Computers and Structures 87 (2009) 59–72

Contents lists available at ScienceDirect

Computers and Structures

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

A parallel spectral element method for dynamic three-dimensional nonlinear elasticity problems

S. Dong [a,*], Z. Yosibash [b]

[a] Center for Computational and Applied Mathematics, Department of Mathematics, Purdue University, West Lafayette, IN 47907, USA
[b] Pearlstone Center for Aeronautical Studies, Department of Mechanical Engineering, Ben-Gurion University of the Negev, Beer-Sheva, Israel

ARTICLE INFO

Article history:
Received 2 April 2008
Accepted 12 August 2008
Available online 25 September 2008

Keywords:
Spectral element method
hp finite element method
Exponential convergence
Jacobi polynomial
Nonlinear elasticity
Message passing interface

ABSTRACT

We present a high-order method employing Jacobi polynomial-based shape functions, as an alternative to the typical Legendre polynomial-based shape functions in solid mechanics, for solving dynamic three-dimensional geometrically nonlinear elasticity problems. We demonstrate that the method has an exponential convergence rate spatially and a second-order accuracy temporally for the four classes of problems of linear/geometrically nonlinear elastostatics/elastodynamics. The method is parallelized through domain decomposition and message passing interface (MPI), and is scaled to over 2000 processors with high parallel performance.

1. Introduction

During the past two decades p/hp-versions of the finite element method (FEM) have evolved considerably, shown to provide robust and efficient results for problems of solid mechanics (particularly linear elasticity) [32,33]. In linear elasticity the high-order methods possess many advantages over "classical FEMs", such as considerably higher convergence rates, the flexibility of using large aspect ratios of elements without significant deterioration in accuracy, locking-free behavior with respect to thickness for plate and shell-like structures, and with respect to the Poisson ratio for nearly incompressible materials. The efficiency and advantages of p-FEMs were also extended in recent years to nonlinear problems such as elasto-plasticity [20,15], flow–structure interaction [30,31], and finite-deformation problems with follower loads [26,14,35,19], demonstrating that p-FEM's advantages carry over to nonlinear solid mechanics problems.

Despite the significant growth in the breadth of its applicability, two areas have received relatively less attention in p/hp-FEM for solid mechanics: dynamic problems and parallelism. To the best of our knowledge, the only applications of p-FEMs to dynamic problems are those in [18,29]. In [18] linear elastodynamic problems were investigated and some formulations towards future finite-deformation implementation were briefly mentioned. In [29] p-FEMs for curved elastic and isotropic beams taking into account geometric nonlinearities were used to investigate the vibrations occurring due to harmonic excitations. In regard to parallelism, to fully take advantage of present-day large-scale parallel computers which are usually equipped with 1000–10,000 processors, efficient parallelization of solid mechanics p/hp-FEM codes is required and remains an extremely challenging problem. The stringency of the challenge becomes clear if one takes into account the fact that existing p-FEM codes for solid mechanics usually scale to about 32 or 64 processors [2,17,27,22,25,28,7]. This is in part due to the less efficient or obsolescent technologies such as purely shared-memory parallelism [2,22] and parallel virtual machine (PVM) [17,28] employed in these applications. The situation will likely be exacerbated with the advent of peta-scale parallel computers at the end of this or the beginning of next decade, with a processor count currently anticipated on the order of 1,000,000.

The shape functions based on Legendre polynomials [32] have been the dominant bases in p-FEM implementations, and have witnessed widespread applications in solid mechanics. In this paper, we employ a new set of shape functions based on generalized/warped tensor products of the more general Jacobi polynomials [13,23] to solve the three-dimensional elasticity equations. Although Jacobi polynomial-based shape functions have been used for solving Navier–Stokes equations in high-order computational fluid dynamics (CFD) [23,10,11,8,12], they have largely been ignored in solid mechanics. (The method in high-order CFD is popularly termed spectral element method.) Compared to those based on Legendre polynomials, the Jacobi polynomial-based shape functions possess several advantages. For example, they result in mass and stiffness matrices with more favorable numerical conditioning [6]. The Jacobi-based approach provides a unified treatment for

[*] Corresponding author.
E-mail address: sdong@math.purdue.edu (S. Dong).

0045-7949/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compstruc.2008.08.008

polymorphic geometric shapes (i.e. hexahedrons, pentahedrons/prisms, tetrahedrons and pyramids), and is very flexible in generating hybrid meshes in hp-extensions. They also allow for very high element orders; up to the order $p = 100$ can be easily obtained. It is therefore highly desirable to exploit these advantages of Jacobi polynomial-based shape functions to construct an efficient high-order method capable of handling commonly encountered polymorphic elements for three-dimensional solid mechanics problems.

Herein we enlarge the fields of the application of p-FEMs to problems of three-dimensional solid dynamics, both linear elastodynamics and geometrically nonlinear hyper-elastodynamics (finite deformations). Since these problems are very demanding in terms of computational resources, parallelization algorithms are introduced through message passing interface (MPI) and the parallel performance is demonstrated on more than 2000 CPUs. In our implementation we employ the hierarchical shape functions based on Jacobi polynomials for spatial discretizations, instead of the typical Legendre polynomial-based shape functions [32]. The resulting method can handle all types of commonly encountered three-dimensional elements of high order, e.g. hexahedrons, pentahedrons/prisms, tetrahedrons and pyramids. For temporal discretization we have chosen to employ the average acceleration variant of the Newmark-β scheme [5,3,21] (with Newmark parameters $\gamma = \frac{1}{2}$ and $\beta = \frac{1}{4}$). This is an implicit scheme that exhibits second-order convergence in time and is unconditionally stable under linear analysis. Although the Newmark scheme may become unstable in some cases for geometrically nonlinear elastodynamic problems, in the present applications stability is achieved, and other effective time integration schemes (e.g. [1,4]) will be investigated in the context of high-order methods in a future study.

For simplicity of presentation we consider the so-called St. Venant–Kirchhoff constitutive equation as a model problem for the finite-strain solid dynamic applications. This model, although non-physical at large strains, is simple enough for detailed numerical investigations, and can be easily replaced by any other more realistic hyper-elastic model. It is obtained from the strain energy density function:

$$\Psi(\mathbf{E}) = \frac{\lambda}{2}(\mathrm{tr}\,\mathbf{E})^2 + \mu\mathbf{E} : \mathbf{E} \tag{1}$$

by using hyper-elasticity relations:

$$\mathbf{S} = \frac{\partial\Psi}{\partial\mathbf{E}} = \lambda(\mathrm{tr}\,\mathbf{E})\mathbf{I} + 2\mu\mathbf{E}, \tag{2}$$

where $\mathbf{S}$ is the second Piola–Kirchhoff stress tensor, $\mathbf{E}$ is the Green–Lagrange strain tensor, and λ and μ are material coefficients.

This paper is organized as follows. We first briefly introduce the shape functions based on Jacobi polynomials in Section 2. The departing point is the linear three-dimensional elastodynamic formulation and its implementation in the p-FEM framework in Section 3.1. Both spatial discretization by p-FEM and temporal time-marching Newmark scheme are briefly described. In Section 3.2, we formulate the dynamic-hyper-elastic problem where an iterative Newton–Raphson scheme and the Newmark scheme are applied to the nonlinear problem. In Section 4, we discuss the parallelization of the linear and nonlinear solvers. In Section 5, example problems having analytical solutions are used to demonstrate the spatial exponential convergence rate and the temporal second-order accuracy of our schemes. Finally, we demonstrate the capabilities of the linear and nonlinear solvers with several elastostatic and elastodynamic test problems, and show that very large hyper-elastic problems can be computed on thousands of processors in a fraction of the time required for classical codes (Section 6). We summarize the paper in Section 7.

2. Shape functions based on Jacobi polynomials

In this section, we briefly introduce the hierarchical shape functions based on Jacobi polynomials that are employed in this paper (more details can be found in [23]). To facilitate the discussion we first define three principal functions on the interval $-1 \leqslant x \leqslant 1$ denoted by $\psi_i^a(x)$, $\psi_{ij}^b(x)$ and $\psi_{ijk}^c(x)$ $(0 \leqslant i \leqslant I,\ 0 \leqslant j \leqslant J,\ 0 \leqslant k\ \leqslant K$ where I, J and K are positive integers), which form the basis for constructing the shape functions in three-dimensional space:

$$\psi_i^a(x) = \begin{cases} \frac{1-x}{2}, & i = 0, \\ \frac{1-x}{2}\frac{1+x}{2}P_{i-1}^{1,1}(x), & 1 \leqslant i < I, \\ \frac{1+x}{2}, & i = I, \end{cases} \tag{3}$$

$$\psi_{ij}^b(x) = \begin{cases} \psi_j^a(x), & i = 0,\ 0 \leqslant j \leqslant J, \\ \left(\frac{1-x}{2}\right)^{i+1}, & 1 \leqslant i < I,\ j = 0, \\ \left(\frac{1-x}{2}\right)^{i+1}\frac{1+x}{2}P_{j-1}^{2i+1,1}(x), & 1 \leqslant i < I,\ 1 \leqslant j < J, \\ \psi_j^a(x), & i = I,\ 0 \leqslant j \leqslant J, \end{cases} \tag{4}$$

$$\psi_{ijk}^c(x) = \begin{cases} \psi_{jk}^b(x), & i = 0,\ 0 \leqslant j \leqslant J, \\ & 0 \leqslant k \leqslant K, \\ \psi_{ik}^b(x), & 0 \leqslant i \leqslant I,\ j = 0,\ 0 \leqslant k \leqslant K, \\ \left(\frac{1-x}{2}\right)^{i+j+1}, & 1 \leqslant i < I,\ 1 \leqslant j < J, \\ & k = 0, \\ \left(\frac{1-x}{2}\right)^{i+j+1}\frac{1+x}{2}P_{k-1}^{2i+2j+1,1}(x), & 1 \leqslant i < I,\ 1 \leqslant j < J, \\ & 1 \leqslant k < K, \\ \psi_{ik}^b(x), & 0 \leqslant i \leqslant I,\ j = J, \\ & 0 \leqslant k \leqslant K, \\ \psi_{jk}^b(x), & i = I,\ 0 \leqslant j \leqslant J, \\ & 0 \leqslant k \leqslant K. \end{cases} \tag{5}$$

In the above expressions $P_n^{\alpha,\beta}(x)$ $(\alpha, \beta > -1)$ are the Jacobi polynomials which represent the family of polynomial solutions to a singular Sturm–Liouville problem.

Assume that the coordinates of the standard domain are denoted by ξ_1, ξ_2 and ξ_3. Then the hierarchical shape functions in three-dimensional space are defined in the standard domain as follows in terms of the above principal functions ($\phi_{pqr}(\xi_1, \xi_2, \xi_3)$ denoting the shape function with p, q, r being appropriate indices):

- For a hexahedral element $\{(\xi_1, \xi_2, \xi_3) | -1 \leqslant \xi_1, \xi_2, \xi_3 \leqslant 1\}$, the shape function is defined by

$$\phi_{pqr}(\xi_1, \xi_2, \xi_3) = \psi_p^a(\xi_1)\psi_q^a(\xi_2)\psi_r^a(\xi_3). \tag{6}$$

- For a prismatic element $\{(\xi_1, \xi_2, \xi_3) | -1 \leqslant \xi_1, \xi_3;\ \ \xi_1 + \xi_3 \leqslant 1;$ $-1 \leqslant \xi_2 \leqslant 1\}$, the shape function is defined by

$$\phi_{pqr}(\xi_1, \xi_2, \xi_3) = \psi_p^a(\overline{\eta}_1)\psi_q^a(\xi_2)\psi_{pr}^b(\xi_3), \tag{7}$$

  where $\overline{\eta}_1 = \frac{2(1+\xi_1)}{1-\xi_3} - 1$.
- For a tetrahedral element $\{(\xi_1, \xi_2, \xi_3) | -1 \leqslant \xi_1, \xi_2, \xi_3;$ $\xi_1 + \xi_2 + \xi_3 \leqslant 1\}$, the shape function is defined by

$$\phi_{pqr}(\xi_1, \xi_2, \xi_3) = \psi_p^a(\eta_1)\psi_{pq}^b(\eta_2)\psi_{pqr}^c(\xi_3), \tag{8}$$

  where $\eta_1 = \frac{2(1+\xi_1)}{-\xi_2-\xi_3} - 1$ and $\eta_2 = \frac{2(1+\xi_2)}{1-\xi_3} - 1$.
- For a pyramidic element $\{(\xi_1, \xi_2, , \xi_3) | -1 \leqslant \xi_1, \xi_2, \xi_3;\ \xi_1 + \xi_3 \leqslant 1;$ $\xi_2 + \xi_3 \leqslant 1\}$, the shape function is defined by

$$\phi_{pqr}(\xi_1, \xi_2, \xi_3) = \psi_p^a(\overline{\eta}_1)\psi_q^a(\eta_2)\psi_{pqr}^c(\xi_3). \tag{9}$$

The set of shape functions can be classified into vertex modes, edge modes, face modes and interior modes to facilitate implementations. Herein the basis functions defined in Eqs. (6)–(9) will be employed to discretize the three-dimensional linear and geometrically nonlinear elastodynamic equations.

3. Formulation and discretization

Consider the deformation of a three-dimensional object occupying domain $\Omega \subset \mathbf{R}^3$ with boundary $\partial\Omega = \partial\Omega_D \cup \partial\Omega_N$, where Dirichlet boundary conditions (BC) are provided on $\partial\Omega_D$ and Neumann-type (traction) BCs on $\partial\Omega_N$.

3.1. Linear elastodynamics

Let us first assume infinitesimal deformations. The equation describing the object deformation, in its weak form, can be expressed as follows: Find the displacement field $\mathbf{u}(\mathbf{x},t) \in \mathbf{U}(t) = \{\mathbf{w}(\mathbf{x},t) \in [H^1(\Omega)]^3 | \mathbf{w}(\mathbf{x},t) = \mathbf{u}_D(\mathbf{x},t) \text{ on } \partial\Omega_D\}$ ($\mathbf{u}_D(\mathbf{x},t)$ being Dirichlet BCs) such that

$$\int_\Omega \rho \frac{\partial^2 \mathbf{u}}{\partial t^2} \cdot \mathbf{v} \, d\Omega = -\int_\Omega \mathbf{S} : \nabla\mathbf{v} \, d\Omega + \int_{\partial\Omega_N} \mathbf{T} \cdot \mathbf{v} \, d\Gamma + \int_\Omega \rho \mathbf{f} \cdot \mathbf{v} \, d\Omega \quad \forall \mathbf{v} \in \mathbf{U}_0, \tag{10}$$

where $\mathbf{U}_0 = \{\mathbf{w}(\mathbf{x},t) \in [H^1(\Omega)]^3 | \mathbf{w}(\mathbf{x},t) = \mathbf{0} \text{ on } \partial\Omega_D\}$. In the above equation, $\mathbf{S}$, $\mathbf{T}$, $\mathbf{f}$ and ρ are the stress tensor, external traction force on $\partial\Omega_N$, external body force and the mass density, respectively. This equation is complemented by appropriate initial conditions (IC) to form an initial/boundary-value problem. The St. Venant–Kirchhoff constitutive equation is assumed for the material (Eq. (2)).

We expand the displacement $\mathbf{u} = (u_1, u_2, u_3)$ and the test function $\mathbf{v} = (v_1, v_2, v_3)$ in terms of the shape functions described in Section 2:

$$u_i(\mathbf{x},t) = \sum_{p=1}^{N_m} \hat{u}_{ip}(t)\phi_p(\mathbf{x}), \tag{11}$$

$$v_i(\mathbf{x},t) = \sum_{p=1}^{N_m} \hat{v}_{ip}(t)\phi_p(\mathbf{x}), \quad i = 1,2,3. \tag{12}$$

In the above equation, N_m is the total number of modes (shape functions), $\hat{u}_{ip}$ and $\hat{v}_{ip}$ are the expansion coefficients. $\phi_p(\mathbf{x})$ denote the shape functions, with p being the index after appropriately organizing the three-dimensional modal indices in Section 2 into a one-dimensional vector. Substitute expressions (11) and (12) into Eq. (10), and we obtain

$$\mathbf{M}\frac{d^2\mathbf{U}}{dt^2} = -\mathbf{K}\mathbf{U} + \mathbf{F}. \tag{13}$$

In the above equation $\mathbf{U}$ is a vector of unknown expansion coefficients of length $3N_m$

$$\mathbf{U} = [\cdots \quad \mathbf{U}_p^T \quad \mathbf{U}_{p+1}^T \quad \cdots]^T, \tag{14}$$

where $\mathbf{U}_p = [\,\hat{u}_{1p} \quad \hat{u}_{2p} \quad \hat{u}_{3p}\,]^T$. The superscript $(\cdot)^T$ denotes transposition. We have organized the vector in a way such that the three displacement components corresponding to a particular mode are adjacent, rather than that the N_m modes of y displacement follow those of the x displacement and so forth. This ordering leads to a smaller bandwidth of the coefficient matrix, facilitates the Schur complement, and is conducive to domain decomposition in the parallel implementation. $\mathbf{F}$ is the vector of the external forces with length $3N_m$, and with the same ordering of indices, $\mathbf{F} = [\cdots \quad \mathbf{F}_p^T \quad \mathbf{F}_{p+1}^T \cdots]^T$, where $\mathbf{F}_p = [F_{1p} \quad F_{2p} \quad F_{3p}]^T$ and $F_{ip} = \int_{\partial\Omega_N} T_i\phi_p \, d\Gamma + \int_\Omega \rho f_i \phi_p \, d\Omega$ ($i = 1,2,3$), where T_i and f_i are the components of the external traction and body forces. Correspondingly, the $3N_m \times 3N_m$ mass matrix $\mathbf{M}$ is organized into a matrix of $N_m \times N_m$ blocks, each block being a 3×3 submatrix:

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_{11} & \cdots & \mathbf{M}_{1N_m} \\ \vdots & \ddots & \vdots \\ \mathbf{M}_{N_m1} & \cdots & \mathbf{M}_{N_mN_m} \end{bmatrix}, \quad \mathbf{M}_{pq} = \left(\int_\Omega \rho\phi_p\phi_q \, d\Omega\right)\mathbf{I}_3, \quad p,q = 1,\ldots,N_m, \tag{15}$$

where $\mathbf{I}_3$ is the 3×3 identity matrix. Similarly, the $3N_m \times 3N_m$ stiffness matrix $\mathbf{K}$ is also organized into a matrix of $N_m \times N_m$ blocks, and the 3×3 submatrix $\mathbf{K}_{pq}$ ($p,q = 1,\ldots,N_m$) of a block is given by $\mathbf{K}_{pq} = \int_\Omega (\mathbf{D}\phi_p)^T \mathbf{B} (\mathbf{D}\phi_q) d\Omega$, where

$$\mathbf{D}\phi_p = \begin{bmatrix} \frac{\partial\phi_p}{\partial x} & 0 & 0 & \frac{\partial\phi_p}{\partial y} & 0 & \frac{\partial\phi_p}{\partial z} \\ 0 & \frac{\partial\phi_p}{\partial y} & 0 & \frac{\partial\phi_p}{\partial x} & \frac{\partial\phi_p}{\partial z} & 0 \\ 0 & 0 & \frac{\partial\phi_p}{\partial z} & 0 & \frac{\partial\phi_p}{\partial y} & \frac{\partial\phi_p}{\partial x} \end{bmatrix}^T,$$

$$\mathbf{B} = \begin{bmatrix} \lambda+2\mu & \lambda & \lambda & & & \\ \lambda & \lambda+2\mu & \lambda & & & \\ \lambda & \lambda & \lambda+2\mu & & & \\ & & & \mu & & \\ & & & & \mu & \\ & & & & & \mu \end{bmatrix}.$$

In the implementation, in Eq. (13) the contributions from the known expansion coefficients due to Dirichlet BCs are moved to the right-hand side before the equation is solved.

We employ the Newmark-β scheme [3,5,21] with parameters, $\gamma = \frac{1}{2}$ and $\beta = \frac{1}{4}$, to discretize Eq. (13) in time. Let n denote the index of the time step

$$\mathbf{M}\ddot{\mathbf{U}}^{n+1} + \mathbf{K}\mathbf{U}^{n+1} = \mathbf{F}^{n+1}, \tag{16}$$

where we have used $(\dot{\cdot})$ to denote the temporal derivative. Employing the following expressions for the displacement and velocity at time step $(n+1)$ from the Newmark scheme (Δt denoting the time step size):

$$\dot{\mathbf{U}}^{n+1} = \dot{\mathbf{U}}^n + \frac{1}{2}(\ddot{\mathbf{U}}^n + \ddot{\mathbf{U}}^{n+1})\Delta t, \tag{17}$$

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \dot{\mathbf{U}}^n\Delta t + \frac{1}{4}(\ddot{\mathbf{U}}^n + \ddot{\mathbf{U}}^{n+1})(\Delta t)^2. \tag{18}$$

Eq. (16) is transformed to

$$\left(\frac{4}{(\Delta t)^2}\mathbf{M} + \mathbf{K}\right)\mathbf{U}^{n+1} = \mathbf{F}^{n+1} + \mathbf{M}\left(\frac{4}{(\Delta t)^2}\mathbf{U}^n + \frac{4}{\Delta t}\dot{\mathbf{U}}^n + \ddot{\mathbf{U}}^n\right). \tag{19}$$

The resulting symmetric linear system can be solved with the conjugate gradient iterative solver.

Remarks. Because the interior modes of an element are coupled only with the boundary modes (i.e. vertex/edge/face modes) of the same element, in the implementation we arrange the vector $\mathbf{U}$ such that the vertex modes are followed by the edge modes, face modes, and the interior modes, and a Schur complement is performed on the linear system (19) to condense out all the interior modes. This results in a smaller linear system of equations about the boundary modes only, which is then solved with the conjugate gradient solver. Schur complement also results in N_{el} (N_{el} denoting the number of elements) linear systems of much smaller sizes, with a separate system for the interior modes of each element. These small linear systems about the interior modes are solved with a direct solver.

3.2. Geometrically nonlinear elastodynamics

We next consider the finite deformation of the object. Let $\mathbf{X}$ denote the position vector of a material point in the initial configuration of the object, Ω_0, at time $t = 0$, and let $\mathbf{x}$ denote its position vector at time t in the deformed configuration, $\Omega(t)$. Then the displacement vector $\mathbf{u}$ is a function of $\mathbf{X} : \mathbf{u} = \mathbf{x}(\mathbf{X},t) - \mathbf{X}$.

The weak form of the momentum equation represents the principle of virtual work. With respect to the initial configuration, it can be stated as follows: Find the displacement field $\mathbf{u}(\mathbf{X},t) \in \mathbf{V}(t) = \{\mathbf{w}(\mathbf{X},t) \in [H^1(\Omega_0)]^3 | \mathbf{w}(\mathbf{X},t) = \mathbf{u}_D(\mathbf{X},t) \text{ on } \partial\Omega_{0D}\}$ ($\partial\Omega_{0D}$ is the Dirichlet boundary in the initial configuration) such that

$$\mathscr{P}(\mathbf{u},\mathbf{v}) = \int_{\Omega_0} \mathbf{S} : \frac{1}{2}\left(\left(\frac{\partial \mathbf{v}}{\partial \mathbf{X}}\right)^{\mathrm{T}} \cdot \mathbf{F}(\mathbf{u}) + \mathbf{F}^{\mathrm{T}}(\mathbf{u}) \cdot \frac{\partial \mathbf{v}}{\partial \mathbf{X}}\right) \mathrm{d}\Omega_0 \\ - \int_{\partial\Omega_{0N}} \mathbf{T}\cdot\mathbf{v}\,\mathrm{d}\Gamma - \int_{\Omega_0} \rho_0 \mathbf{f}\cdot\mathbf{v}\,\mathrm{d}\Omega_0 + \int_{\Omega_0} \rho_0 \frac{\partial^2 \mathbf{u}}{\partial t^2}\cdot\mathbf{v}\,\mathrm{d}\Omega_0 = 0 \quad \forall \mathbf{v} \in \mathbf{V}_0, \tag{20}$$

where $\mathbf{V}_0 = \{\mathbf{w}(\mathbf{X},t) \in [H^1(\Omega_0)]^3 | \mathbf{w}(\mathbf{X},t) = 0 \text{ on } \partial\Omega_{0D}\}$. In the above equation $\mathbf{S}$, $\mathbf{f}$ and ρ_0 are the second Piola–Kirchhoff stress tensor, external body force, and the structural mass density in the initial configuration, respectively. The external traction force $\mathbf{T}$ is assumed to be deformation-independent (i.e. non-follower load). The deformation gradient tensor $\mathbf{F}(\mathbf{u})$ is defined by $\mathbf{F}(\mathbf{u}) = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \mathbf{I}_3 + \frac{\partial \mathbf{u}}{\partial \mathbf{X}}$.

We again consider the St. Venant–Kirchhoff constitutive law for the material (Eq. (1)). Then Eq. (2) can be rewritten as

$$\mathbf{S} = \lambda(\mathrm{tr}\,\mathbf{E})\mathbf{I} + 2\mu\mathbf{E} = \mathscr{C}^{(4)} : \mathbf{E}, \tag{21}$$

where $\mathscr{C}^{(4)}$ is a fourth-order constant tensor (elasticity tensor) representing the material properties, and $\mathbf{E}$ is the Green–Lagrange strain tensor, $\mathbf{E}(\mathbf{u}) = \frac{1}{2}(\mathbf{F}^{\mathrm{T}}(\mathbf{u}) \cdot \mathbf{F}(\mathbf{u}) - \mathbf{I}_3)$. We use $\mathscr{P}^{\mathrm{int}}(\mathbf{u},\mathbf{v})$ to denote the first term in Eq. (20), which represents the virtual work due to the internal stress and is nonlinear with respect to the displacement $\mathbf{u}$. The second and the third terms represent the virtual work due to the external forces, and will be denoted by $\mathscr{P}^{\mathrm{ext}}(\mathbf{u},\mathbf{v})$. The last term is the virtual work due to the inertia, and will be denoted by $\mathscr{P}^{\mathrm{inert}}(\mathbf{u},\mathbf{v})$. Eq. (20) can therefore be symbolically written as

$$\mathscr{P}(\mathbf{u},\mathbf{v}) = \mathscr{P}^{\mathrm{int}}(\mathbf{u},\mathbf{v}) - \mathscr{P}^{\mathrm{ext}}(\mathbf{u},\mathbf{v}) + \mathscr{P}^{\mathrm{inert}}(\ddot{\mathbf{u}},\mathbf{v}) = 0 \quad \forall \mathbf{v} \in \mathbf{V}_0. \tag{22}$$

We now employ the Newmark scheme to solve the nonlinear elastodynamic equation (22). At time step $(n+1)$

$$\mathscr{P}(\mathbf{u}^{n+1},\mathbf{v}) = \mathscr{P}^{\mathrm{int}}(\mathbf{u}^{n+1},\mathbf{v}) - \mathscr{P}^{\mathrm{ext}}(\mathbf{u}^{n+1},\mathbf{v}) + \mathscr{P}^{\mathrm{inert}}(\ddot{\mathbf{u}}^{n+1},\mathbf{v}) = 0 \\ \forall \mathbf{v} \in \mathbf{V}_0. \tag{23}$$

The velocity and acceleration of time step $(n+1)$ can be expressed in terms of variables of time step n:

$$\dot{\mathbf{u}}^{n+1} = -\dot{\mathbf{u}}^n + \frac{2}{\Delta t}(\mathbf{u}^{n+1} - \mathbf{u}^n), \tag{24}$$

$$\ddot{\mathbf{u}}^{n+1} = -\ddot{\mathbf{u}}^n - \frac{4}{\Delta t}\dot{\mathbf{u}}^n + \frac{4}{(\Delta t)}^{2}(\mathbf{u}^{n+1} - \mathbf{u}^n). \tag{25}$$

Substitute expression (25) into Eq. (23), and we get

$$\mathscr{P}(\mathbf{u}^{n+1},\mathbf{v}) = \mathscr{P}^{\mathrm{int}}(\mathbf{u}^{n+1},\mathbf{v}) - \mathscr{P}^{\mathrm{ext}}(\mathbf{u}^{n+1},\mathbf{v}) \\ + \mathbb{P}^{\mathrm{inert}}(\mathbf{u}^{n+1},\mathbf{u}^n,\dot{\mathbf{u}}^n,\ddot{\mathbf{u}}^n,\mathbf{v}) \\ = 0 \quad \forall \mathbf{v} \in \mathbf{V}_0, \tag{26}$$

where $\mathbb{P}^{\mathrm{inert}}(\mathbf{u}^{n+1},\mathbf{u}^n,\dot{\mathbf{u}}^n,\ddot{\mathbf{u}}^n,\mathbf{v}) = \mathscr{P}^{\mathrm{inert}}(-\ddot{\mathbf{u}}^n - \frac{4}{\Delta t}\dot{\mathbf{u}}^n + \frac{4}{(\Delta t)^2}(\mathbf{u}^{n+1} - \mathbf{u}^n), \mathbf{v})$. Eq. (26) needs to be solved for $\mathbf{u}^{n+1}$ at every time step, and then Eqs. (24) and (25) can be used to compute the velocity and the acceleration.

Eq. (26) is nonlinear with respect to $\mathbf{u}^{n+1}$ due to the nonlinearity of the term $\mathscr{P}^{\mathrm{int}}(\mathbf{u}^{n+1},\mathbf{v})$. We can solve this equation for $\mathbf{u}^{n+1}$ iteratively. Let $\mathbf{u}^{n+1,(k)}$ denote the solution at the kth iteration, and let us assume that $\mathbf{u}^{n+1,(k)}$ is close to the sought solution $\mathbf{u}^{n+1}$. So we can linearize equation (26) about $\mathbf{u}^{n+1,(k)}$ in the direction of an increment $\Delta\mathbf{u}$:

$$\mathscr{P}(\mathbf{u}^{n+1},\mathbf{v}) = \mathscr{P}(\mathbf{u}^{n+1,(k)},\mathbf{v}) + \mathbb{D}\mathscr{P}(\mathbf{u}^{n+1,(k)},\mathbf{v})(\Delta\mathbf{u}) + \cdots = \mathbf{0} \\ \forall \mathbf{v} \in \mathbf{V}_0. \tag{27}$$

In the above equation the neglected terms represent higher-order terms with respect to $\Delta\mathbf{u}$. The tangential stiffness matrix $\mathbb{D}\mathscr{P}(\mathbf{u}^{n+1},\mathbf{v})$ is determined by

$$\mathbb{D}\mathscr{P}(\mathbf{u}^{n+1},\mathbf{v})(\Delta\mathbf{u}) = \mathbb{D}\mathscr{P}^{\mathrm{int}}(\mathbf{u}^{n+1},\mathbf{v})(\Delta\mathbf{u}) - \mathbb{D}\mathscr{P}^{\mathrm{ext}}(\mathbf{u}^{n+1},\mathbf{v})(\Delta\mathbf{u}) \\ + \mathbb{D}\mathbb{P}^{\mathrm{inert}}(\mathbf{u}^{n+1},\mathbf{u}^n,\dot{\mathbf{u}}^n,\ddot{\mathbf{u}}^n,\mathbf{v})(\Delta\mathbf{u}) \\ = \int_{\Omega_0} \frac{1}{2}\left(\left(\frac{\partial \mathbf{v}}{\partial \mathbf{X}}\right)^{\mathrm{T}} \cdot \mathbf{F}(\mathbf{u}^{n+1}) + \mathbf{F}^{\mathrm{T}}(\mathbf{u}^{n+1}) \cdot \frac{\partial \mathbf{v}}{\partial \mathbf{X}}\right) : \mathscr{C}^{(4)} \\ : \frac{1}{2}\left(\left(\frac{\partial(\Delta\mathbf{u})}{\partial \mathbf{X}}\right)^{\mathrm{T}} \cdot \mathbf{F}(\mathbf{u}^{n+1}) + \mathbf{F}^{\mathrm{T}}(\mathbf{u}^{n+1}) \cdot \frac{\partial(\Delta\mathbf{u})}{\partial \mathbf{X}}\right) \mathrm{d}\Omega_0 \\ + \int_{\Omega_0} \mathbf{S}(\mathbf{u}^{n+1}) : \left(\frac{\partial(\Delta\mathbf{u})}{\partial \mathbf{X}} \cdot \frac{\partial \mathbf{v}}{\partial \mathbf{X}}\right) \mathrm{d}\Omega_0 \\ + \frac{4}{(\Delta t)^2} \int_{\Omega_0} \rho_0 (\Delta\mathbf{u}) \cdot \mathbf{v}\,\mathrm{d}\Omega_0. \tag{28}$$

Since we consider the deformation-independent external loads only in this paper, there is no contribution to the tangential stiffness matrix from the external loads, i.e. $\mathbb{D}\mathscr{P}^{\mathrm{ext}}(\mathbf{u}^{n+1},\mathbf{v}) = 0$. The linear equation (27) (with respect to $\Delta\mathbf{u}$) indicates that if the known solution at the kth iteration does not satisfy the principle of virtual work, i.e. $\mathscr{P}(\mathbf{u}^{n+1,(k)},\mathbf{v}) \neq 0$, we must find a change in $\mathscr{P}$ in the direction of $\Delta\mathbf{u}$. Therefore, we arrive at the following Newton–Raphson iterative procedure to solve for $\mathbf{u}^{n+1}$:

Loop over k until convergence

(1) Solve the following equation for $\Delta\mathbf{u}$ with a linear equation solver:

$$\mathbb{D}\mathscr{P}(\mathbf{u}^{n+1,(k)},\mathbf{v})(\Delta\mathbf{u}) = -\mathscr{P}(\mathbf{u}^{n+1,(k)},\mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{V}_0; \tag{29}$$

(2) Update solution, $\mathbf{u}^{n+1,(k+1)} = \mathbf{u}^{n+1,(k)} + \Delta\mathbf{u}$, and index, $k = k + 1$.

To discretize Eq. (29) in space, we expand $\Delta\mathbf{u}$ in terms of the shape functions of Section 2:

$$(\Delta u)_i(\mathbf{X},t) = \sum_{p=1}^{N_m} (\Delta\hat{u})_{ip}(t)\phi_p(\mathbf{X}), \quad i = 1,2,3, \tag{30}$$

where $(\Delta\hat{u})_{ip}$ are the expansion coefficients. Then Eq. (29) is transformed into the following system of linear equations:

$$\left(\frac{4}{(\Delta t)^2}\mathbf{M} + \mathbf{K}\right)\Delta\mathbf{U} = \mathbf{R}, \tag{31}$$

which can be solved with an iterative linear equation solver. In the above equation, $\Delta\mathbf{U}$ is a vector of the expansion coefficients $(\Delta\hat{u})_{ip}$, with the same ordering as that in linear elastodynamics (see Eq. (14)). The mass matrix $\mathbf{M}$ is given by Eq. (15) with ρ replaced by ρ_0. Similar to the formulation of linear elastodynamics, the stiffness matrix $\mathbf{K}$ is a $3N_m \times 3N_m$ matrix organized into a $N_m \times N_m$ matrix of blocks, each block being a 3×3 submatrix $\mathbf{K}_{pq}$ ($p,q = 1,\ldots,N_m$) given by

$$\mathbf{K}_{pq,ms} = \sum_{i,j,k,l=1}^{3} \int_{\Omega_0} \frac{1}{2}\left(\frac{\partial\phi_p}{\partial X_i}\mathbf{F}_{mj} + \mathbf{F}_{mi}\frac{\partial\phi_p}{\partial X_j}\right) \\ \times \mathscr{C}^{(4)}_{ijkl}\frac{1}{2}\left(\frac{\partial\phi_q}{\partial X_k}\mathbf{F}_{sl} + \mathbf{F}_{sk}\frac{\partial\phi_q}{\partial X_l}\right)\mathrm{d}\Omega_0 \\ + \delta_{ms}\sum_{i,j=1}^{3}\int_{\Omega_0}\mathbf{S}_{ij}\frac{\partial\phi_q}{\partial X_i}\frac{\partial\phi_p}{\partial X_j}\,\mathrm{d}\Omega_0, \\ m,s = 1,2,3, \tag{32}$$

where $\mathbf{F}_{ij}$, $\mathbf{S}_{ij}$, $\mathscr{C}^{(4)}_{ijkl}$ and $\mathbf{X}_i$ are the components of the deformation tensor $\mathbf{F}$, second Green–Kirchhoff stress tensor $\mathbf{S}$, elasticity tensor $\mathscr{C}^{(4)}$, and the position vector $\mathbf{X}$, respectively. δ_{ms} is the Kronecker delta function. The right-hand side (RHS) of Eq. (31), $\mathbf{R}$, is the residual vector with length $3N_m$, with the same ordering as $\Delta\mathbf{U}$. Its elements can be represented by $\mathbf{R}_{mp}$ ($m = 1,2,3$ and $p = 1,\ldots,N_m$). Corresponding to Eq. (22) there are three sources of contribution to the residual vector

$$\mathbf{R} = \mathbf{R}^{\text{int}} + \mathbf{R}^{\text{ext}} + \mathbf{R}^{\text{inert}}, \tag{33}$$

where $\mathbf{R}^{\text{int}}$ represents the contribution from the internal stress

$$\mathbf{R}^{\text{int}}_{mp} = -\sum_{i,j=1}^{3} \int_{\Omega_0} \mathbf{S}_{ij} \frac{1}{2}\left(\frac{\partial \phi_p}{\partial \mathbf{X}_i} \frac{\partial \mathbf{u}_m^{n+1,(k)}}{\partial \mathbf{X}_j} + \frac{\partial \mathbf{u}_m^{n+1,(k)}}{\partial \mathbf{X}_i} \frac{\partial \phi_p}{\partial \mathbf{X}_j} \right) \mathrm{d}\Omega_0, \quad m = 1,2,3, \tag{34}$$

$\mathbf{R}^{\text{ext}}$ represents the contribution from the external loads

$$\mathbf{R}^{\text{ext}}_{mp} = \int_{\partial\Omega_{0N}} \mathbf{T}_m \phi_p \,\mathrm{d}\Gamma + \int_{\Omega_0} \rho_0 \mathbf{f}_m \phi_p \,\mathrm{d}\Omega_0, \quad m = 1,2,3 \tag{35}$$

and $\mathbf{R}^{\text{inert}}$ represents the contribution from the inertia

$$\mathbf{R}^{\text{inert}}_{mp} = \int_{\Omega_0} \rho_0 \left(\ddot{\mathbf{u}}^n_m + \frac{4}{\Delta t}\dot{\mathbf{u}}^n_m + \frac{4}{(\Delta t)^2}\mathbf{u}^n_m \right) \phi_p \,\mathrm{d}\Omega_0 - \frac{4}{(\Delta t)^2} \times \int_{\Omega_0} \rho_0 \mathbf{u}_m^{n+1,(k)} \phi_p \,\mathrm{d}\Omega_0, \quad m = 1,2,3. \tag{36}$$

In Eqs. (34)–(36) we have used the symbol $\mathbf{A}_m$ ($m = 1,2,3$) to denote the three components of a vector $\mathbf{A}$.

Remarks: (1) The above scheme is second-order accurate in time, which will be demonstrated with an analytical solution in Section 5.

(2) The Newton–Raphson iterative scheme typically converges to the machine accuracy within only a few iterations. For the test problems in Sections 5 and 6 the typical number of Newton–Raphson iterations is 3–8.

(3) In the implementation a Schur complement is employed to condense out the interior modes in Eq. (31), which is then solved with some linear equation solver such as the conjugate gradient solver. Thus there exist two levels of iterations in the solution process within a time step: Newton–Raphson iteration at the outer level and conjugate gradient iteration at the inner level.

(4) Initial conditions. The initial acceleration field $\ddot{\mathbf{u}}^0(\mathbf{X})$ is required for computing the residual vector $\mathbf{R}$; see Eq. (36). It can be computed based on the initial displacement ($\mathbf{u}^0$) and velocity ($\dot{\mathbf{u}}^0$) fields (provided by initial conditions) by solving the following equation:

$$\mathscr{P}(\mathbf{u}^0, \mathbf{v}) = \mathscr{P}^{\text{int}}(\mathbf{u}^0, \mathbf{v}) - \mathscr{P}^{\text{ext}}(\mathbf{u}^0, \mathbf{v}) + \mathscr{P}^{\text{inert}}(\ddot{\mathbf{u}}^0, \mathbf{v}) = 0 \quad \forall \mathbf{v} \in \mathbf{V}_0. \tag{37}$$

Expand $\ddot{\mathbf{u}}^0$ in terms of the shape functions

$$\ddot{\mathbf{u}}^0(\mathbf{X}) = \sum_{p=1}^{N_m} \hat{\ddot{\mathbf{u}}}^0_p \phi_p(\mathbf{X}), \tag{38}$$

where $\hat{\ddot{\mathbf{u}}}^0_p$ are the expansion coefficients. We substitute the above expression into Eq. (37) and obtain

$$\mathbf{M}\hat{\ddot{\mathbf{U}}}^0 = \mathbf{R}^0, \tag{39}$$

where $\hat{\ddot{\mathbf{U}}}^0$ is a vector of the expansion coefficients, with the same ordering as $\Delta\mathbf{U}$ of Eq. (31), and the vector $\mathbf{R}^0$ on RHS is given by

$$\mathbf{R}^0_{mp} = \int_{\partial\Omega_{0N}} \mathbf{T}_m \phi_p \,\mathrm{d}\Gamma + \int_{\Omega_0} \rho_0 \mathbf{f}_m \phi_p \,\mathrm{d}\Omega_0 - \int_{\Omega_0} \mathbf{S}_{ij} \times \frac{1}{2}\left(\frac{\partial \phi_p}{\partial \mathbf{X}_i} \frac{\partial \mathbf{u}^0_m}{\partial \mathbf{X}_j} + \frac{\partial \mathbf{u}^0_m}{\partial \mathbf{X}_i} \frac{\partial \phi_p}{\partial \mathbf{X}_j} \right) \mathrm{d}\Omega_0, \quad m = 1,2,3, \; p = 1,\ldots,N_m, \tag{40}$$

where $\mathbf{R}^0_{mp}$ are the elements of $\mathbf{R}^0$ and $\mathbf{u}^0_m$ are the components of the initial displacement $\mathbf{u}^0$.

(5) Time-dependent Dirichlet boundary conditions. If the displacement is provided as a time-dependent function on some boundary, $\mathbf{u}(\mathbf{X},t)|_{\partial\Omega_{0D}} = \mathbf{u}_D(\mathbf{X},t)$, then $\dot{\mathbf{u}}|_{\partial\Omega_{0D}}$ and $\ddot{\mathbf{u}}|_{\partial\Omega_{0D}}$ will be required when solving Eq. (31). To be consistent with the discretization scheme, we employ Eqs. (24) and (25) to compute $\dot{\mathbf{u}}|_{\partial\Omega_{0D}}$ and $\ddot{\mathbf{u}}|_{\partial\Omega_{0D}}$, which retains the second-order temporal accuracy for problems involving time-dependent Dirichlet boundary conditions.

4. Parallelization

The method in Section 3 has been parallelized through domain decomposition and message passing interface (MPI). To facilitate parallel processing the mesh of polymorphic elements has been partitioned into N_p (N_p is the number of processors in the computation) disjoint subdomains (groups of elements) using METIS [24], and each processor carries out computations on the group of elements of a subdomain. In contrast to the master/slave models employing parallel virtual machine (PVM) (see e.g. [17,28]), in the current mode of parallel processing there is no manager (or master) process that plays a more prominent role (e.g. starting computation, distributing data, assembling data). All processes are cooperating peers that communicate with one another during the computation.

All data structures are distributed across the processors, including the stiffness/mass matrices, load and solution vectors, as well as auxiliary data structures such as the boundary conditions. In particular, for Schur complement only the *local elemental* Schur-complemented stiffness matrices (to be exact, a combination of the stiffness and mass matrices; see Eq. (31)) are stored. This is made possible by the modal index ordering outlined in Section 3, i.e. boundary modes (vertex/edge/face modes) followed by interior modes. As a result, the global Schur-complemented system can be assembled from the local elemental Schur-complemented system, that is

$$\mathbf{K}_b - \mathbf{K}_c\mathbf{K}_i^{-1}\mathbf{K}_c^{\mathrm{T}} = \mathscr{A}_b^{\mathrm{T}}[\mathbf{K}^e_b - \mathbf{K}^e_c(\mathbf{K}^e_i)^{-1}(\mathbf{K}^e_c)^{\mathrm{T}}]\mathscr{A}_b, \tag{41}$$

where $\mathbf{K}_b$, $\mathbf{K}_c$ and $\mathbf{K}_i$ are submatrices of the global stiffness matrix $\begin{bmatrix} \mathbf{K}_b & \mathbf{K}_c \\ \mathbf{K}_c^{\mathrm{T}} & \mathbf{K}_i \end{bmatrix}$, respectively representing global boundary–boundary coupling, boundary–interior coupling, and interior–interior interactions. Note that $\mathbf{K}_i$ is a block diagonal matrix with each block corresponding to the matrix of an elemental interior–interior modal interactions The left-hand side (LHS) of Eq. (41) is the global Schur-complemented coefficient matrix (for boundary modes). On the RHS of Eq. (41) $\mathscr{A}_b^{\mathrm{T}}$ is the boundary assembly matrix, assembling the global boundary degrees of freedom from the local boundary degrees of freedom, and $\mathscr{A}_b$ scatters global boundary degrees of freedom to the local boundary degrees of freedom. $\mathbf{K}^e_b$, $\mathbf{K}^e_c$ and $\mathbf{K}^e_i$ are submatrices of the elemental stiffness matrix, $\begin{bmatrix} \mathbf{K}^e_b & \mathbf{K}^e_c \\ (\mathbf{K}^e_c)^{\mathrm{T}} & \mathbf{K}^e_i \end{bmatrix}$, respectively representing local elemental boundary–boundary coupling, boundary–interior coupling, and interior–interior interactions. Only the elemental Schur-complemented coefficient matrix, $\mathbf{K}^e_b - \mathbf{K}^e_c(\mathbf{K}^e_i)^{-1}(\mathbf{K}^e_c)^{\mathrm{T}}$, needs to be stored in the implementation.

The above distributed data structure is crucial to achieving high performance. Note that different processors carry out the same computation steps, but on different groups of elements (i.e. single program multiple data). Within a time step, except for the calculation of the residual vector norm in the Newton iteration and the conjugate gradient iteration for solving the boundary modes of the increment displacement fields, all computations are local to each processor, involving no inter-processor communications. These include, for example, computations of the tangential stiffness matrix, its Schur complement, the load vector, the interior modes of the increment displacement fields, and the matrix–vector multiplication inside the conjugate gradient iteration. No assembly of the global matrix or the global equation system is needed with the distributed data structures.

Inter-processor communications are involved predominantly in the parallel solution of the Schur-complemented linear equation system for the boundary modes, i.e. Eqs. (19) and (31), with the preconditioned conjugate gradient solver. This solver is invoked once every time step for linear elastodynamic problems, and once every Newton–Raphson iteration (within a time step) for geometrically nonlinear elastodynamic problems. It is also invoked for computing the initial acceleration field at the pre-processing stage, see Eq. (39). In the current implementation, we have employed a simple Jacobi preconditioner in the solver. Since all the vectors (and the coefficient matrix) involved in the conjugate gradient solver are distributed across the processors, each processor has only parts of the vectors. some entries of the partial vectors stored on different processors correspond to the same global degrees of freedom, which correspond to the modes residing on the boundaries of different mesh partitions that are shared by two or more processors. The typical pattern of operations in the conjugate gradient solver requiring inter-processor communications is the computation of the inner product of two distributed vectors. This type of operations, for example, is used for computing the norm of the residual vector for convergence check or for computing the coefficients to update the vectors. Inter-processor communications in such operations require special care because entries in the partial vectors on different processors corresponding to the shared global degrees of freedom may contain inconsistent (or partial) results from preceding operations. Even if these entries in partial vectors contain consistent data on different processors, the multiplicities of the shared global degrees of freedom still need to be taken into account when taking the inner product. We employ a fan-in-fan-out parallel reduction algorithm on a binary tree or hypercube [16,34] for such communications.

5. Convergence studies

In this section, we establish the spatial and temporal accuracies of the method presented in Section 3 by comparing simulation results against analytical solutions for four classes of problems: linear elastostatics, linear elastodynamics, geometrically nonlinear elastostatics, and geometrically nonlinear elastodynamics. The primary purpose is not to study the physical structural behavior, but rather to verify the consistency of the scheme and investigate its convergence behavior. Therefore, for the test problems in the following sections we have used material properties with contrived values, and paid little attention to their correspondence to the structures in physical reality. In this section and Section 6, we have also omitted all the units for physical parameters and variables when working on the test problems. We have assumed that for each test problem a system of consistent units are used for all the physical variables and parameters. We consider the St. Venant–Kirchhoff material in the following tests, and use Young's modulus E and the Poisson ratio v for the material properties for most test problems, which are related to the material properties λ and μ in Eq. (1) by

$$\mu = \frac{E}{2(1+v)}, \quad \lambda = \frac{vE}{(1+v)(1-2v)}. \tag{42}$$

5.1. Linear elastostatics

We first investigate the convergence of the scheme for linear elastostatic problems. Consider the deformation of a cubic object occupying the domain $0 \leqslant x \leqslant 1$, $0 \leqslant y \leqslant 1$ and $0 \leqslant z \leqslant 1$, with a Young's modulus E and a Poisson ratio v (Fig. 1a), and assume small deformations. The displacements on the face $x = 0$ is known

$$u_x = A\sin ay, \quad u_y = u_z = 0, \tag{43}$$

where u_x, u_y and u_z are the displacements in x-, y- and z-directions, respectively, and A and a are prescribed constants. A traction force field, $\mathbf{T} = (T_x, T_y, T_z)$, is applied on the rest of the faces, and is given by

$$\begin{aligned} &T_x = n_y E_3 (aA\cos ay + bB\cos bx), \\ &T_y = n_x E_3 (aA\cos ay + bB\cos bx), \quad T_z = 0, \end{aligned} \tag{44}$$

where $\mathbf{n} = (n_x, n_y, n_z)$ is the outward-pointing unit vector normal to the surface, B and b are prescribed constants, and $E_3 = \frac{E}{2(1+v)}$. A body force field, $\rho\mathbf{f} = (f_x, f_y, f_z)$, is applied on the object with the following functions:

$$f_x = E_3 a^2 A\sin ay, \quad f_y = E_3 b^2 B\sin bx, \quad f_z = 0. \tag{45}$$

This problem has the following analytic solution for the displacements of the object:

$$\begin{cases} u_x = A\sin ay, \\ u_y = B\sin bx, \\ u_z = 0. \end{cases} \tag{46}$$

To compute the deformation of the object with the scheme in Section 3 we discretize the domain with 5 tetrahedral elements; Thick solid lines mark the edges of the elements in Fig. 1a. We impose the Dirichlet BC on the face $x = 0$ with the displacements in Eq. (43), and the traction/Neumann BCs on the other faces with the traction forces in Eq. (44). To study the convergence behavior, we systematically vary the element order (i.e. highest polynomial degree in the expansions) from 2 to 11, and for each element order calculate the L^∞, L^2 and H^1 errors of the computed displacement fields with respect to the exact solution in Eq. (46). In Fig. 1b, we plot these errors (in logarithmic scale) as a function of the element order (in linear scale) for the problem with the following parameter values:

$$A = B = 1.0, \quad a = b = 1.0, \quad E = 1000, \quad v = 0.3.$$

The error curves show the trend of approximately a straight line, indicating that the numerical errors decreases exponentially as the element order increases.

5.2. Linear elastodynamics

We next study the spatial and temporal convergence of the scheme for linear elastodynamic problems. Consider the vibration of a cubic object ($0 \leqslant x \leqslant 1$, $0 \leqslant y \leqslant 1$, $0 \leqslant z \leqslant 1$) with a mass density ρ, a Young's modulus E and a Poisson ratio v. Assume that the deformation of the object is small throughout the time so that the linear elastic equations can be used. The displacements on the face $x = 0$ is known

$$u_x = C\sin bx, \quad u_y = u_z = 0, \tag{47}$$

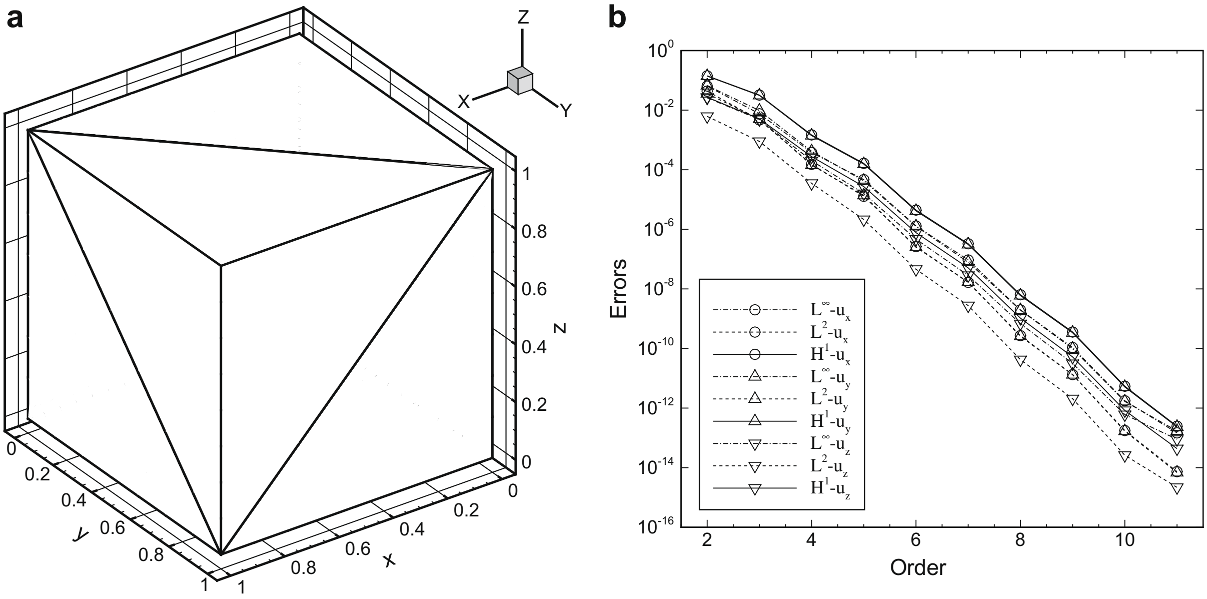

Fig. 1. Linear elastostatics: (a) cubic object discretized with 5 tetrahedral elements. (b) L^∞, L^2 and H^1 errors as a function of the element order showing spatial exponential convergence rate.

where C and b are prescribed constants. The following time-dependent traction force field is applied on the rest of the faces:

$$\begin{cases} T_x = n_x E_1 (A + B \sin at + bC \cos bx), \\ T_y = n_y E_2 (A + B \sin at + bC \cos bx), \\ T_z = n_z E_2 (A + B \sin at + bC \cos bx), \end{cases} \tag{48}$$

where A, B and a are prescribed constants, t is time, $\mathbf{n} = (n_x, n_y, n_z)$ is the outward-pointing unit vector normal to the object surface, and the parameters $E_1 = \frac{(1-\nu)E}{(1+\nu)(1-2\nu)}$ and $E_2 = \frac{\nu E}{(1+\nu)(1-2\nu)}$. The following body force field, $\rho\mathbf{f} = (f_x, f_y, f_z)$, is applied on the object:

$$f_x = E_1 b^2 C \sin bx - \rho a^2 B x \sin at, \quad f_y = f_z = 0. \tag{49}$$

At $t = 0$ the displacement and velocity fields are provided as follows:

$$\begin{cases} u_x = Ax + C \sin bx, \quad u_y = u_z = 0, \\ \dot{u}_x = Bax, \quad \dot{u}_y = \dot{u}_z = 0. \end{cases} \tag{50}$$

The above problem has the following analytic solution for the displacements of the object:

$$\begin{cases} u_x = (A + B \sin at)x + C \sin bx, \\ u_y = 0, \\ u_z = 0. \end{cases} \tag{51}$$

To simulate the time-dependent deformation of the object, we employ the method presented in Section 3.1 and discretize the domain with one hexahedral element. We impose the Dirichlet BC (Eq. (47)) on the face $x = 0$ and traction/Neumann BCs (Eq. (48)) on the other faces, and use the displacement and velocity fields from Eq. (50) as the initial conditions. In order to investigate the spatial convergence of the scheme we fix the time step size Δt, and systematically vary the element order between 2 and 8. For each element order we integrate the momentum equation (10) over time from $t = 0$ to $t = t_f$, and compute the L^∞, L^2 and H^1 errors of the computed displacement fields at $t = t_f$ with respect to the exact solution (Eq. (51)). To check the temporal accuracy of the scheme, we fix the element order and vary the time step size Δt. For each value of Δt we integrate the momentum equation (10) over time from $t = 0$ to $t = t_f$, and compute the L^∞, L^2 and H^1 errors of the displacement fields at $t = t_f$ with respect to the exact solution (Eq. (51)). Fig. 2a shows the errors of the computed results (in logarithmic scale) versus the element order (in linear scale) for a fixed time step size $\Delta t = 0.01$ for the problem with the following parameter values:

$$\begin{aligned} &A = 1.9, \quad B = 0.1, \quad C = 1.4, \quad a = 1.0, \quad b = 1.2, \\ &\rho = 1000, \quad E = 100, \quad \nu = 0.3, \quad t_f = 0.1. \end{aligned} \tag{52}$$

Evidently the numerical errors decrease exponentially as the element order increases, suggesting an exponential convergence rate of the scheme spatially. Fig. 2b plots the numerical errors versus the time step size Δt, both in logarithmic scales, for a fixed element order 3 for the problem with the following parameter values:

$$\begin{aligned} &A = 1.2, \quad B = 0.1, \quad C = 0.05, \quad a = 1.0, \quad b = 0.02, \\ &\rho = 1000, \quad E = 100, \quad \nu = 0.3, \quad t_f = 1.6. \end{aligned} \tag{53}$$

The results show that as Δt is reduced by half the numerical errors is decreased by a factor of 4, suggesting that the scheme has achieved a second-order accuracy in time.

5.3. Geometrically nonlinear elastostatics

We next investigate the convergence of the scheme for geometrically nonlinear elastostatic problems. Consider the finite deformation of the cubic object initially occupying the domain, $0 \leqslant X \leqslant 1$, $0 \leqslant Y \leqslant 1$ and $0 \leqslant Z \leqslant 1$, with a Young's modulus E and a Poisson ratio ν. It is known that the face $X = 0$ is clamped, and there is a traction force, $\mathbf{T} = (T_X, T_Y, T_Z)$, applying on the rest of the faces given by the following function:

$$\begin{cases} T_X = n_X(\lambda/2 + \mu)Aa \cos(aX)(A^2a^2 \cos^2(aX) - 1), \\ T_Y = n_Y(A^2a^2 \cos^2(aX) - 1)\lambda/2, \\ T_Z = n_Z(A^2a^2 \cos^2(aX) - 1)\lambda/2, \end{cases} \tag{54}$$

where $\mathbf{n} = (n_X, n_Y, n_Z)$ is the outward-pointing unit vector normal to the surface in the initial configuration, A and a are prescribed constants, λ and μ are the material properties in Eq. (1). There is also the following body force field on the object, $\rho_0\mathbf{f} = (f_X, f_Y, f_Z)$:

$$f_X = -(\lambda/2 + \mu)Aa^2 \sin(aX)(1 - 3A^2a^2 \cos^2(aX)), \quad f_Y = f_Z = 0. \tag{55}$$

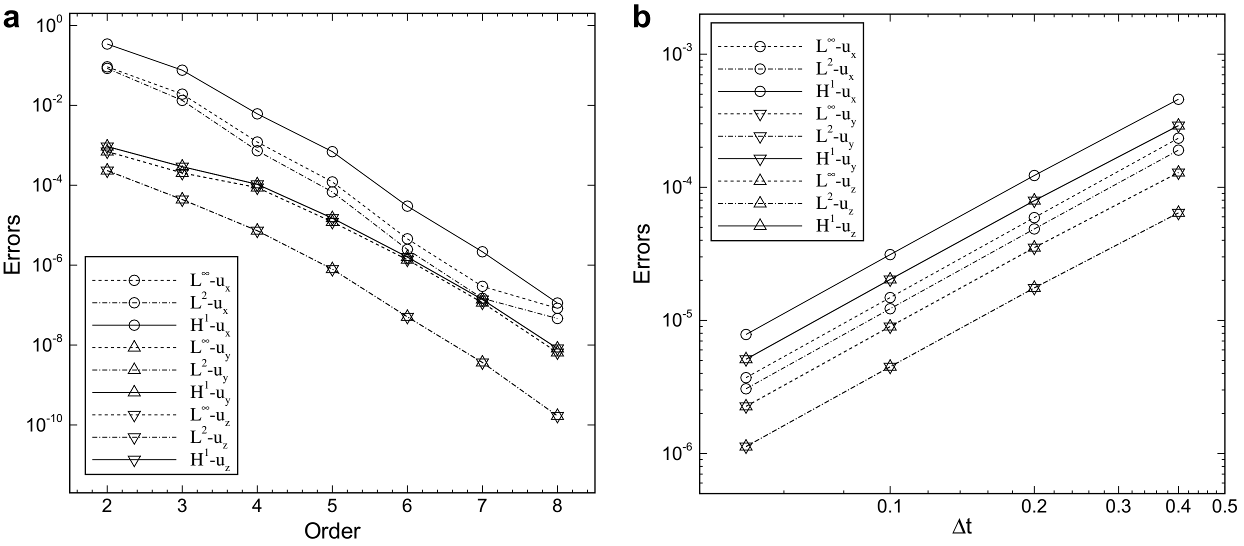

Fig. 2. Linear elastodynamics: (a) L^∞, L^2 and H^1 errors of the displacements at $t = 0.1$ versus the element order for a fixed time step size $\Delta t = 0.01$. (b) L^∞, L^2 and H^1 errors of the displacements at $t = 1.6$ versus Δt for a fixed element order 3. A single hexahedral element is used to discretize the domain.

The displacements of the object for this problem can be expressed by the following analytic functions in terms of coordinates of the initial configurations:

$$\begin{cases} u_X = A\sin(aX) - X, \\ u_Y = 0, \\ u_Z = 0, \end{cases} \tag{56}$$

where u_X, u_Y and u_Z are the displacements in x-, y- and z-directions, respectively.

We compute the displacement fields of the object by solving the weak form of the momentum equation for geometrically nonlinear problems with the scheme in Section 3.2 (omitting the inertia term). In the initial configuration we discretize the domain with 2 prismatic elements. In Fig. 3a, thick solid lines mark the edges of these elements. Dirichlet BC is imposed on face $X = 0$ and traction BCs according to Eq. (54) are imposed on the other faces. We study the convergence of the scheme by varying the element order between 2 and 10, and at each element order calculate the L^∞, L^2 and H^1 errors of the computed results against the exact solution in Eq. (56). In Fig. 3b, we plot these errors (logarithmic scale) as a function of the element order (linear scale) for the problem with the following parameter values:

$$A = 1.9, \quad a = 1.0, \quad E = 1000, \quad \nu = 0.3. \tag{57}$$

The scheme has obviously achieved a spatially exponential convergence rate for this class of problems.

5.4. Geometrically nonlinear elastodynamics

We next investigate the convergence of the scheme for geometrically nonlinear elastodynamic problems. Consider the vibration of the cubic object with the initial configuration, $0 \leqslant X \leqslant 1$, $0 \leqslant Y \leqslant 1$ and $0 \leqslant Z \leqslant 1$, and an initial mass density ρ_0, Young's

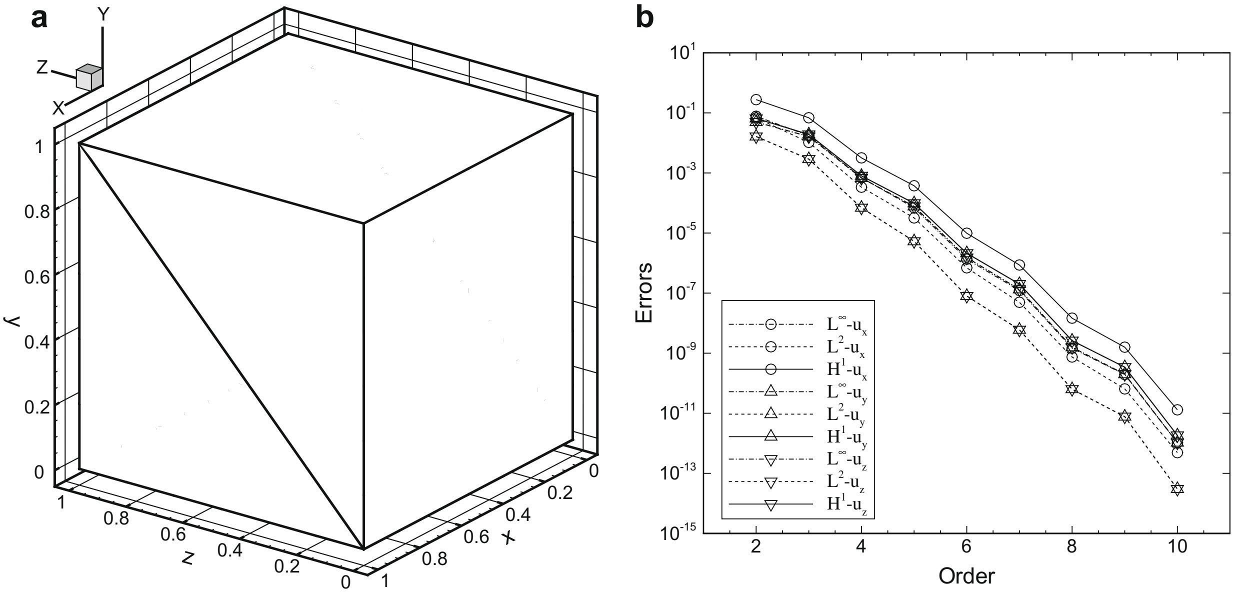

Fig. 3. Geometrically nonlinear elastostatics: (a) cubic object discretized with 2 prismatic elements. (b) L^∞, L^2 and H^1 errors versus element order demonstrating exponential convergence rate.

modulus E, and Poisson ratio ν. We assume that the deformation of the object is finite throughout the time so that geometrically nonlinear formulations of the momentum equation need to be used. The face $X = 0$ of the object is clamped, and a time-dependent traction force field is applied on the other faces, $\mathbf{T} = (T_X, T_Y, T_Z)$:

$$\begin{cases} T_X = n_X(\lambda/2+\mu)(A + B\sin(at) + Cb\cos(bX)) \\ \qquad \times((A + B\sin(at) + Cb\cos(bX))^2 - 1), \\ T_Y = n_Y((A + B\sin(at) + Cb\cos(bX))^2 - 1)\lambda/2, \\ T_Z = n_Z((A + B\sin(at) + Cb\cos(bX))^2 - 1)\lambda/2, \end{cases} \tag{58}$$

where A, B, C, a and b are prescribed constants, $\mathbf{n} = (n_X, n_Y, n_Z)$ is the outward-pointing unit vector normal to the surface. The following body force field is applied on the object, $\rho_0\mathbf{f} = (f_X, f_Y, f_Z)$:

$$\begin{cases} f_X = -Ba^2X\sin(at) - \frac{1}{\rho_0}(\lambda/2+\mu)Cb^2\sin(bX) \\ \qquad \times[1 - 3(A + B\sin(at) + Cb\cos(bX))^2], \quad t > 0, \\ f_Y = 0, \\ f_Z = 0. \end{cases} \tag{59}$$

The displacements (u_X, u_Y, u_Z) and velocity $(\dot{u}_X, \dot{u}_Y, \dot{u}_Z)$ at $t = 0$ is known

$$\begin{cases} u_X = (A-1)X + C\sin(bX), \quad u_Y = u_Z = 0, \\ \dot{u}_X = BaX, \quad \dot{u}_Y = \dot{u}_Z = 0. \end{cases} \tag{60}$$

With these conditions the problem has the following analytic solution ($t \geqslant 0$) for the displacements of the object in terms of coordinates of the initial configuration:

$$\begin{cases} u_X = (A - 1 + B\sin(at))X + C\sin(bX), \\ u_Y = 0, \\ u_Z = 0. \end{cases} \tag{61}$$

To simulate the time-dependent finite deformation of the object, we employ the scheme in Section 3.2 and discretize the domain with 5 tetrahedral elements (see Fig. 1a). A Dirichlet BC is employed on the face $X = 0$, and traction/Neumann BCs (Eq. (58)) are imposed on the rest of the faces. To study the spatial convergence of the scheme, we fix the time step size Δt, and systematically vary the element order between 2 and 9. At each order we integrate the momentum equation (20) over time from $t = 0$ to $t = t_f$, and compute the L^∞, L^2 and H^1 errors of the displacement fields at $t = t_f$ against the exact solution (see Eq. (61)). To study the temporal convergence, we fix the element order and systematically vary the time step size Δt. For each value of Δt we integrate the momentum equation (20) over time from $t = 0$ to $t = t_f$, and compute the L^∞, L^2 and H^1 errors of the displacement fields at $t = t_f$ against the exact solution. In Fig. 4a, we plot the errors of the computed results (logarithmic scale) as a function of the element order (linear scale) for a fixed $\Delta t = 0.01$ for the problem with the following parameter values:

$$\begin{aligned} &A = 1.2, \quad B = 0.1, \quad C = 1.0, \quad a = 1.0, \quad b = 1.4, \\ &\rho_0 = 1.0, \quad E = 1000, \quad \nu = 0.3, \quad t_f = 0.2. \end{aligned} \tag{62}$$

Evidently, the numerical errors decrease exponentially with increasing element order, indicating a spatially exponential convergence rate. In Fig. 4b, we plot the errors as a function of Δt, with both axis in logarithmic scales, for a fixed element order 4 for a problem with the following parameter values:

$$\begin{aligned} &A = 1.2, \quad B = 0.2, \quad C = 0.1, \quad a = 1.0, \quad b = 0.01, \\ &\rho_0 = 1.0, \quad E = 100, \quad \nu = 0.3, \quad t_f = 0.2. \end{aligned} \tag{63}$$

As the time step size Δt is reduced by half, the numerical errors are reduced by a factor of 4, suggesting that the scheme has a second-order accuracy in time for geometrically nonlinear elastodynamic problems.

6. Test problems

In this section, we apply the method in Section 3 to solve several test problems of linear/geometrically nonlinear elastostatics/dynamics. We will demonstrate the capability of the method for situations involving large deformations, and the parallel performance of the method for up to more than 2000 processors. We assume that the materials follow the St. Venant–Kirchhoff constitutive equation.

6.1. Deformation of a prismatic bar

We first consider the deformation of a prismatic bar. The purpose is to demonstrate the differences between linear and

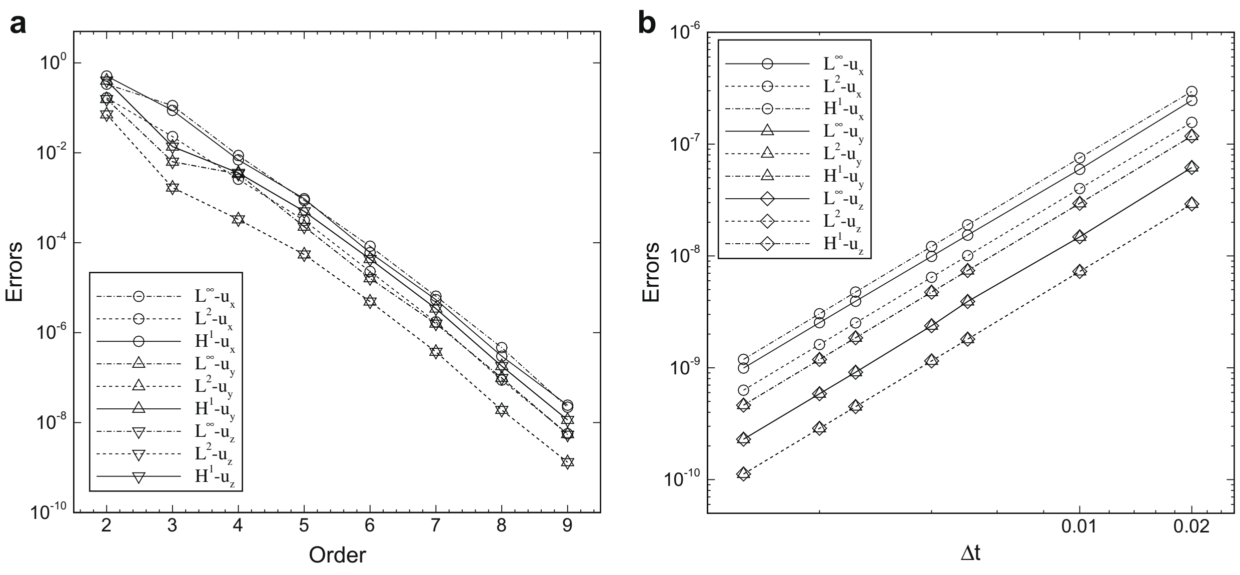

Fig. 4. Geometrically nonlinear elastodynamics: (a) L^∞, L^2 and H^1 errors of the displacements at $t = 0.2$ as a function of the element order for a fixed time step size $\Delta t = 0.01$. (b) L^∞, L^2 and H^1 errors of the displacements at $t = 0.2$ as a function of Δt for a fixed element order 4. Five tetrahedral elements are used to discretize the domain.

geometrically nonlinear elastostatic solutions of the problem under identical conditions.

Fig. 5a illustrates the configuration of the bar in a deformed state. In the initial undeformed state, the straight prismatic bar has an axial length $L_x = 3.0$, and the cross-section of the bar is an equilateral triangle with a side dimension 1.0. The triangular face $\overline{ADE}$ (vertex D is invisible in Fig. 5a) is aligned with the plane $x = 0$, and vertex E is located on the y-axis. The quadrilateral face $\overline{ABCD}$ is initially aligned with the plane $y = 0$. The left face of the bar, $x = 0$, is clamped, and on the right face, $\overline{BCF}$, a traction force is applied in the $-y$-direction, $T_y = -F_y$, where F_y is the force magnitude.

We study the deformation of the bar under two situations: (1) assuming small displacement so that the linear elastostatic equations can be used, and (2) assuming finite deformation so that geometrically nonlinear elastostatic equations need to be considered. We discretize the initial domain of the object with two identical prismatic elements in the x-direction. The thick solid lines in Fig. 5a mark the edges of these elements. We impose the Dirichlet BC on the face $\overline{ADE}$ (zero displacements), a traction BC on the face $\overline{BCF}$, and a traction-free BC on the rest of the object surfaces.

To ensure that the computation results are converged, we systematically increase the element order from 2 to 13 for a fixed force magnitude $F_y = 1.0$. In Fig. 5b, we plot the x and y displacements (z displacement is zero) of the vertex F as a function of the element order from the linear and geometrically nonlinear elastostatic solutions with a Young's modulus $E = 1000$ and a Poisson ratio $\nu = 0.3$. As the element order increases, both the x and y displacements, for both linear and geometrically nonlinear solutions, increase quite significantly initially, and level off gradually as the order further increases. The displacements remain essentially unchanged beyond the element order 6. While the difference in the y displacement between the converged linear and geometrically nonlinear solutions for this traction force magnitude is minor, the deviation in the x displacements is quite substantial. The x displacement from the geometrically nonlinear solution is notably smaller than that from the linear solution.

We next compare the linear and geometrically nonlinear solutions under identical traction forces. Fig. 5c shows the x and y displacements of vertex F, from both linear and geometrically nonlinear solutions, as a function of the traction force magnitude F_y for a fixed element order 12. Both x and y displacements from the linear solution increase linearly with increasing traction force, obviously unphysical under large force magnitudes. On the other hand, compared to the linear solution the y displacement from the geometrically nonlinear solution is about the same as the force magnitude F_y is low, and becomes comparatively smaller as F_y becomes large ($F_y > 2.5$). The x displacement of the geometrically nonlinear solution exhibits a trend qualitatively different from that of the linear solution. As the force magnitude increases, the x displacement of the nonlinear solution increases initially, reaching a maximum value at some point, and decreases as F_y further in-

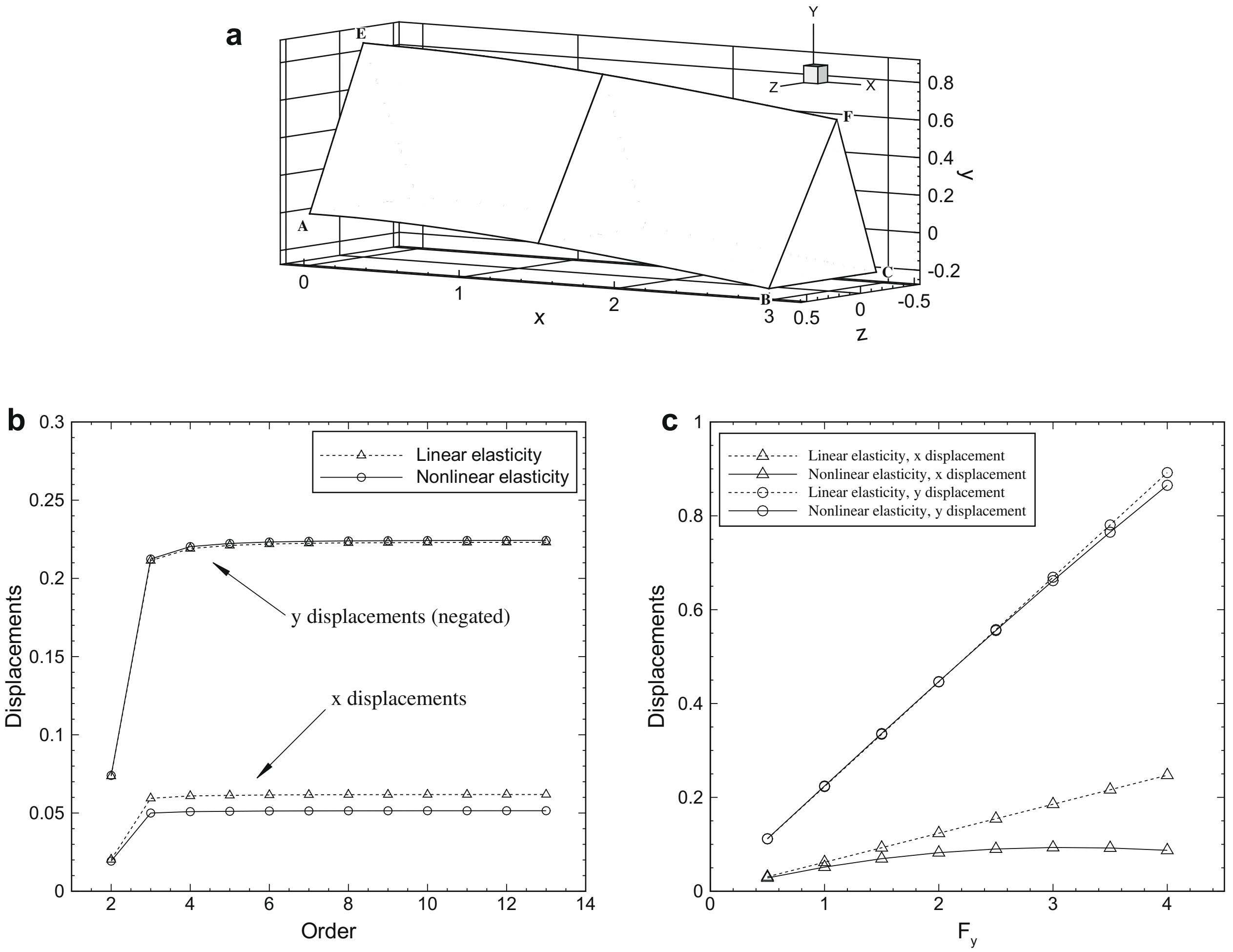

Fig. 5. Comparison of linear and geometrically nonlinear solutions for elastostatic problems: (a) Prismatic bar configuration (deformed). Left face $x = 0$ is clamped while a traction force is applied on the right face $\overline{BCF}$. (b) Displacements of vertex F versus the element order for a fixed traction force magnitude 1.0. (c) Displacements of vertex F as a function of the traction force magnitude for a fixed element order 12. y displacement has been made negative in plots (b) and (c).

creases. It is lower than that of the linear solution for the entire range of F_y values studied here.

6.2. Free vibration of a rectangular board

In the second test problem we study the free vibration of a three-dimensional long rectangular board. The purpose is to demonstrate the differences between linear and geometrically nonlinear solutions for elastodynamic problems under otherwise identical conditions.

Fig. 6 shows the board at its equilibrium position (thin solid lines) and at the initial configuration for dynamic simulations (thick solid lines). At equilibrium, the board has a length $L_x = 4.0$ in the x-direction, a width $L_y = 0.6$ in the y-direction, and a thickness $L_z = 0.2$ in the z-direction. The left face of the board, $x = 0$, is clamped. We assume that at equilibrium the board has a mass density, $\rho_0 = 100$, a Young's modulus, $E = 1000$, and a Poisson ratio, $\nu = 0.3$. The initial configuration of the board for the dynamic simulations is generated with the following steps:

1. At the equilibrium position apply a traction force in z-direction, $T_z = 0.2$, on the right face of the board, $x = L_x$;
2. Compute the deformation of the board using the elastostatic solver. Employ the linear elastostatic solver in this step if the initial configuration is for linear elastodynamic simulations, and employ the geometrically nonlinear elastostatic solver in this step if the initial configuration is for geometrically nonlinear elastodynamic simulations;
3. Use the deformed state of the board as the initial configuration for the dynamic simulations.

For the dynamic simulations, the board is assumed to be at rest with the initial configuration for $t < 0$. At $t = 0$ the board is released from its initial configuration and allowed to freely vibrate. We study the vibration of the board under two situations: (1) assuming small displacement throughout the time so that linear elastodynamic equations can be used, and (2) assuming finite deformation so that geometrically nonlinear elastodynamic equations need to be employed. We discretize the domain of the board at equilibrium with 8 identical hexahedral elements (see Fig. 6) in the x-direction. Dirichlet BC (zero displacements) is applied to the left face of the board, and traction-free BCs are imposed on the other faces of the board. The initial displacements are prescribed based on the initial configuration of the board, with zero initial velocities.

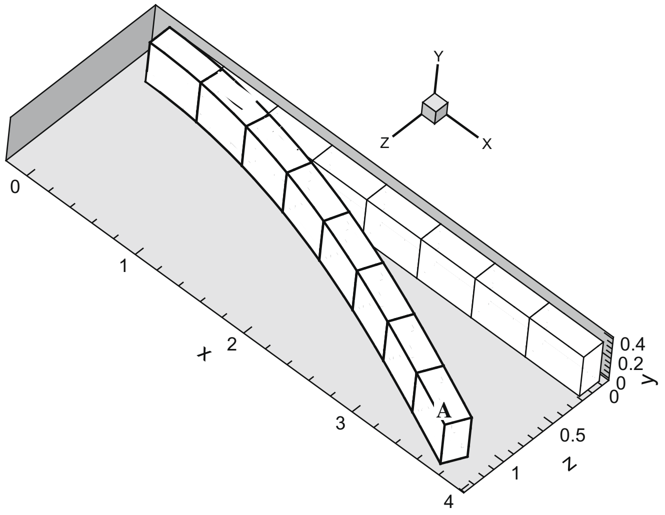

Fig. 6. Free vibration of a rectangular board. Thick solid lines mark the initial configuration for dynamic simulations. Thin solid lines mark the equilibrium position.

In Fig. 7a, we plot the time histories of the z displacements of the vertex A (see Fig. 6) from the linear and geometrically nonlinear simulations. The two signals have been shifted in time so that they are aligned. The results are obtained using an element order 5 and a time step size $\Delta t = 0.005$. Simulations have also been conducted using an element order 4, and we observe no significant difference in the results from those of order 5. We have also done simulations using time step sizes $\Delta t = 0.01$ and 0.05, and observed that the highest frequencies of the vibration are not well resolved with $\Delta t = 0.05$, and only marginally resolved with $\Delta t = 0.01$. An interesting result from Fig. 7a is that the linear and the geometrically nonlinear elastodynamic simulations result in the same essential frequency in this problem. The vibration amplitude from the geometrically nonlinear solution is observed to be slightly lower than that from the linear solution.

The spectral content of the displacement signals reveals additional characteristics of the vibrations. We have calculated the FFTs of the two z displacement signals in Fig. 7a corresponding to the linear and geometrically nonlinear solutions, and computed their power spectra. In Fig. 7b, we plot the power spectral densities of the two signals as a function of the frequency. The overlap of the primary peaks of the two signals confirms that they have the same basic frequency. Compared to that of the linear solution, the spectrum of the geometrically nonlinear solution demonstrate significantly more peaks at high frequencies, and clusters of discrete peaks can also be observed around certain frequencies, which is likely a manifestation of the nonlinear interactions between different frequency components.

We next investigate the temporal and spectral characteristics of the x displacements. In Fig. 8a, we plot the time histories of the x displacements of the vertex A (see Fig. 6) of the linear and geometrically nonlinear solutions, which have also been shifted in time for alignment. The linear and the geometrically nonlinear solutions exhibit qualitatively different characteristics. The x displacement signal of the linear solution is essentially a sinusoidal function. In contrast, the x displacement signal of the nonlinear solution demonstrates an additional frequency component with twice the basic frequency. Under the linear elasticity assumption, as the board is displaced in the $+z$-direction the x displacement of vertex A increases with increasing z displacement. And as the board is displaced in the $-z$-direction the x displacement of vertex A decreases with increasing z displacement magnitude. Therefore, one observes essentially a sinusoidal signal from the linear solution. For the geometrically nonlinear solution, as the board is displaced in the $+z$-direction, however, with increasing z displacement the x displacement of vertex A increases initially, reaching a maximum at some point, and then decreases as the z displacement increases further. And as the board is displaced in the $-z$-direction, the x displacement of vertex A decreases with increasing z displacement magnitude. As a result, the x displacement reaches the maximum twice within a period, which is why an additional high-frequency component can be observed in the nonlinear solution.

Fig. 8b shows a comparison of the power spectra of the x displacement signals of the linear and geometrically nonlinear solutions. It can be observed again that the linear and nonlinear solutions yield the same basic frequency for this problem. The nonlinear solution exhibits a significantly more complicated spectrum. In addition to the second main frequency, we observe a substantially larger number of peaks at high frequencies and also clusters of spectral peaks around certain frequencies, similar to that of the z displacement. The cluster of peaks around some frequencies becomes so dense that the spectrum appears nearly continuous.

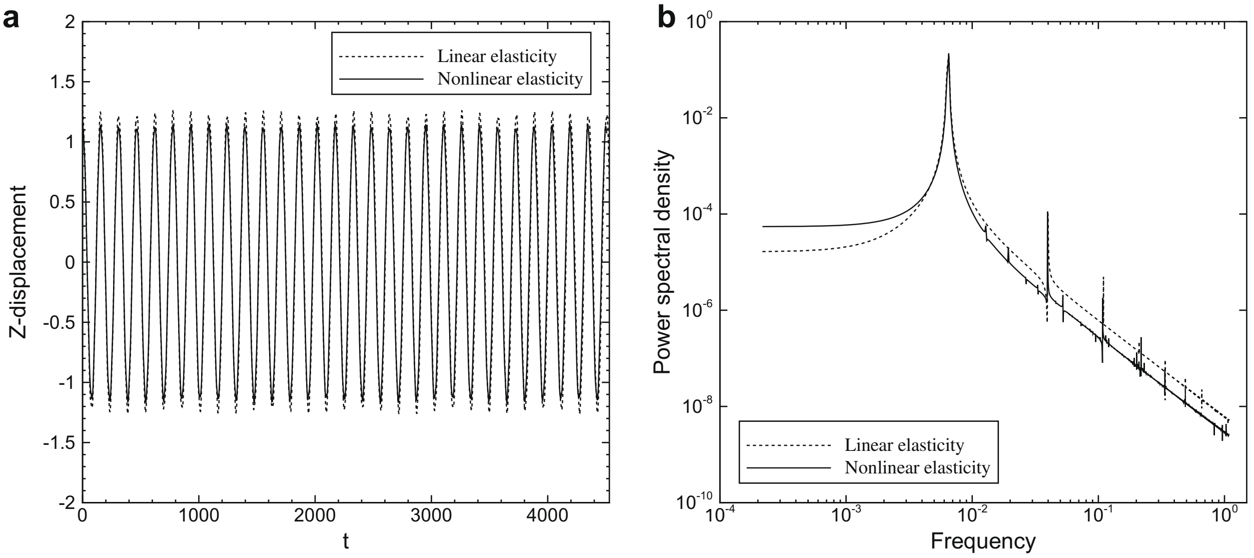

Fig. 7. Comparison of linear and geometrically nonlinear solutions for the free vibration of a rectangular board: (a) time histories of z displacements of vertex A from linear and nonlinear solutions. (b) Comparison of the power spectra of the z displacement signals in (a).

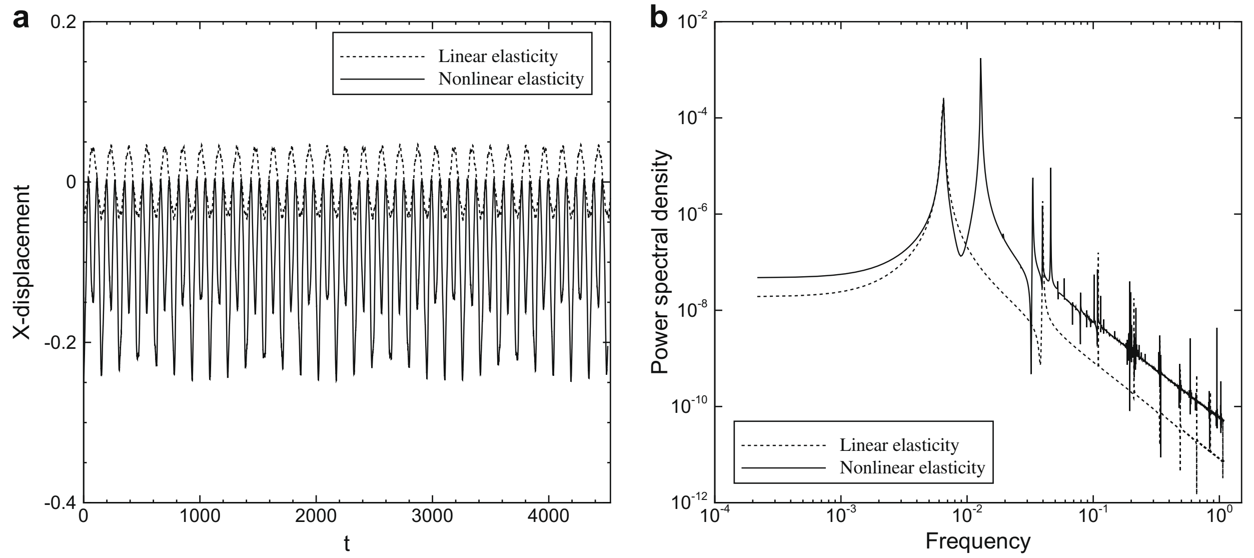

Fig. 8. Comparison of linear and geometrically nonlinear solutions for the free vibration of a rectangular board: (a) time histories of x displacements of vertex A from linear and nonlinear solutions. (b) Comparison of the power spectra of the x displacement signals in (a).

6.3. Large deformation of a rectangular bar

In this problem we attempt to demonstrate the capability of the geometrically nonlinear solver for problems involving large deformations. We consider the large deformation of a three-dimensional rectangular bar under multi-stage traction forces. In its initial configuration (see Fig. 9), the bar has a length $L_x = 4.0$ in the x-direction, a width $L_y = 0.6$ in the y-direction, and a thickness $L_z = 0.2$ in the z-direction. The left end of the bar is in the plane $x = 0$, and is clamped. The back surface of the bar is initially in the plane $z = 0$, and the bottom surface is initially in the plane $y = 0$. The bar is assumed to have a Young's modulus $E = 1000$ and a Poisson ratio $\nu = 0.3$. Consider the following three-stage traction forces applying on the free end (right end) of the bar:

1. First apply a constant traction force in the z-direction, T_z, to the free end of the bar at the initial configuration.
2. In the deformed configuration at the end of the previous stage, apply a constant traction force in the $-x$-direction, $-T_x$, to the free end of the bar.
3. In the deformed configuration at the end of the previous stage, apply a constant traction force in the $-z$-direction, $-T'_z$, to the free end of the bar.

Fig. 9 shows the final and intermediate configurations, together with the initial configuration, of the bar under such a three-stage force with values

$$T_z = 1.0, \quad -T_x = -2.0, \quad -T'_z = -11.0. \tag{64}$$

The domain has been discretized with 8 hexahedral elements in the x-direction, and the element order 4 is used in the computations. It is evident that the bar has undergone extremely large deformations under these traction forces. These results demonstrate that the high-order scheme presented in Section 3.2 is capable of handling problems involving very large deformations.

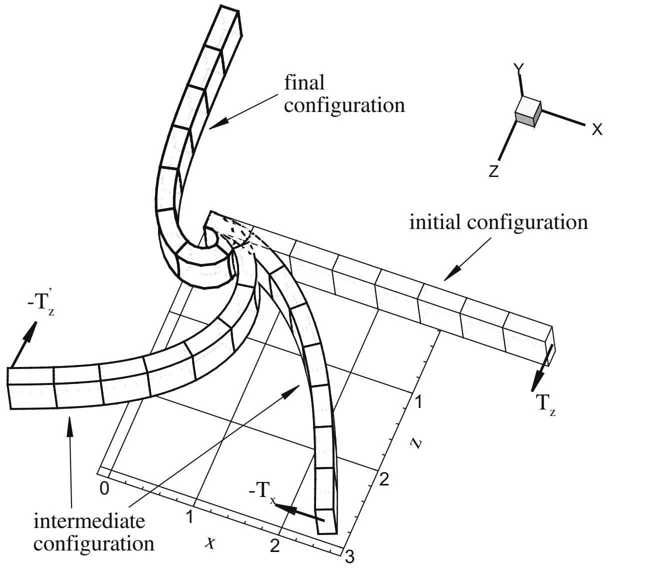

Fig. 9. Large deformation of a rectangular bar under multi-stage forces. Left end of the bar is clamped.

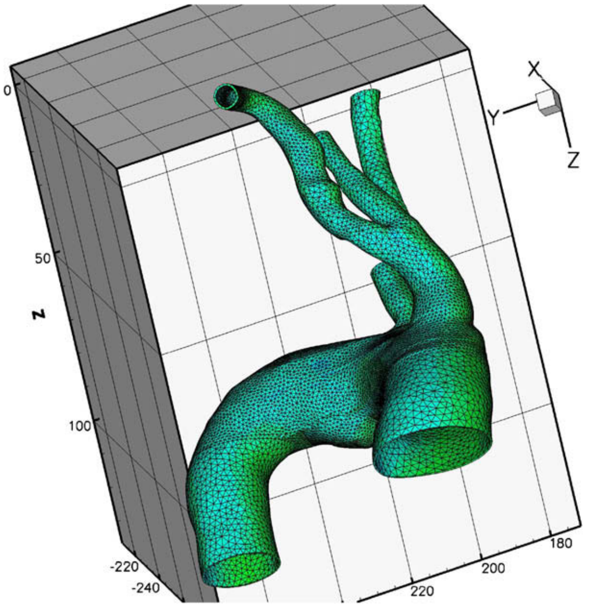

Fig. 10. Human aorta: Mesh consisting of 22,664 prismatic elements. Color contours show the distribution of wall stress component σ_{xy}.

6.4. Parallel performance – wall deformation of the human aorta

All the test problems considered so far involve simple meshes with small numbers of elements. We next apply the method together with massively parallel computing to solve a test problem involving realistic geometries with a significantly increased number of elements. We consider the deformation of the wall of the human aorta. The primary purpose here is to demonstrate the efficiency of the parallelized method and its capability in tackling large-scale structural problems with massively parallel computers. So we have chosen the material properties and the problem setup primarily based on numerical convenience, and made little effort concerning their correspondence to physical realities.

Fig. 10 shows the wall mesh around the bifurcation of the human aortic artery, and the color contours in the plot show the distribution of the wall stress component σ_{xy}. The geometry of this problem was extracted from patient-specific computed tomography (CT) images [9]. The mesh consists of 22,664 prismatic elements, and the wall has been assumed to have a Young's modulus 1000 and a Poisson ratio 0.3. A traction force, $\mathbf{T} = (T_x, T_y, T_z)$ has been applied on the inner surface of the wall. The surfaces at all the outlets are assumed to be traction-free, and the outer surface of the wall is assumed to be clamped. We have also assumed small deformations for the arterial wall in this test. We realize that the above conditions and assumptions by no means reflect the physical or physiological reality. However, the problem provides a realistic example for studying the performance of the parallel solver.

Under the above conditions and assumptions we solve this linear elastostatic problem with the parallelized method (see Sections 3 and 4). We employ an element order 9 (for all elements) so that

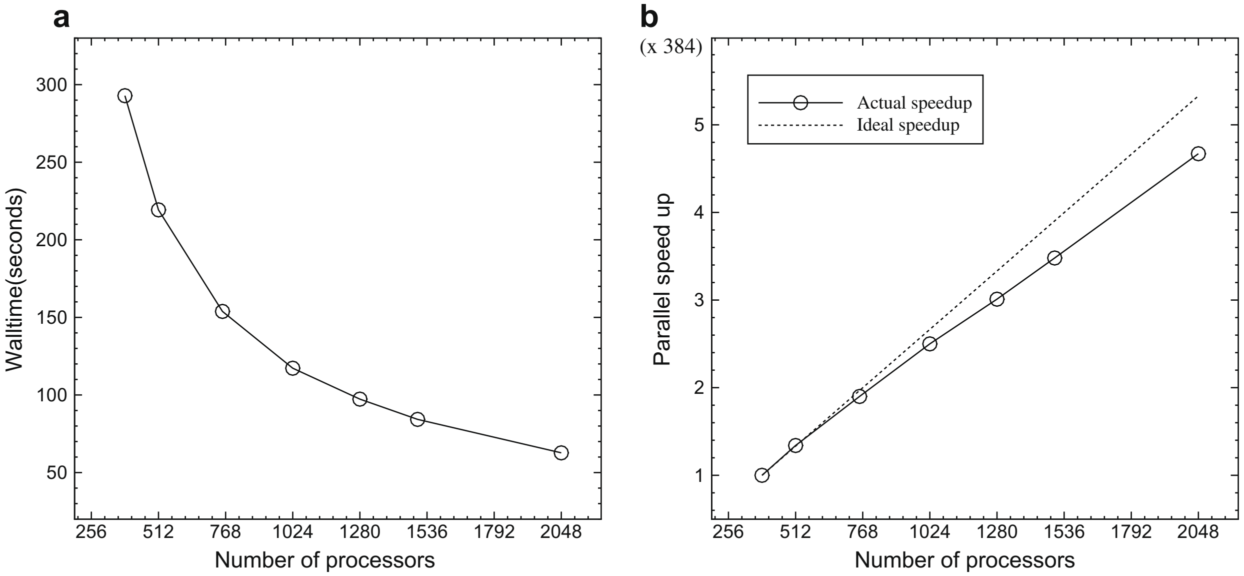

Fig. 11. Parallel performance (fixed problem size): (a) wall time versus the number of processors, and (b) parallel speedup versus the number of processors.

the overall problem size is fixed, and systematically vary the number of processors in the computations. Fig. 11a shows the wall clock time of the computations as a function of the number of processors (between 384 and 2048). (384 is the minimum number of processors that can handle this problem size in terms of memory requirement.) This is for the traction force values $T_x = T_y = 0.001$ and $T_z = 0$. The wall time here refers to the total simulation time, including the computations of the stiffness matrix and the load vector, and the solution of linear systems. The timing data was collected on the Cray XT3 (AMD Opteron processor, 2.6 GHz) machine at Pittsburgh Supercomputing Center (PSC) under dedicated mode with exclusive access to the allocated compute nodes. At least three independent runs have been conducted at each data point, and the average wall time from these runs are used. It is evident from Fig. 11a that the wall clock time is reduced substantially and consistently as the number of processors increases in the computations.

Fig. 11b shows the parallel speedup as a function of the number of processors, which is a more intuitive presentation of the information contained in Fig. 11a. Parallel speedup S_p is defined bY $S_p = \frac{\Delta T_1}{\Delta T_p}$, where ΔT_1 is the wall time it takes to solve the problem on one processor, and ΔT_p is the wall time it takes to solve the same problem on p processors. Since the minimum number of processors for this problem size is 384, we have used the wall time it takes on 384 processors as the baseline time when computing the speedup in Fig. 11b, and correspondingly multiplied the speedup by a coefficient 384. Within 1024 processors the actual speedup is quite close to the ideal speedup. Beyond 1024 processors, the curve of the actual speedup exhibits the trend of nearly a straight line. These results suggest that the parallelized method has achieved quite high parallel performance.

7. Concluding remarks

In this paper, we have presented a high-order method (p-version) employing Jacobi polynomial-based hierarchical shape functions, as an alternative to the typical Legendre polynomial-based shape functions in solid mechanics, for solving three-dimensional geometrically nonlinear elasticity problems. The Jacobi polynomial-based approach provides a unified treatment for polymorphic elements, and the resulting method can handle all commonly encountered types of elements (hexahedrons, tetrahedrons, prisms/pentahedrons, pyramids). We have employed the Newmark scheme to discretize the dynamic three-dimensional geometrically nonlinear momentum equations, and a Newton–Raphson iterative scheme within a time step. The method has been parallelized with domain decomposition and message passing interface, and the parallelized method has been scaled to over 2000 processors with high parallel performance. By comparison with analytic solutions, we have demonstrated that the method has an exponential convergence rate spatially and a second-order accuracy in time for the four classes of problems of linear elastostatics, linear elastodynamics, geometrically nonlinear elastostatics, and geometrically nonlinear elastodynamics.

Acknowledgments

The first author gratefully acknowledges the support from US NSF and the DOE/PSAAP program. Computer time was provided by the US TeraGrid through an MRAC grant and by the Rosen Center for Advanced Computing at Purdue University. The authors would like to thank members of the CRUNCH group at Brown University, especially Prof. G.E. Karniadakis and Dr. L. Grinberg, for help and useful discussions.

References

[1] Armero E, Romero I. On the formulation of high-frequency dissipative time-stepping algorithms for nonlinear dynamics. Part II: Second-order methods. Comput Methods Appl Mech Eng 2001;190:6783–824.

[2] Babuska I, Elman HC, Markley K. Parallel implementation of the hp-version of the finite element method on a shared-memory architecture. SIAM J Sci Statist Comput 1992;13(6):1433–59.

[3] Bathe KJ. Finite element procedures. Prentice Hall; 1996.

[4] Bathe KJ. Conserving energy and momentum in nonlinear dynamics: a simple implicit time integration scheme. Comput Struct 2007;85:437–45.

[5] Bathe KJ, Cimento AP. Some practical procedures for the solution of nonlinear finite element equations. Comput Methods Appl Mech Eng 1980;22:59–85.

[6] Bittencourt ML, Vazquez MG, Vazquez TG. Construction of shape functions for the h- and p-versions of the FEM using tensorial product. Int J Numer Methods Eng 2007;71:529–63.

[7] Dey S, Datta DK. A parallel hp-FEM infrastructure for three-dimensional structural acoustics. Int J Numer Methods Eng 2006;68:583–603.

[8] Dong S. Direct numerical simulation of turbulent Taylor–Couette flow. J Fluid Mech 2007;587:373–93.

[9] Dong S, Insley J, Karonis NT, Papka M, Binns J, Karniadakis GE. Simulating and visualizing human arterial system on the TeraGrid. Future Gener Comput Syst 2006;22:1011–7.

[10] Dong S, Karniadakis GE. DNS of flow past a stationary and oscillating cylinder at *Re* = 10,000. J Fluid Struct 2005;20:519–31.

[11] Dong S, Karniadakis GE, Ekmekci A, Rockwell D. A combined DNS–PIV study of the turbulent near wake. J Fluid Mech 2006;569:185–207.

[12] Dong S, Triantafyllou GS, Karniadakis GE. Elimination of vortex streets in bluff-body flows. Phys Rev Lett 2008;100:204501.

[13] Dubiner M. Spectral methods on triangles and other domains. J Sci Comput 1991;6:345.

[14] Düster A, Hartmann S, Rank E. p-FEM applied to finite isotropic hyperelastic bodies. Comput Methods Appl Mech Eng 2003;192:5147–66.

[15] Düster A, Rank E. A p-version finite element approach for two- and three-dimensional problems of the J_2 flow theory with non-linear isotropic hardening. Int J Numer Methods Eng 2002;53:49–63.

[16] Foster IT. Designing and building parallel programs. Addison-Wesley; 1995.

[17] Guo X, Myers K. A parallel solver for the hp-version of the finite element methods. Comput Methods Appl Mech Eng 1996;133:229–46.

[18] Heisserer U. Solution of the semidiscrete equations of structural dynamics by the generalized-α method and its implementation in a p-FEM code. PhD thesis, Technical University of Munich, Institute of Computer Sciences in Civil Engineering, Munich, Germany; 2001.

[19] Heisserer U, Hartmann S, Düster A, Yosibash Z. On volumetric locking-free behavior of p-version finite elements under finite deformations. Commun Numer Methods Eng; in press.

[20] Holzer SM, Yosibash Z. The p-version of the finite element method in incremental elasto-plastic analysis. Int J Numer Methods Eng 1996;39:1859–78.

[21] Humar JL. Dynamics of structures. A.A. Balkema Publishers; 2002.

[22] Janik TJ. Parallel implementation of the p-version of the finite element method for elliptic equations on a shared-memory architecture. Adv Comput Math 1998;8:97–110.

[23] Karniadakis GE, Sherwin SJ. Spectral/hp element methods for computational fluid dynamics. 2nd ed. New York (NY), USA: Oxford University Press; 2005.

[24] Karypis G, Kumar V. A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM J Sci Comput 1998;20:359–92.

[25] Laszloffy A, Long J, Patra AK. Simple data management, scheduling and solution strategies for managing the irregularities in parallel adaptive hp finite element simulations. Parallel Comput 2000;26:1765–88.

[26] Noel AT, Szabó BA. Formulation of geometrically non-linear problems in the spatial reference frame. Int J Numer Methods Eng 1997;40:1263–80.

[27] Oden JT, Patra A, Feng Y. Parallel domain decomposition solver for adaptive hp finite element methods. SIAM J Numer Anal 1997;34:2090–118.

[28] Rank E, Rücker M, Düster A, Bröker H. The efficiency of the p-version finite element in a distributed computing environment. Int J Numer Methods Eng 2001;52:589–604.

[29] Ribeiro P. A p-version first order shear deformation finite element for geometrically non-linear vibration of curved beams. Int J Numer Methods Eng 2004;61:2696–715.

[30] Scholz D, Düster A, Rank E. Model-adaptive structural FEM computations for fluid–structure interaction. In: Third MIT conference on computational fluid and solid mechanics, Cambridge, USA; 2005.

[31] Scholz D, Kollmannsberger S, Düster A, Rank E. Thin solids for fluid–structure interaction. Lect Notes Comput Sci Eng 2006;53:294–335.

[32] Szabó BA, Babuška I. Finite element analysis. New York: John Wiley & Sons; 1991.

[33] Szabó BA, Düster A, Rank E. The p-version of the finite element method. In: Hughes TJR, Stein E, de Borst R, editors. Encyclopedia of computational mechanics, vol. I. John Wiley & Sons; 2004. p. 119–39.

[34] Tufo HM, Fischer PF. Fast parallel direct solvers for coarse grid problems. J Parallel Distr Comput 2001;61:151–77.

[35] Yosibash Z, Hartmann S, Heisserer U, Duester A, Rank E, Szanto M. Axisymmetric pressure boundary loading for finite deformation analysis using p-FEM. Comput Methods Appl Mech Eng 2007;196:1261–77.