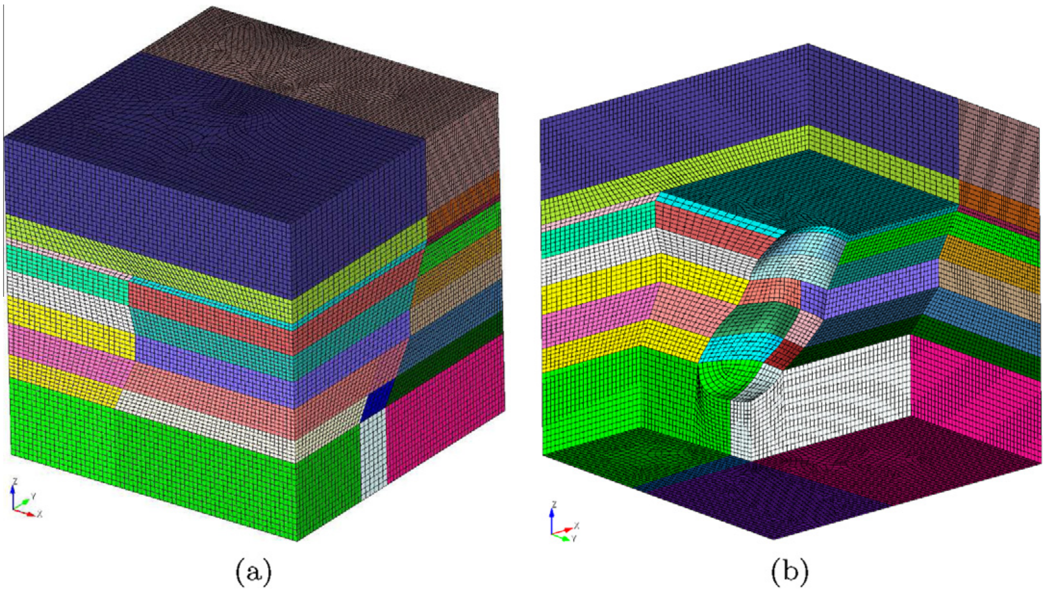


Fig. 20. (a) Spectral-element mesh for a 3D model of the Pyhäsalmi mine. (b) Interior section of the mesh visualizing the ore body. Colours represent the different volumes created for meshing.

profiles of single- and multistage simulations in the elastoplastic medium. Results computed by the serial and parallel programs are in excellent agreement.

Fig. 15(a)–(f) show the resulting displacement fields and displacement vectors obtained from the multistage simulation in an elastoplastic medium. We obtain a maximum numerical discrepancy on the order of $\sim 4.5 \times 10^{-4}$ m between single- and multistage displacement fields. During the first excavation stage, the excavation height is small and, therefore, upward displacement of the floor is dominant. As the excavation height increases, displacements on the side walls become dominant and hence failure may occur.

Using the same model, we compute displacements for a range of cohesion values considering both single- and multistage excavations. We use both serial and parallel programs. Fig. 16(a) and (b) show the resulting maximum displacements and corresponding number of nonlinear iterations, respectively, observed at different excavation stages. Both excavation stages 1 and 2 result in a small displacement requiring very few nonlinear iterations. During these stages the model behaves mostly as an elastic material. On the other hand, during excavation stage 3, as cohesion decreases, the model undergoes plastic deformation, requiring larger numbers of nonlinear iterations to converge. Maximum displacements computed by the serial and parallel programs are in perfect agreement for all cohesion values (Fig. 16(a)), and the number of nonlinear iterations is also similar (Fig. 16(b)). Displacements computed for single-stage excavation are similar to those for multistage excavation, but some discrepancies are observed after the model undergoes plastic deformation. The required number of nonlinear iterations is different compared to the multistage excavation case. For this excavation model, the limiting value of cohesion is estimated to be ~ 24 kN/m^2 (Fig. 16(a) and (b)).

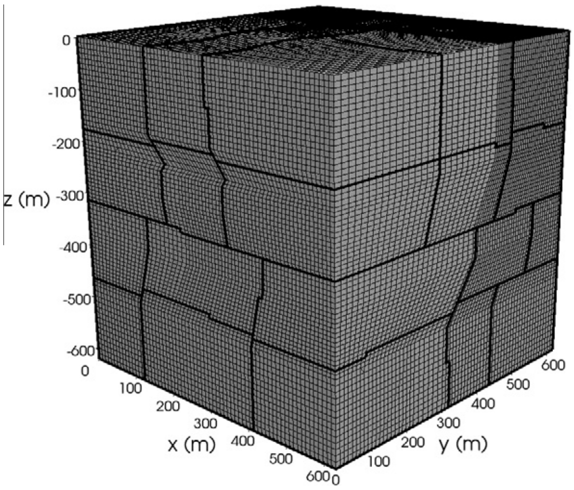


Fig. 21. Spectral-element mesh of a 3D model of the Pyhäsalmi mine partitioned into 32 subdomains. Thick black lines represent subdomain interfaces.

4.4. Example 4: Excavation in a mine

In this example, we apply our parallel program to an underground ore mine, namely the Pyhäsalmi mine in central Finland. This mine consists of a volcanogenic massive sulphide (VMS) deposit, and produces mainly copper, zinc, and pyrite concentrates. The

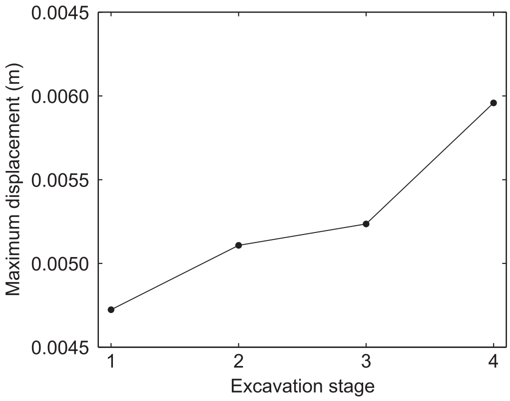


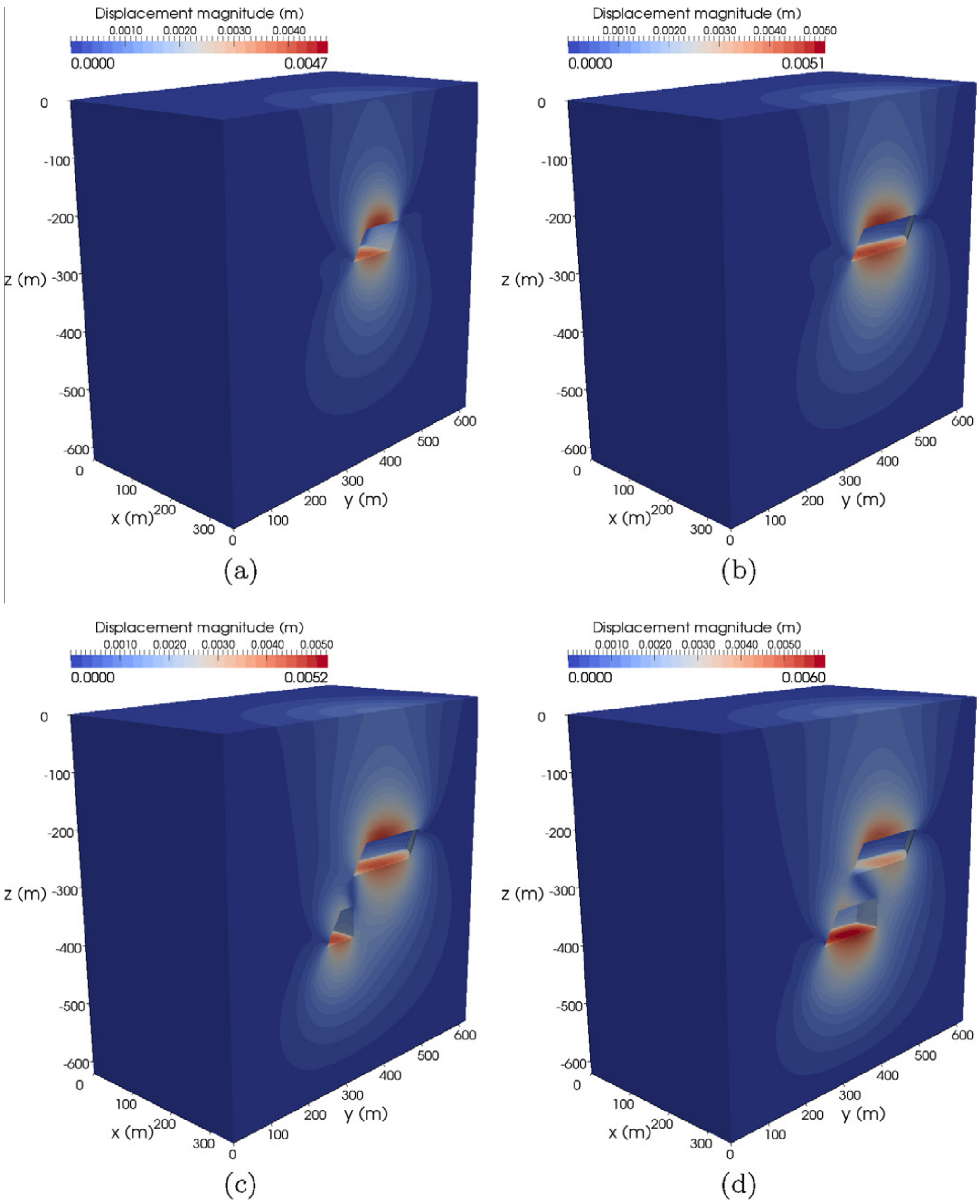
Fig. 22. Maximum displacement at four excavation stages. The solutions at all stages converge in one iteration per load increment.

copper-zinc ore body in the mine extends down to a depth of ~1.4 km (Fig. 17(a)), and microearthquakes are frequently observed (e.g., [85]). The in-mine seismic network comprises eighteen geophones [86] that are used to record and locate microearthquakes induced by mining operations. Fig. 18 illustrates the 3D wave-speed model used for the simulation of seismic wave propagation (see e.g., [87]), which consists of an ore body, host rock, and stopes (i.e., mined out voids) (Table 1).

To simulate multistage excavation, we estimate Young's modulus and Poisson's ratio from the seismic properties summarized in Table 1 (e.g., [88]). The estimated values of Young's moduli for the host rock and the ore body are approximately 60 GPa and 149 GPa, respectively. The approximate Poisson's ratios for the ore body and the host rock are 0.25 and 0.24, respectively. We consider only overburden pressure as initial stress, computed within the SEM program before the excavation loop begins.

Due to the complex structure of, in particular, mined-out cavities, it is difficult to generate a hexahedral mesh for the complete model (Fig. 17(b)). Therefore, we simplify the original model and consider only two major stopes (Fig. 18). We assume four excavation stages, as shown in Fig. 19(a)–(d). Unlike previous examples, there are two different excavation regions within the model. Each region is excavated in two stages. We assume that the four excavation stages are performed sequentially, as shown in Fig. 19(a)–(d). For this example, we use the same boundary conditions as in the previous examples.

Even with this simplified model, generation of a hexahedral mesh with CUBIT is nontrivial. To be able to generate a high-quality mesh, we need to decompose the complex geometry into meshable volumes. We divide this particular model into 78 volumes that can be meshed with the functionalities available in CUBIT (Fig. 20(a)). We mesh the model with an average element size of 9.5 m for the



(a) (b) (c) (d)

Fig. 23. Displacement fields computed at four excavation stages, visualized in the section taken at $x = 350$ m. (a) Stage 1. (b) Stage 2. (c) Stage 3. (d) Stage 4. Displacement fields in each figure are independently scaled to their range.

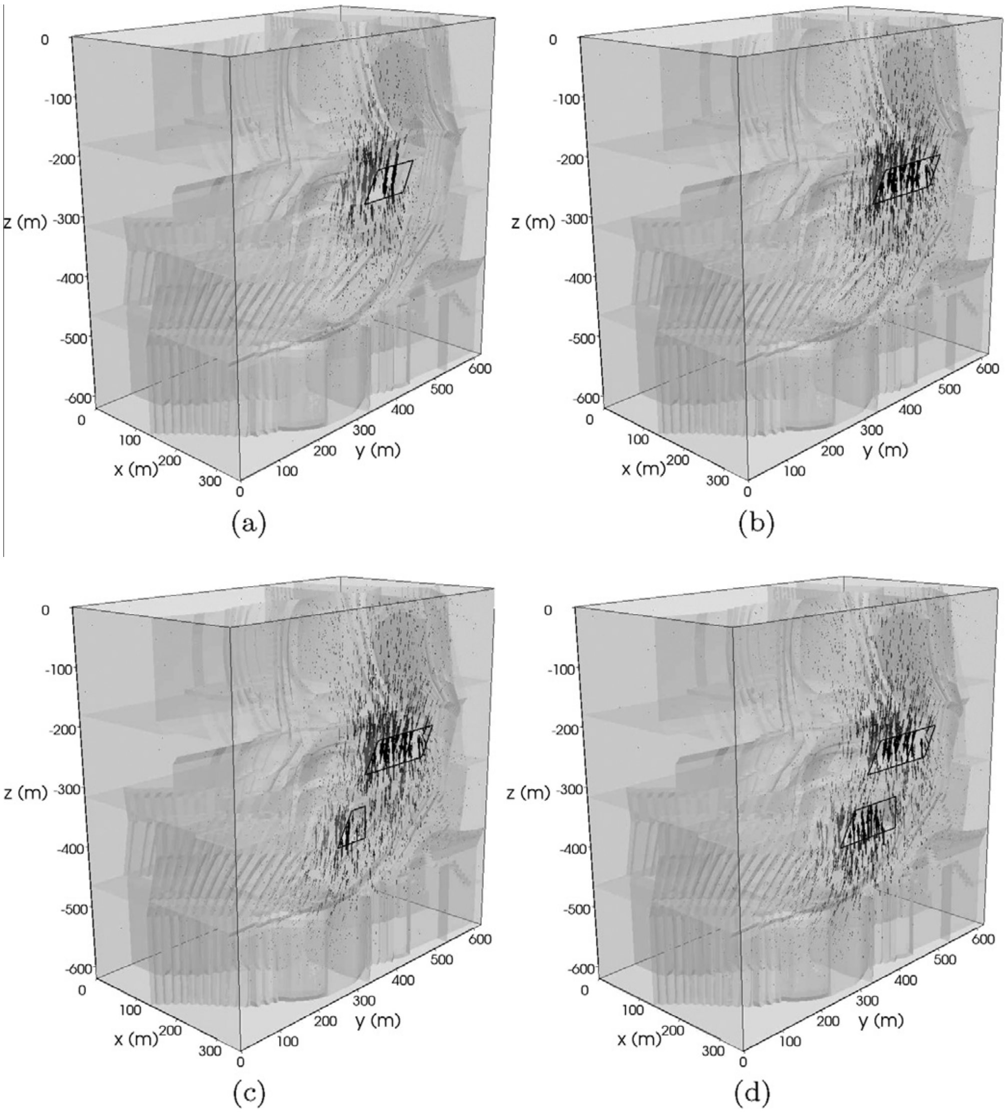


Fig. 24. Displacement vectors computed at four excavation stages, visualized in the section taken at $x = 350$ m. (a) Stage 1. (b) Stage 2. (c) Stage 3. (d) Stage 4. Shaded surfaces represent interfaces of partitioned subdomains. Vector fields in each figure are independently scaled to their range.

rock and 10 m for the ore body, resulting in a total of 107,712 spectral elements (Fig. 20(a) and (b)), which leads to a total of 7,161,572 nodes using three GLL points in each dimension of each element.

We partition the mesh into 32 domains for parallel processing (Fig. 21). We use a fixed-partition strategy and, therefore, distribution of the load among processors may vary in time and become less optimal as excavation progresses.

For this example, we set relative tolerances of 10^{-8} for conjugate gradient iterations and 10^{-5} for nonlinear iterations, as in the previous examples. We apply the excavation load in 10 increments.

Fig. 22 shows maximum displacements computed at four excavation stages. Only one iteration per load increment is needed for convergence, suggesting fully elastic behavior, due to the fact that the rock and ore body are very sound, having large cohesion and friction values. In fact, a single increment suffices for this model. There is no plastic deformation during these excavation stages. Figs. 23 and 24 illustrate displacement fields and displacement vectors, respectively, computed at four excavation stages. The upward displacements of roofs and downward displacements of floors are larger than displacements of side walls. In excavation stages 3 and 4, we observe regions of small displacements in the slab (i.e., the model portion between the two excavated cavities). The top surface of the slab experiences upward displacements

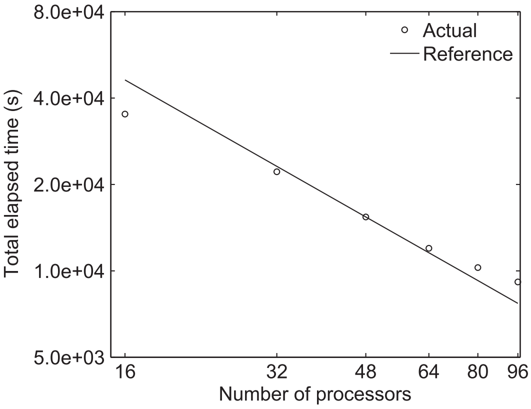


Fig. 25. Total elapsed time for a fixed problem size run on 16, 32, 48, 64, 80, and 96 processors, compared to a reference line computed using the total elapsed time on 48 processors.

and the bottom surface experiences downward displacements due to removal of material during respective excavations. However, small displacements do not necessarily imply stability of the slab. Such slabs may actually behave as a bending plate under certain circumstances (e.g., [89]).

4.5. Parallel performance

Finally, we conduct a strong-scaling performance test of our parallel program using the model of the Pyhäsalmi mine. We measure total computation times for different numbers of processors, keeping the problem size fixed. We run the parallel program on 16, 32, 48, 64, 80, and 96 processors. The results shown in Fig. 25 illustrate that the code scales reasonably well for large problems. Although we have used a fixed-partition strategy, parallel performance is reasonable. For the cases in which load-balancing is severely affected, partitioning of the intact region at every excavation stage may be a suitable option, despite requiring extra computation for preprocessing and determination of the communication topology. In the future, it may be important to implement efficient algorithms for node renumbering, e.g., the reverse Cuthill–McKee algorithm [90], to improve parallel performance (e.g., [91]). In addition, GPU (Graphics processing unit) could be utilized to further improve the performance in future versions of our software (e.g., [92–94]).

5. Discussion and conclusions

We have successfully implemented a spectral-element method for 3D multistage excavation. The numerical method is parallelized based on domain decomposition using MPI. Our program satisfies the uniqueness principle for multistage excavation in linear elastic materials. We have validated both the serial and parallel versions of the program, and demonstrated several simulations of multistage excavation in elastoplastic materials. We have simulated multistage excavation in a heterogeneous model of the Pyhäsalmi mine in Finland. This simulation illustrates a potential application of the software to complex and large-scale excavations. Due to very sound rock in the mine, unstable zones are barely visible during the excavation stages. We plan to perform future simulations with a more realistic model of the Pyhäsalmi mine that better captures the in situ stress state and actual excavation stages. Since mining operations are often initiated by blasting, it would be helpful to include blasting effects by considering brittle failure.

We have used a mesh with uniform element size for purposes of demonstration. Using a mesh with local refinement, e.g., geometrically adaptive meshing, may be important for efficient simulations of large-scale problems. For load-controlled problems in elastoplastic media, the constant-stiffness approach overestimates the stiffness near collapse. As a result, a large number of iterations is required to achieve convergence (e.g., [16]). Hence, other more accurate and efficient integration algorithms for elastoplastic constitutive relationships may be important, e.g., modified Euler methods with drift correction and automatic error control (e.g., [95,96]), or return mapping algorithms (e.g., [64,97]).

In some excavation problems, it may also be important to assess long-term creeping behaviour of structures such as tunnels or mines. This would require implementation of viscoelasticity or viscoplasticity. We have only implemented material nonlinearity, and in the future it will also be important to implement geometrical nonlinearity for the analysis of large displacements, e.g., based on adaptive mesh refinement.

Our software called SPECFEM3D_GEOTECH is open-source, and the entire package is freely available via www.geodynamics.org.

Acknowledgments

We thank Daniela Kühn, Michael Roth, and Valérie Maupin for helpful discussions and suggestions, Katja Sahala and ISS for access to the mine model and in-mine data, and Ricardo M. Garcia, Jr. for his help with meshing. Parallel programs were run on the Titan cluster owned by the University of Oslo and the Norwegian metacenter for High Performance Computing (NOTUR), and operated by the Research Computing Services group at USIT, the University of Oslo IT-department; and at the Princeton Institute for Computational Science and Engineering (PICSciE), USA. 3D data were visualized using the open-source parallel visualization software package ParaView/VTK (www.paraview.org). This work was funded in part by the Norwegian Research Council, and supported by industry partners BP, Statoil, and Total. We thank two anonymous reviewers for their insightful remarks.

References

[1] Hoek E, Brown T. Underground excavations in rock. Institution of Mining and metallurgy; 1990.
[2] Dunlop P, Duncan JM. Development of failure around excavated slopes. J Soil Mech Found Division 1970;96(2):471–93.
[3] Hughes TJR. The finite element method: linear static and dynamic finite element analysis. Prentice-Hall; 1987.
[4] Bathe KJ. Finite element procedures. Prentice Hall; 1995.
[5] Zienkiewicz OC, Taylor RL. The finite element method for solid and structural mechanics. Elsevier Butterworth-Heinemann; 2005.
[6] Christian JT, Wong IH. Errors in simulating excavation in elastic media by finite elements. Soils Found 1973;13(1):1–10.
[7] Clough GW, Mana AI. Lessons learned in finite element analysis of temporary excavations. In: International conference on numerical method in geomechanics. ASCE; 1976.
[8] Mana, AI, Finite element analysis of deep excavation behavior in soft clay, Ph.D. thesis, Stanford University, Stanford, CA, 1978.
[9] Ishihara K. Relations between process of cutting and uniqueness of solutions. Soils Found 1970;10:50–65.
[10] Ghaboussi J, Pecknold DA. Incremental finite element analysis of geometrically altered structures. Int J Numer Methods Eng 1984;20(11):2051–64.
[11] Desai CS, Sargand S. Hydrid FE procedure for soil-structure interaction. J Geotech Eng 1984;110:473–86.
[12] Borja RI, Lee SR, Seed BR. Numerical simulation of excavation in elastoplastic soils. Int J Numer Anal Methods Geomech 1989;13(6):231–49. http://dx.doi.org/10.1002/nag.1610130302.
[13] Borja RI. Analysis of incremental excavation based on critical state theory. J Geotech Eng 1990;116(6):964–85.
[14] Comodromos E, Hatzigogos T, Pitilakis K. Multi-stage finite element algorithm for excavation in elastoplastic soils. Comput Struct 1993;46(2):289–98. http://dx.doi.org/10.1016/0045-794(93)90193-H.
[15] Wilson EL. The static condensation algorithm. Int J Numer Methods Eng 1974;8:198–203. http://dx.doi.org/10.1002/nme.1620080115.
[16] Smith IM, Griffiths DV. Programming the finite element method. John Wiley & Sons; 2004.
[17] Chandrasekaran VS, King GJW. Simulation of excavation using finite elements. J Geotech Eng Division 1974;100(9):1086–9.
[18] Brown PT, Booker JR. Finite element analysis of excavation. Comput Geotech 1985;1(3):207–20. http://dx.doi.org/10.1016/0266-352(85)90024-2.
[19] Griffiths DV, Koutsabeloulis N. Finite element analysis of vertical excavations. Comput Geotech 1985;1(3):221–35.
[20] Ou C-Y, Chiou D-C, Wu T-S. Three-dimensional finite element analysis of deep excavations. J Geotech Eng 1996;122(5):337–45.
[21] Galli G, Grimaldi A, Leonardi A. Three-dimensional modelling of tunnel excavation and lining. Comput Geotech 2004;31(3):171–83.
[22] Hou Y, Wang J, Zhang L. Finite-element modeling of a complex deep excavation in Shanghai. Acta Geotech 2009;4:7–16. http://dx.doi.org/10.1007/s11440-008-0062-3.
[23] Smith IM, Leng J, Margetts L. Parallel three-dimensional finite element analysis of excavation, In: 13th ACME conference, University of Sheffield, UK, 2005.
[24] Holt DA, Griffiths DV. Transient analysis of excavations in soil. Comput Geotech 1992;13(3):159–74. http://dx.doi.org/10.1016/0266-352(92)90002-B.
[25] Hsi JP, Small JC. Simulation of excavation in a poro-elastic material. Int J Numer Anal Methods Geomech 1992;16:25–43. http://dx.doi.org/10.1002/nag.1610160104.
[26] Cundall PA, Strack ODL. A discrete numerical model for granular assemblies. Géotechnique 1979;29(1):47–65.
[27] Itasca, Itasca software products – FLAC, FLAC3D, PFC2D, PFC3D, UDEC, and 3DEC (2004). <www.itascacg.com>.
[28] Zhu Z, Li H, Liu, Q, He X, Numerical simulation for tunnel excavation in stratified rock mass by FLAC3D, In: IEEE 10th International Conference Computer-Aided Industrial Design & Conceptual Design, 2009, pp. 2271–2274.

[29] Belytschko T, Black T. Elastic crack growth in finite elements with minimal remeshing. Int J Numer Methods Eng 1999;45(5):601–20.

[30] Sukumar N, Moës N, Moran B, Belytschko T. Extended finite element method for three-dimensional crack modeling. Int J Numer Methods Eng 2000;48(11):1549–70.

[31] Hori M, Oguni K, Sakaguchi H. Proposal of FEM implemented with particle discretization for analysis of failure phenomena. J Mech Phys Solids 2005;53(3):681–703.

[32] Oguni K, Wijerathne MLL, Okinaka T, Hori M. Crack propagation analysis using PDS-FEM and comparison with fracture experiment. Mech Mater 2009;41(11):1242–52.

[33] Munjiza A. The combined finite-discrete element method. John Wiley and Sons; 2004.

[34] Price MA, Armstrong CG. Hexahedral mesh generation by medial surface subdivision: Part II. Solids with flat and concave edges. Int J Numer Methods Eng 1997;40(1):111–36.

[35] Tautges TJ. The generation of hexahedral meshes for assembly geometry: survey and progress. Int J Numer Methods Eng 2001;50(12):2617–42.

[36] Shepherd J, Johnson C. Hexahedral mesh generation constraints. Eng Comput 2008;24:195–213.

[37] Sandia National Laboratories, CUBIT 13.0 User Documentation, 27-May-2011, 2011. <cubit.sandia.gov>.

[38] Rainsberger R, TrueGrid user's manual, XYZ scientific applications, Inc., Livermore, CA, version 2.3.0 Edition 2006. <www.truegrid.com>.

[39] Geuzaine C, Remacle JF. Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. Int J Numer Methods Eng 2009;79(11):1309–31.

[40] Hesthaven JS, Teng CH. Stable spectral methods on tetrahedral elements. SIAM J Sci Comput 1999;21(6):2352–80.

[41] Taylor MA, Wingate BA. A generalized diagonal mass matrix spectral element method for non-quadrilateral elements. Appl Numer Math 2000;33(1-4):259–65.

[42] Komatitsch D, Martin R, Tromp J, Taylor MA, Wingate BA. Wave propagation in 2-D elastic media using a spectral element method with triangles and quadrangles. J Comput Acoust 2001;9:703–18.

[43] Mercerat ED, Vilotte JP, Sánchez-Sesma FJ. Triangular spectral element simulation of two-dimensional elastic wave propagation using unstructured triangular grids. Geophys J Int 2006;166:679–98.

[44] Durufle M, Grob P, Joly P. Influence of Gauss and Gauss–Lobatto quadrature rules on the accuracy of a quadrilateral finite element method in the time domain. Numer Methods Part Differ Equat 2009;25:526–51.

[45] Seriani G, Oliveira SP. Dispersion analysis of spectral-element methods for elastic wave propagation. Wave Motion 2008;45:729–44.

[46] De Basabe JD, Sen MK. Stability of the high-order finite elements for acoustic or elastic wave propagation with high-order time stepping. Geophys J Int 2010;181(1):577–90.

[47] Patera AT. A spectral element method for fluid dynamics: laminar flow in a channel expansion. J Comput Phys 1984;54:468–88.

[48] Canuto C, Hussaini MY, Quarteroni A, Zang TA. Spectral methods in fluid dynamics. Springer; 1988.

[49] Cohen G. Higher-order numerical methods for transient wave equations. Berlin, Germany: Springer-Verlag; 2002.

[50] Deville MO, Fischer PF, Mund EH. High-order methods for incompressible fluid flow. Cambridge, UK: Cambridge University Press; 2002.

[51] Seriani G. 3-D large-scale wave propagation modeling by spectral element method on Cray T3E multiprocessor. Comput Methods Appl Mech Eng 1994;164:235–47.

[52] Faccioli E, Maggio F, Paolucci R, Quarteroni A. 2D and 3D elastic wave propagation by a pseudo-spectral domain decomposition method. J Seismol 1997;1(3):237–51.

[53] Komatitsch D, Vilotte JP. The spectral element method: an efficient tool to simulate the seismic response of 2D and 3D geological structures. Bull Seismol Soc Am 1998;88(2):368–92.

[54] Komatitsch D, Tromp J. Introduction to the spectral element method for three-dimensional seismic wave propagation. Geophys J Int 1999;139:806–22.

[55] Vai R, Castillo-Covarrubias JM, Sánchez-Sesma FJ, Komatitsch D, Vilotte JP. Elastic wave propagation in an irregularly layered medium. Soil Dyn Earthquake Eng. 1999;18(1):11–8.

[56] Tromp J, Komatitsch D, Liu Q. Spectral-element and adjoint methods in seismology. Commun Comput Phys 2008;3(1):1–32.

[57] Oye V, Gharti HN, Aker E, Kühn D. Moment tensor analysis and comparison of acoustic emission data with synthetic data from spectral element method. SEG Tech Prog Expanded Abstracts 2010;29(1):2105–9. http://dx.doi.org/10.1190/1.3513260.

[58] Peter D, Komatitsch D, Luo Y, Martin R, Le Goff N, Casarotti E, et al. Forward and adjoint simulations of seismic wave propagation on fully unstructured hexahedral meshes. Geophys J Int 2011;186(2):721–39.

[59] di Prisco C, Stupazzini M, Zambelli C. Nonlinear SEM numerical analyses of dry dense sand specimens under rapid and dynamic loading. Int J Numer Anal Methods Geomech 2007;31:757–88.

[60] Gharti, HN, Komatitsch D, Oye V, Martin R, Tromp J, Application of an elastoplastic spectral-element method to 3D slope stability analysis. Int J Numer Methods Eng 2012. http://dx.doi.org/10.1002/nme.3374.

[61] Luccioni LX, Pestana JM, Taylor RL. Finite element implementation of non-linear elastoplastic constitutive laws using local and global explicit algorithms with automatic error control. Int J Numer Methods Eng 2001;50:1191–212.

[62] Dupros F, De Martin F, Foerster E, Komatitsch D, Roman J. High-performance finite-element simulations of seismic wave propagation in three-dimensional nonlinear inelastic geological media. Parallel Comput 2010;36:308–25.

[63] Babuska I, Suri M. The p – and h-p versions of the finite element method, an overview. Comput Methods Appl Mech Eng 1990;80:5–26.

[64] Simo JC, Hughes TJR. Computational inelasticity. Springer; 1998.

[65] Schwab C. p – and hp- finite element methods: theory and applications to solid and fluid mechanics. USA: Oxford University Press; 1999.

[66] Belytschko T, Liu WK, Moran B. Nonlinear finite elements for continua and structures. Wiley; 2000.

[67] Hughes TJR, Levit I, Winget J. An element-by-element solution algorithm for problems of structural and solid mechanics. Comput Methods Appl Mech Eng 1983;36(2):241–54. http://dx.doi.org/10.1016/0045-782(83)90115-9.

[68] Gropp W, Lusk E, Skjellum A. Using MPI, portable parallel programming with the message-passing interface. Cambridge, USA: MIT Press; 1994.

[69] Pacheco P. Parallel programming with MPI. Morgan Kaufman; 1997.

[70] Terzaghi K. Theoretical soil mechanics. New York: J. Wiley; 1943.

[71] Zienkiewicz O, Cormeau I. Visco-plasticity–plasticity and creep in elastic solids—a unified numerical solution approach. Int J Numer Methods Eng 1974; 8(4):821–45.

[72] Topping BHV, Khan AI. Parallel finite element computations. Saxe-Coburg Publications; 1999.

[73] Liu Y, Zhou W, Yang Q. A distributed memory parallel element-by-element scheme based on Jacobi-conditioned conjugate gradient for 3D finite element analysis. Finite Elem Anal Des 2007;43:494–503.

[74] Pellegrini F, Roman J. SCOTCH: a software package for static mapping by dual recursive bipartitioning of process and architecture graphs. Lect Notes Comput Sci 1996;1067:493–8.

[75] Karypis G, Kumar V. Multilevel k-way partitioning scheme for irregular graphs. J Parallel Distrib Comput 1998;48(1):96–129.

[76] Law KH. A parallel finite element solution method. Comput Struct 1986;23(6):845–58. http://dx.doi.org/10.1016/0045-794(86)90254-3.

[77] King RB, Sonnad V. Implementation of an element-by-element solution algorithm for the finite element method on a coarse-grained parallel computer. Comput Methods Appl Mech Eng 1987;65(1):47–59. http://dx.doi.org/10.1016/0045-782(87)90182-4.

[78] Barragy E, Carey GF. A parallel element-by-element solution scheme. Int J Numer Methods Eng 1988;26:2367–82.

[79] Sobh NA, Gustafson K. Preconditioned conjugate gradient and finite element methods for massively data-parallel architectures. Comput Phys Commun 1991;65:253–67.

[80] Khan AI, Topping BHV. Parallel finite element analysis using Jacobi-conditioned conjugate gradient algorithm. Adv Eng Softw 1996;25:309–19.

[81] Margetts L, Parallel finite element analysis, Ph.D. thesis, University of Manchester, 2002.

[82] Nakajima K, Okuda H. Parallel iterative solvers with localized ILU preconditioning for unstructured grids on workstation clusters. Int J Comput Fluid Dynam 1999;12(3):315–22.

[83] Taylor DW. Stability of earth slopes. J Boston Soc Civil Eng 1937;24:197–246.

[84] Heyman J. Simple plastic theory applied to soil mechanics, In: Symposium on the role of Plasticity in Soil Mechanics, 1973:151–72.

[85] Oye V, Bungum H, Roth M. Source parameters and scaling relations for mining related seismicity within the Pyhaesalmi ore mine, Finland. Bull Seismol Soc Am 2005;95:1011–26.

[86] Puustjärvi H, Pyhaesalmi modeling project, Section B. Geology, Technical report., Geological Survey of Finland, and Outokumpu Mining Oy 1999.

[87] Gharti HN, Oye V, Roth M. Travel times and waveforms of microseismic data in heterogeneous media. SEG Tech Prog Expanded Abstracts 2008;27(1): 1337–41. http://dx.doi.org/10.1190/1.3059162.

[88] Stein S, Wysession M. An introduction to seismology. Earthquakes and earth structure. Wiley, Blackwell; 2002.

[89] Reddy JN. Introduction to the finite element method. McGraw-Hill; 1993.

[90] Cuthill E, McKee J. Reducing the bandwidth of sparse symmetric matrices. In: Proceedings of the 24th National ACM Conference. New York: ACM Press; 1969. p. 157–72.

[91] Komatitsch D, Labarta J, Michéa D. A simulation of seismic wave propagation at high resolution in the inner core of the earth on 2166 processors of marenostrum. In: Palma JM, Amestoy PR, Daydé M, Mattoso M, Correia Lopes J, editors. High performance computing for computational science – VECPAR 2008. Berlin, Heidelberg: Springer-Verlag; 2008. p. 364–77.

[92] Göddeke D, Strzodka R, Mohd-Yusof J, McCormick P, Buijssen SHM, Grajewski M, et al. Exploring weak scalability for FEM calculations on a GPU-enhanced cluster. Parallel Comput 2007;33(10–11):685–99.

[93] Komatitsch D, Michéa D, Erlebacher G. Porting a high-order finite-element earthquake modeling application to NVIDIA graphics cards using CUDA. J Parallel Distrib Comput 2009;69(5):451–60.

[94] Komatitsch D, Erlebacher G, Göddeke D, Michéa D. High-order finite-element seismic wave propagation modeling with MPI on a large GPU cluster. J Comput Phys 2010;229(20):7692–714. http://dx.doi.org/10.1016/j.jcp.2010.06.024.

[95] Abbo AJ, Sloan SW. An automatic load stepping algorithm with error control. Int J Numer Methods Eng 1996;39:1737–59.

[96] Sloan SW, Abbo AJ, Sheng D. Refined explicit integration of elastoplastic models with automatic error control. Eng Comput 2001;18:121–94.

[97] de Souza Neto EA, Perić D, Owen DRJ. Computational methods for plasticity: theory and applications. Wiley; 2009.