

Adaptive Computations Using Material Forces and Residual-Based Error Estimators on Quadtree Meshes

A. Tabarraei, N. Sukumar*

*Department of Civil and Environmental Engineering, University of California,
One Shields Avenue, Davis, CA 95616.*

Abstract

Quadtree is a hierarchical data structure that is well-suited for h -adaptive mesh refinement. Due to the presence of hanging nodes, classical shape functions are non-conforming on quadtree meshes. In this paper, we use natural neighbor basis functions to construct conforming interpolants on quadtree meshes. To this end, the recently proposed construction of polygonal basis functions is adapted to quadtree elements. A fast technique for calculating stiffness matrix on quadtree meshes is introduced. Residual-based error estimators and material force technique are used to estimate the error on quadtree meshes. The performance of the adaptive technique is demonstrated through the solution of linear and nonlinear boundary-value problems.

Key words: Laplace interpolant, centroidal Voronoi tessellation, quadtree mesh, hanging nodes, configurational forces, error estimation.

1 Introduction

In this paper we use quadtree meshes for h -adaptive mesh refinement. The quadtree data structure provides a fast and efficient method for h -adaptivity based on the concept of geometric subdivision of elements. In a quadtree mesh, if the error in an element exceeds a prescribed tolerance, the element is recursively subdivided into four equal elements. As shown in Fig. 1, the subdivision of an element leads to the generation of hanging nodes (nodes a , b , c , and d in Figs. 1b and 1c) over the element edges if the new elements and their adjacent elements are not of the same size. Due to the presence of these hanging nodes, incompatibilities arise in classical finite element approximations. Special techniques have been used to construct conforming approximations over

* Corresponding author. Email: nsukumar@ucdavis.edu

a revival in the past few years [11, 21–27]. Natural neighbor-based (Laplace) basis functions have been recently used to construct conforming interpolants on polygonal meshes [11]. This technique is described below and is used in Section 3 to construct C^0 admissible interpolants on quadtree meshes.

The natural neighbor interpolations are based on the concept of Voronoi cells and its dual Delaunay triangulation. Consider a set of nodes that are used to discretize a domain $\Omega_0 \subset \mathbf{R}^2$. In Fig. 2a, the Voronoi cells and the Delaunay triangles are shown. The Voronoi cell of a node consists of all points that are closer to that node than to any other node. The Delaunay triangulation is a triangulation (subdivision of an area into triangles) of the convex hull such that the circumcircle of every triangle is an empty circle [28]. Delaunay triangulation can be constructed by connecting the nodes that have a common Voronoi cell edge. In Fig. 2b, a point p is inserted inside the convex hull. If point p lies inside the circumcircle of a Delaunay triangle, all the vertices of the Delaunay triangle are the natural neighbors of the point. In Fig. 2b, nodes 1, 2, 5 and 6 are natural neighbors of point p . By considering point p as a new node of the domain, its Voronoi cell can be constructed in the same way the Voronoi cell of the other nodes are constructed. In Fig. 2c, the Voronoi cells of the domain nodes and point p are presented. Using Fig. 2c, the Laplace interpolant at point p is defined as [29]

$$\phi_a(\boldsymbol{\xi}) = \frac{\alpha_a(\boldsymbol{\xi})}{\sum_{b=1}^n \alpha_b(\boldsymbol{\xi})}, \quad \alpha_a(\boldsymbol{\xi}) = \frac{s_a(\boldsymbol{\xi})}{h_a(\boldsymbol{\xi})}, \quad \boldsymbol{\xi} \in \Omega_0, \quad (1)$$

where $\alpha_a(\boldsymbol{\xi})$ is the Laplace weight function, $s_a(\boldsymbol{\xi})$ is the length of the common edge between Voronoi cell of point p and node a , and $h_a(\boldsymbol{\xi})$ is the Euclidean distance between point p and node a . Laplace basis functions are non-negative, interpolate, and are linearly complete:

$$0 \leq \phi_a(\boldsymbol{\xi}) \leq 1, \quad \phi_a(\boldsymbol{\xi}_b) = \delta_{ab}, \quad \sum_{a=1}^n \phi_a(\boldsymbol{\xi}) = 1, \quad \mathbf{x} \equiv \mathbf{x}(\boldsymbol{\xi}) = \sum_{a=1}^n \phi_a(\boldsymbol{\xi}) \mathbf{x}_a, \quad (2)$$

where δ_{ab} is the Kronecker-delta.

The Laplace interpolant is piece-wise linear on the boundary of the domain and it also satisfies the Kronecker-delta property. Hence essential boundary conditions can be directly imposed in a Galerkin method. As in the classical finite element method, the shape functions are first defined on the reference elements and then through an isoparametric mapping, the shape functions over physical elements are constructed. The polygonal reference elements are regular n -gons (Fig. 3). In a regular n -gons all the vertex nodes lie on the same circumcircle, and hence all the nodes of the element are natural neighbors of any interior point of the reference element. Closed-form expressions for Laplace shape functions on reference elements are presented in Reference [12].

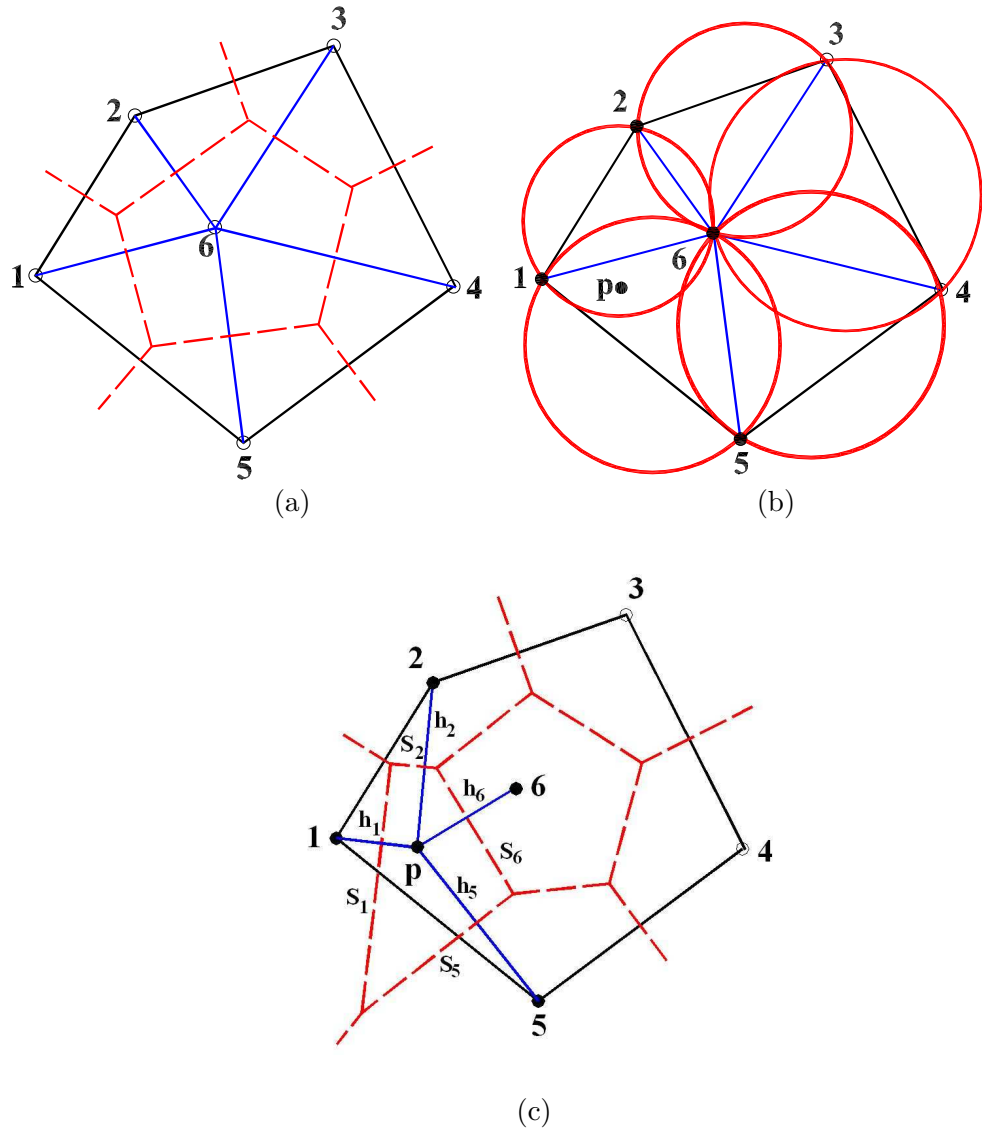


Fig. 2. (a) Voronoi tessellation and resulting Delaunay triangulation; (b) Delaunay triangles and Delaunay circumcircles and (c) Voronoi cell of point p and construction of Laplace basis functions.

In Fig. 4, the Voronoi cells of point p inside a regular pentagonal and hexagonal element are shown. By using an isoparametric mapping from the reference element to the physical element, the shape functions and their derivative over the physical element can be found. The mapping from a hexagonal reference element to a six-noded polygonal element is illustrated in Fig. 5. Since the mapping is isoparametric, the Laplace interpolant remains linear on the edges of the physical element. The linearity of the Laplace interpolant on the boundaries of the physical element leads to conformity of the basis functions on a polygonal mesh. Laplace shape functions reduce to barycentric coordinates on triangular elements and to classical bilinear shape functions over quadrilateral elements [30]. Numerical integration on elements with more than four nodes is

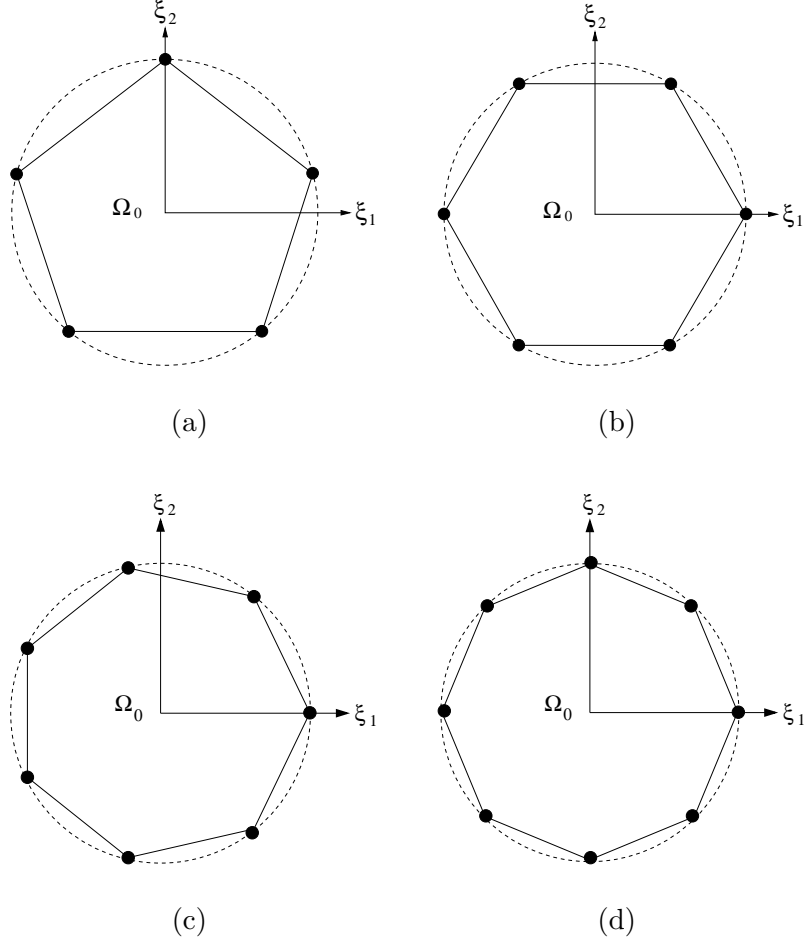


Fig. 3. Reference elements. (a) Pentagon; (b) Hexagon; (c) Heptagon; and (d) Octagon.

not yet well-developed. In Fig. 6, the numerical integration scheme on n -gons ($n > 4$) used in this paper is illustrated. For the purpose of numerical integration the reference polygonal element is sub-divided into subtriangles. The triangular reference element is used as the reference integration element. Two mappings are needed to find the Gauss point location in the physical element. First, the Gauss point position in the polygonal reference element is found by using an affine map from the reference triangular element to the corresponding subtriangle in the reference polygonal element. After finding the Gauss point location in the polygonal reference element, the isoparametric mapping given in Eq. (2) is used to find the Gauss point position in the physical element. The numerical integration procedure can be expressed as [11]:

$$\int_{\Omega_e} f d\Omega = \int_{\Omega_0} f |\mathbf{J}_2| d\Omega = \sum_{j=1}^n \int_{\Omega_0^{\Delta_j}} f |\mathbf{J}_2| d\Omega = \sum_{j=1}^n \int_0^1 \int_0^{1-\xi} f |\mathbf{J}_1^j| |\mathbf{J}_2| d\xi d\eta. \quad (3)$$

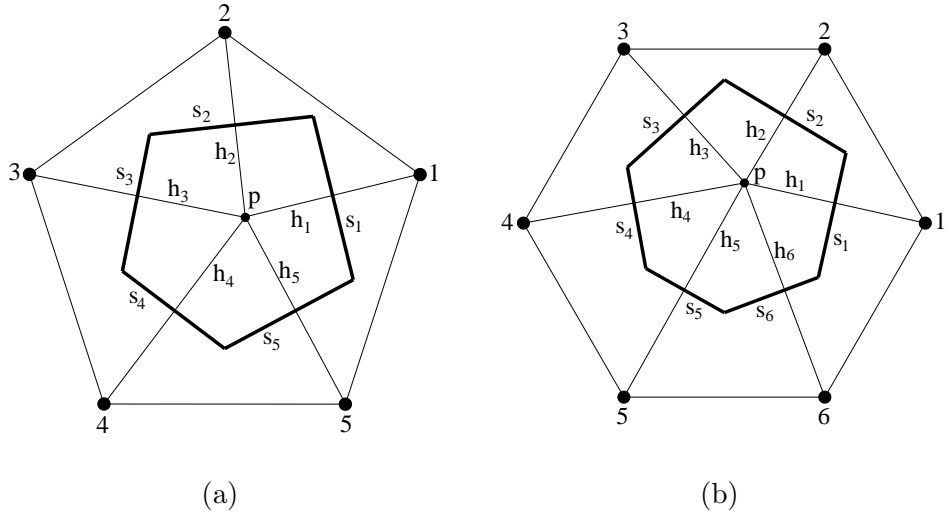


Fig. 4. Voronoi cell of point p inside (a) a pentagonal reference element; and (b) a hexagonal reference element.

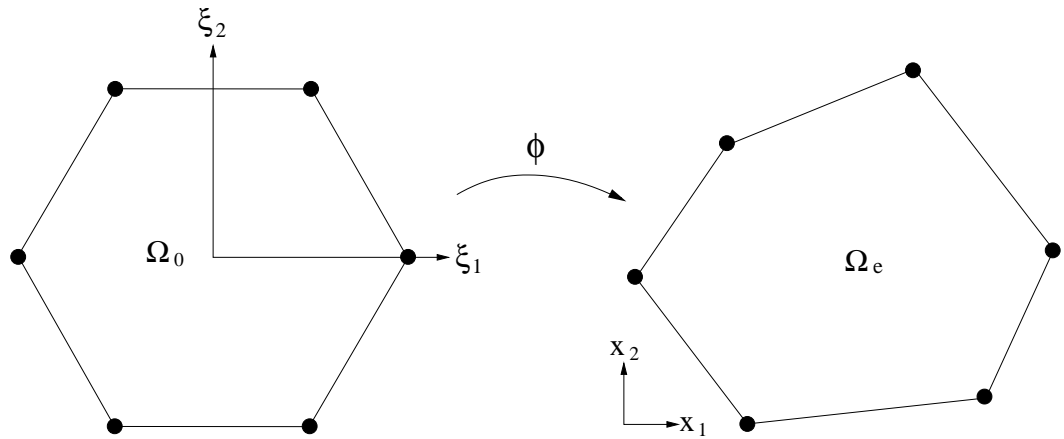


Fig. 5. Isoparametric mapping from a hexagonal reference element to a six-noded physical element.

2.1 Polygonal mesh generation

Polygonal meshes can be generated by tessellating the domain into Voronoi cells. For this purpose, first a set of initial random points called generators are inserted within the domain. By constructing the Voronoi diagram of the random generators, the polygonal mesh is obtained. Meshes with better quality can be obtained by using centroidal Voronoi diagram. In a centroidal Voronoi diagram, the Voronoi cell generators are also the centroid of the Voronoi cell.

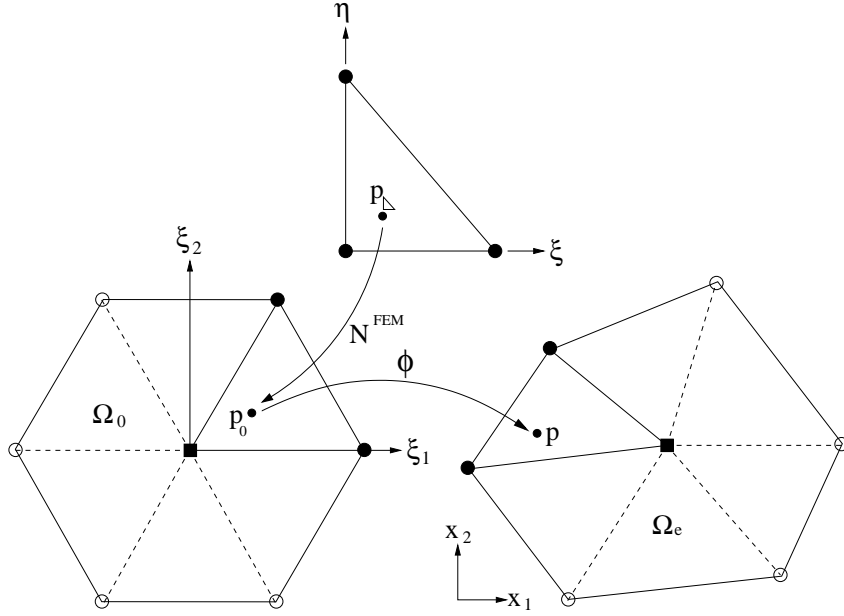


Fig. 6. Numerical integration scheme based on the partition of the reference element.

The centroid of a Voronoi cell is defined as

$$\mathbf{C}_i = \frac{\int_{A_i} \mathbf{x} \rho(\mathbf{x}) dV}{\int_{A_i} \rho(\mathbf{x}) dV}, \quad (4)$$

where A_i is the area of Voronoi cell, \mathbf{x} is the position and $\rho(\mathbf{x})$ is a density function. If $\rho(\mathbf{x})$ is a constant, then the centroid coincides with the mass center. Centroidal Voronoi cells can be produced by employing Lloyd's algorithm [31, 32]. This algorithm is defined as:

- (1) Select an initial set of generator points \mathbf{x}_i ;
- (2) Construct the Voronoi diagram of \mathbf{x}_i ;
- (3) Find the centroid \mathbf{C}_i of each Voronoi cell using Eq. (4);
- (4) If \mathbf{C}_i converges to \mathbf{x}_i stop; otherwise use \mathbf{C}_i as the new set of \mathbf{x}_i ; go to step 2.

The uniform polygonal meshes of Fig. 7 are constructed by choosing a constant density function. A public-domain package [33] is used to construct the centroidal Voronoi generators. The meshes in the left column of Fig. 7 are obtained from the initial set of random generators. The uniform final meshes of Fig. 7 are obtained by utilizing Lloyd's algorithm on the initial meshes. The second type of meshes displayed in Fig. 8 are constructed by using non-constant density functions. The acceptance-rejection method [34] is used to select samples from the nonuniform density function. The initial and centroidal meshes are shown in Fig. 8. As expected, the meshes are refined in the areas with a higher probability of having a generator point. This feature can be used for mesh refinement on polygonal meshes.

Table 1

Relative error in the L^2 norm for the displacement patch test on uniform polygonal meshes.

Number of nodes	Relative error in the L^2 norm	
	Initial mesh	Centroidal mesh
22	4.62×10^{-8}	5.06×10^{-9}
42	4.68×10^{-9}	3.56×10^{-10}
102	3.49×10^{-9}	2.46×10^{-10}
202	2.03×10^{-9}	1.73×10^{-10}
1000	1.03×10^{-9}	1.12×10^{-10}

Table 2

Relative error in the L^2 norm for the displacement patch test on nonuniform polygonal meshes.

$\rho(\mathbf{x})$	Relative error in the L^2 norm	
	Initial mesh	Centroidal mesh
$e^{2x_1+2x_2}$	8.54×10^{-9}	6.81×10^{-10}
$e^{-20(x_1-\frac{1}{2})^2-20(x_2-\frac{1}{2})^2}$	1.92×10^{-8}	1.56×10^{-9}

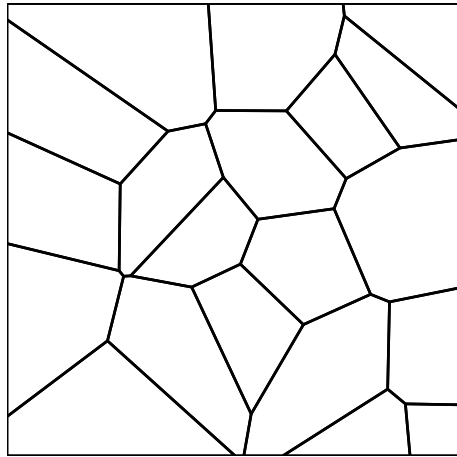
The performance of polygonal finite elements in nonlinear elasticity is verified by conducting a patch test on the meshes displayed in Figs. 7 and 8. The material is Neo-Hookean and the patch test is performed by applying the following essential boundary condition on the boundary of the domain:

$$u_1 = u_2 = x_1 + x_2.$$

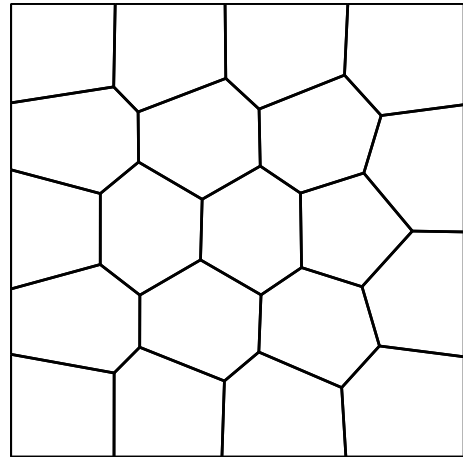
For the purpose of error estimation and the convergence study, the L^2 norm of the displacement error is used:

$$\|\mathbf{u} - \mathbf{u}^h\|_{L^2} = \left(\int_{\Omega} [(\mathbf{u} - \mathbf{u}^h) \cdot (\mathbf{u} - \mathbf{u}^h)] d\Omega \right)^{\frac{1}{2}}, \quad (5)$$

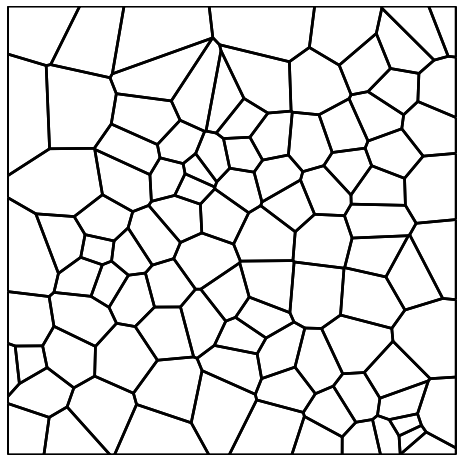
where \mathbf{u} and \mathbf{u}^h are the exact and numerical solutions, respectively. The L^2 norm of the displacement error of the uniform meshes are presented in Table 1. Relative errors of $\mathcal{O}(10^{-8})$ and $\mathcal{O}(10^{-9})$ in the L^2 norm are obtained for the initial and centroidal meshes, respectively. The result of the patch test on nonuniform meshes are presented in Table 2. We observe that the L^2 norm of the error in the displacement of centroidal Voronoi meshes is one order less than the error on the initial meshes.



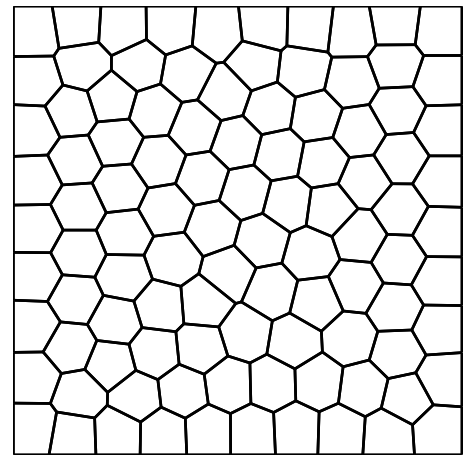
(a)



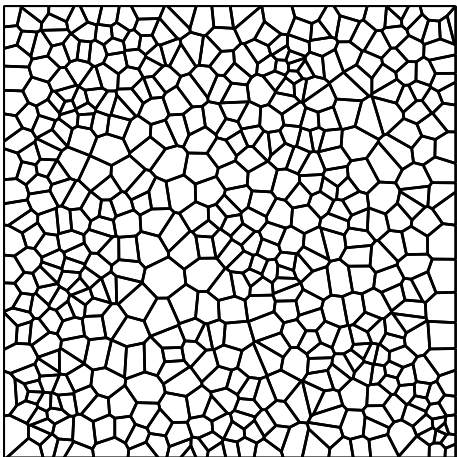
(b)



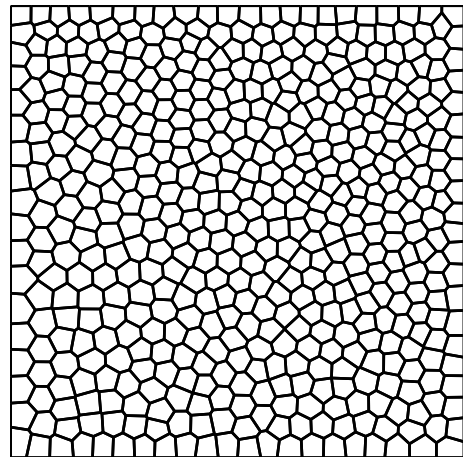
(c)



(d)



(e)



(f)

Fig. 7. Patch test on uniform polygonal meshes. Left: Initial meshes; right: centroidal meshes. Top: 20 generators (42 nodes); Middle: 100 generators (202 nodes); and bottom: 500 generators (1000 nodes)

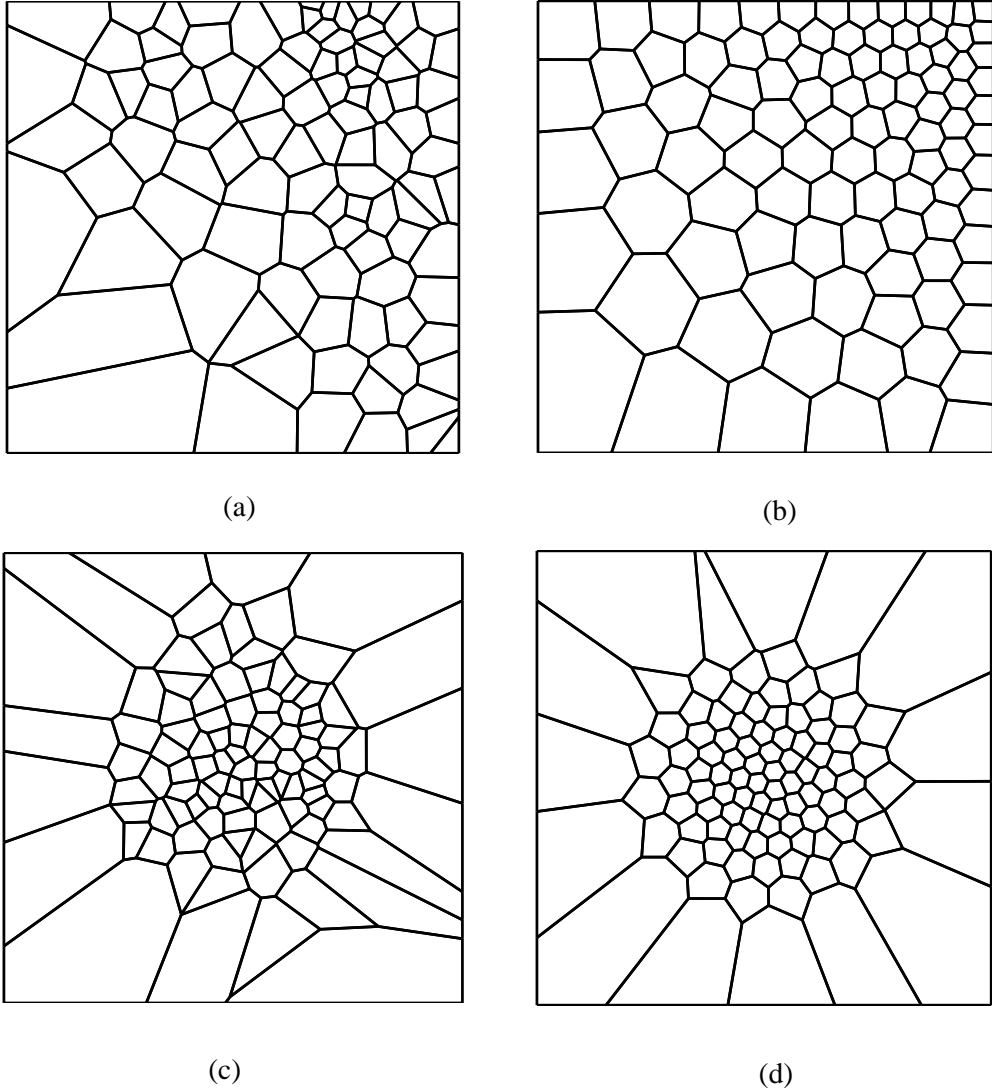


Fig. 8. Patch test on nonuniform polygonal meshes. Left: Initial meshes; right: centroidal meshes. Top: $\rho(\mathbf{x}) = e^{2x_1+2x_2}$; Bottom: $\rho(\mathbf{x}) = e^{-20(x_1-\frac{1}{2})^2-20(x_2-\frac{1}{2})^2}$.

3 Conforming interpolants on quadtree meshes

Quadtree is a hierarchical data structure based on the principle of recursive spatial decomposition of cells into four smaller equal size cells. The tree-based structure of quadtree renders it to be a fast and efficient method for data storage and retrieval. In Fig. 9, a quadtree mesh and its representative tree are presented. After each decomposition hanging nodes are generated if the new elements and their neighbors are at different levels of refinement. Consider edge 1–2 with hanging node a on it. The classical shape function of nodes a , 1 and 2 are quadratic over the edge 1–2 of element A , whereas the classical shape function of these nodes are linear on the edges 1– a and a –2 of elements B and C . Since shape functions of the nodes lying on an edge containing hanging nodes do not match on both sides of the edge, the classical shape functions

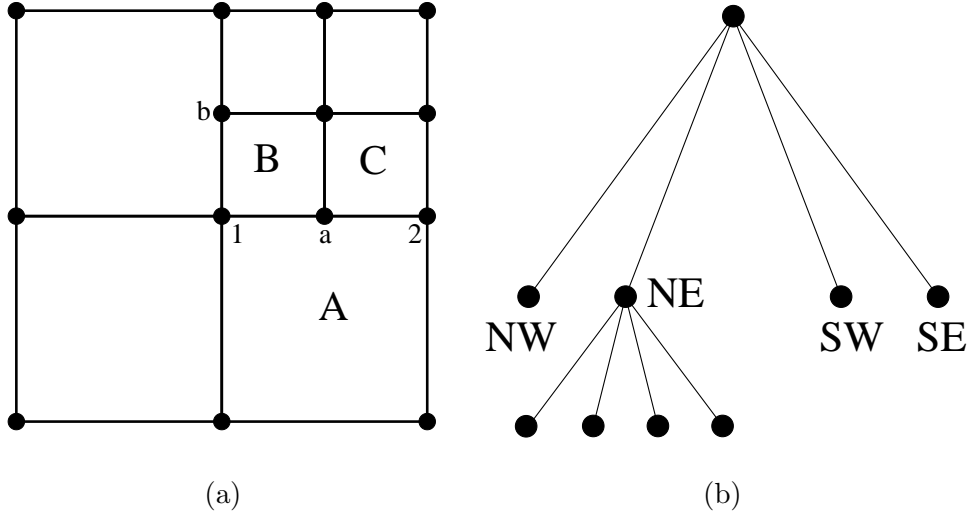


Fig. 9. A quadtree mesh and its representative tree.

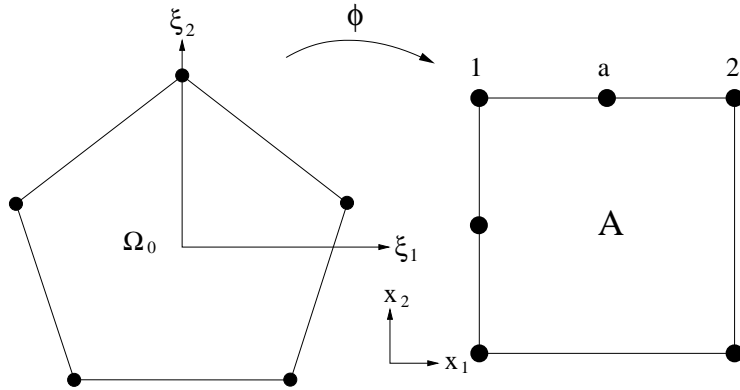


Fig. 10. Mapping from a regular pentagon to a quadtree element with one hanging nodes.

are not conforming over the quadtree meshes upon generation of a hanging node. To obtain conformity along the interelement boundary, the interpolant in element A should be linear along the interface 1-2 to match those of elements B and C . To obtain a linear interpolant on edge 1-2 of element A , this element is considered as a pentagonal element. Using the technique explained in Section 2, first the Laplace interpolant is constructed on the pentagonal reference element and then by using an isoparametric mapping (Fig. 10), the shape functions on element A are obtained. Since Laplace approximant is linear on element edges, the shape functions of nodes a , 1, and 2 are conforming on edge 1-2. This procedure can be performed to construct conforming shape functions for quadtree elements with any number of hanging nodes by using the corresponding polygonal reference element. On using this approach there is no need to restrict the number of hanging nodes to one on each edge (2:1 rule) as is needed in some of the other techniques [2, 3, 6–8, 10].

The mapping illustrated in Fig. 10 from a convex polygon to a square with multi-nodes renders the determinant of the Jacobian to be zero at the hanging nodes. Therefore, the inverse of the Jacobian blows-up at the hanging nodes, which leads to singular shape function derivatives at these points. In Fig. 11, the shape function derivative of the hanging node a in the y -direction is presented. In Fig. 11b, the y -derivative of the shape function of node a on element I is shown, whereas in Fig. 11c the y -derivative along line a - b is displayed. These figures demonstrate that the shape function derivatives are singular at the hanging nodes. We point out that basis functions with singular derivatives at nodes also arises when singular weight functions are used to construct interpolating meshfree basis functions [35]. Although the derivatives in a quadtree element are singular at hanging nodes, numerical tests in Section 3.1 reveal that the patch test is passed to $\mathcal{O}(10^{-13})$ on even refined grids. This indicates that the shape function derivatives are square integrable in the domain. If mean value coordinates [22] are used in quadtree elements with multiple hanging nodes, then piecewise linear interpolation on the boundary is realized and the shape function derivatives are bounded for all points within the element [27, 36]. In the present study, numerical integration on quadtree meshes is performed by subdividing the polygonal reference element into triangles, as is done for polygonal elements (Eq. (3)). This process is illustrated in Fig. 12 for a quadtree element with two hanging nodes.

3.1 Fast assembly of stiffness matrix of quadtree meshes for Poisson equation and elasticity

Consider the quadrilateral elements shown in Fig. 13. Quadrilateral element B is obtained by subdividing the element A into four equal elements. The shape function of node 1 of element A and B in the global coordinate system are:

$$N_1^A(\mathbf{x}) = (1 - x_1)(1 - x_2) \quad N_1^B(\mathbf{x}) = 4\left(\frac{1}{2} - x_1\right)\left(\frac{1}{2} - x_2\right), \quad (6)$$

respectively. On considering the bi-unit reference square, $\boldsymbol{\xi} = [-1, 1]^2$, the global coordinates in element A and B can be expressed in terms of $\boldsymbol{\xi} = (\xi_1, \xi_2)$ as (isoparametric mapping):

$$x_1^A = \frac{\xi_1 + 1}{2} \quad x_2^A = \frac{\xi_2 + 1}{2}, \quad (7a)$$

$$x_1^B = \frac{\xi_1 + 1}{4} \quad x_2^B = \frac{\xi_2 + 1}{4}. \quad (7b)$$

On using Eqs. (6) and (7), shape function derivatives in the local coordinate system are obtained as

$$N_{1,x_1}^B(\boldsymbol{\xi}) = 2N_{1,x_1}^A(\boldsymbol{\xi}) = (\xi_2 - 1), \quad (8a)$$

$$N_{1,x_2}^B(\boldsymbol{\xi}) = 2N_{1,x_2}^A(\boldsymbol{\xi}) = (\xi_1 - 1). \quad (8b)$$

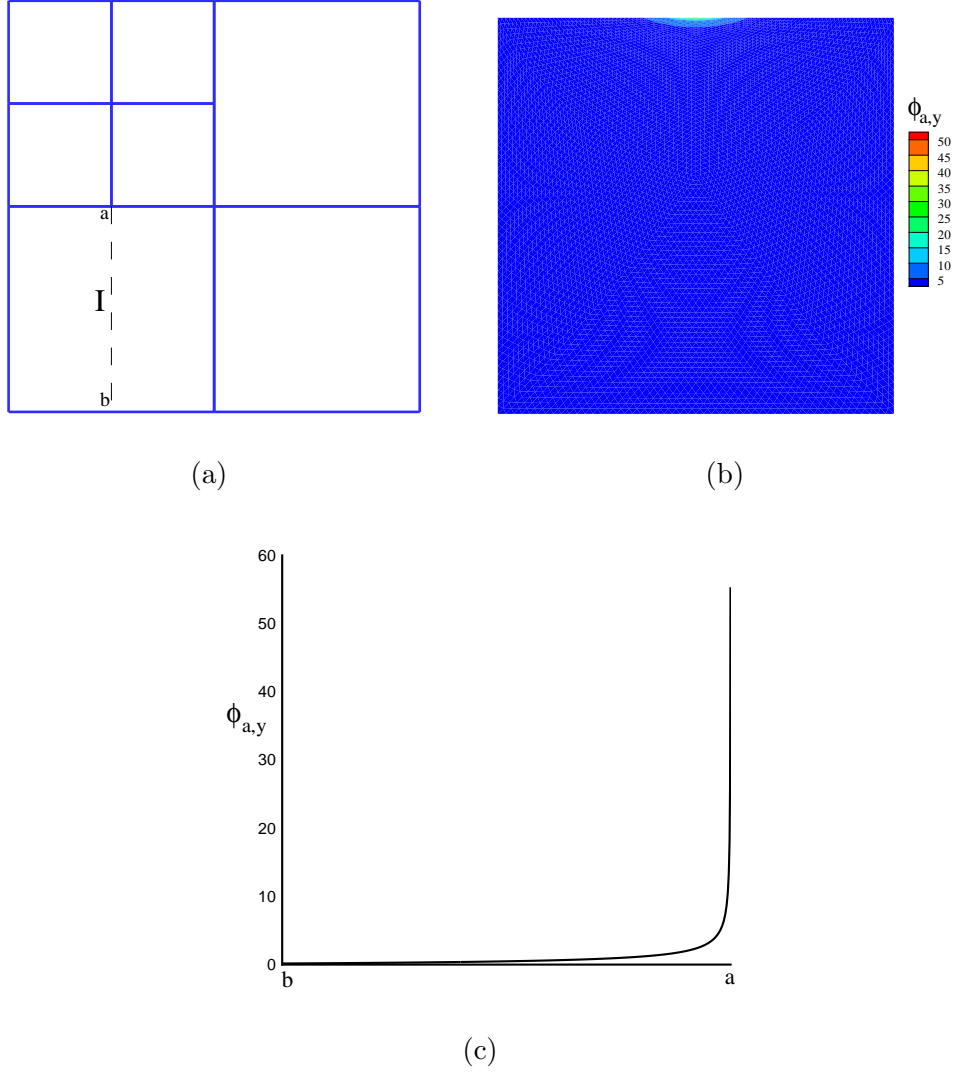


Fig. 11. A quadtree mesh and the shape function derivative of a hanging node. (a) Quadtree mesh; (b) Shape function derivative of node a over element I ; and (c) Shape function derivative of node a along line $a-b$.

It can be shown that Eq. (8) is valid for other corresponding nodes of elements A and B . Therefore, we arrive at the result:

$$\nabla N_a^B = 2\nabla N_a^A \quad (a = 1-4). \quad (9)$$

Now, consider the Poisson equation, $-\nabla^2 u(\mathbf{x}) = f(\mathbf{x})$. The stiffness matrix of this equation is:

$$K_{ab} = \int_{\Omega} \nabla N_a \cdot \nabla N_b dV, \quad (10)$$

where N_a denote the shape functions. On using Eqs. (9) and (10) and by noting that $dV_A = 4.0dV_B$ we obtain $\mathbf{K}^A = \mathbf{K}^B$, which indicates that the stiffness matrix of the subelement is the same as the stiffness matrix of the parent.

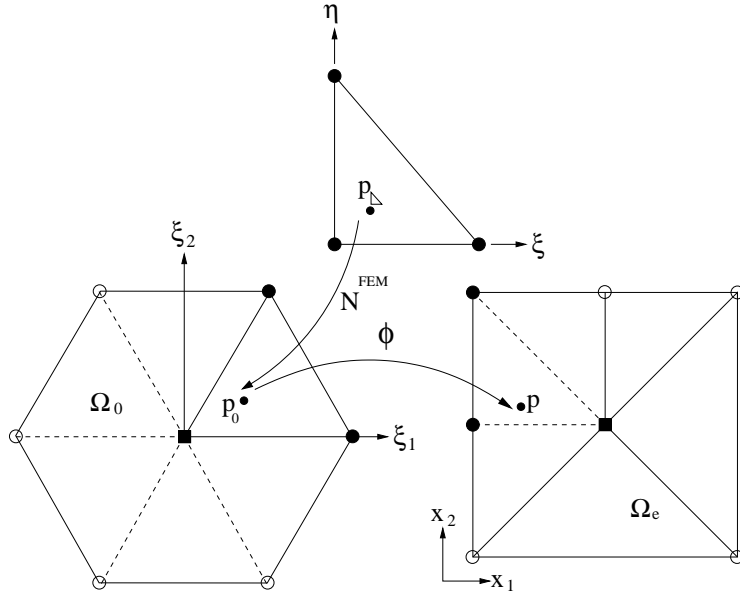


Fig. 12. Numerical integration scheme on a quadtree element with two hanging nodes.

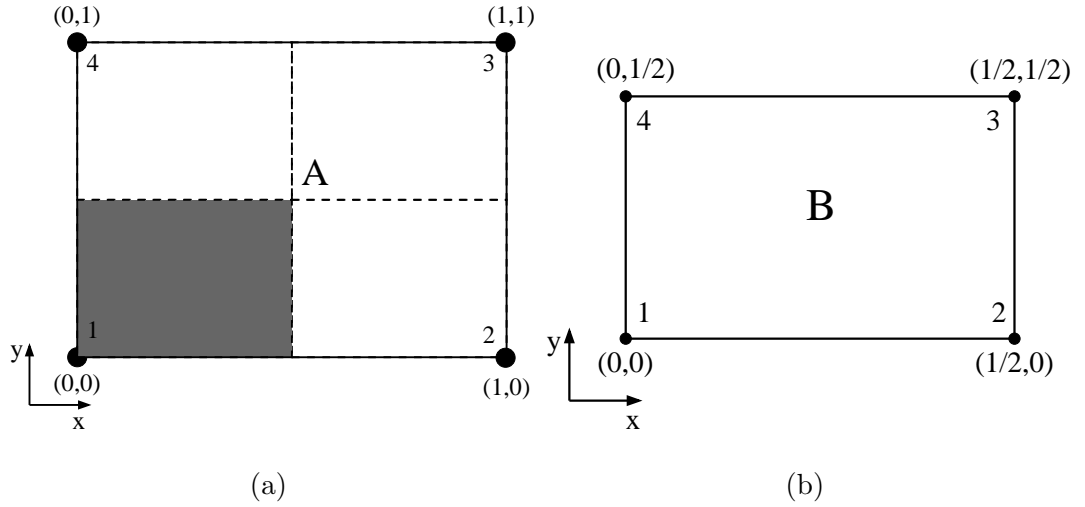


Fig. 13. A quadrilateral element is subdivided into four equal elements. (a) The parent element; and (b) one of its children.

Laplace interpolant is the generalization of classical shape functions from quadrilateral elements to arbitrary convex polygons [11]. This suggests that the stiffness matrix of self-similar quadtree elements (quadtree elements with the same number and with the same position of hanging nodes) are the same. This is numerically verified in this paper and the mathematical proof is similar to the proof presented for the quadrilateral element. This property can be used to speed-up the assembly of the stiffness matrix. Upon using the restricted quadtree mesh (2:1 rule), the number of hanging nodes on each edge can not be more than one, so the number of different types of quadtree

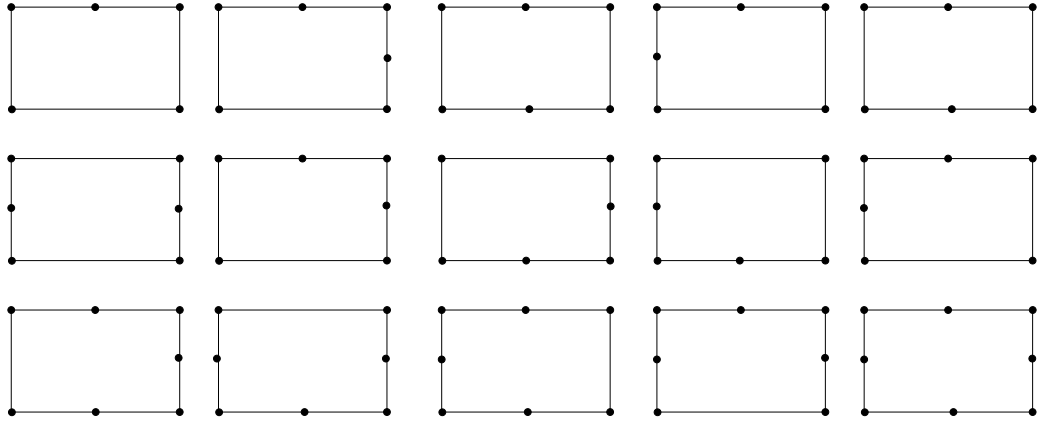


Fig. 14. Different position of hanging nodes in a quadtree element.

elements is restricted to fifteen cases. The fifteen possible type of quadtree elements are shown in Fig. 14. To assemble the global stiffness matrix, first the stiffness matrix \mathbf{K}_0^n of the quadtree elements shown in Fig. 14 is pre-computed and stored to an arbitrary accuracy. To obtain higher accuracy in the stiffness matrix, each subtriangle of Fig. 12 is subdivided into 100 subtriangles with 25 Gauss points in each of the new subtriangles. This procedure is practical, since it is done only once for the fifteen elements shown in Fig. 14. Each of the fifteen stiffness matrices in addition to the stiffness matrix of the four-noded quadrilateral element are pre-computed and stored. The global stiffness matrix can be obtained by using the stored stiffness matrix with no need to repeat the same procedure for all the elements. This procedure is illustrated in Fig. 15 for a mesh consisting of four and five-noded elements. A striking result is that fast and direct assembly is possible, and classical finite element stiffness matrix computations are no longer needed. Moreover, we can define finite-difference stencils on quadtree partitions for Poisson equation and linear elasticity, which can lead to significant gains in speed-up while retaining optimal $\mathcal{O}(h^2)$ convergence in the L^2 norm.

To compare the performance of the fast method with the classical method of assembly, the patch test is conducted on the quadtree meshes shown in Fig. 16. The Laplace equation is solved on the unit square with the exact solution $u(\mathbf{x}) = x_1 + x_2$ imposed on the boundary of the domain. The L^2 and H^1 displacement error norms obtained using the fast and classical methods of assembly are presented in Table 3. The results reveal the accuracy of the fast method is three to five orders better than the accuracy of the classical method.

The time taken to assemble the stiffness matrix using the classical and fast assembly method are presented in Table 4. The fast method takes considerably less time than the classical method. The savings will be more significant for problems requiring mesh adaptivity or remeshing. The improved accuracy in conjunction with the shorter computation time, renders the fast assembly to be a cost-effective technique for stiffness matrix assembly on quadtree meshes.

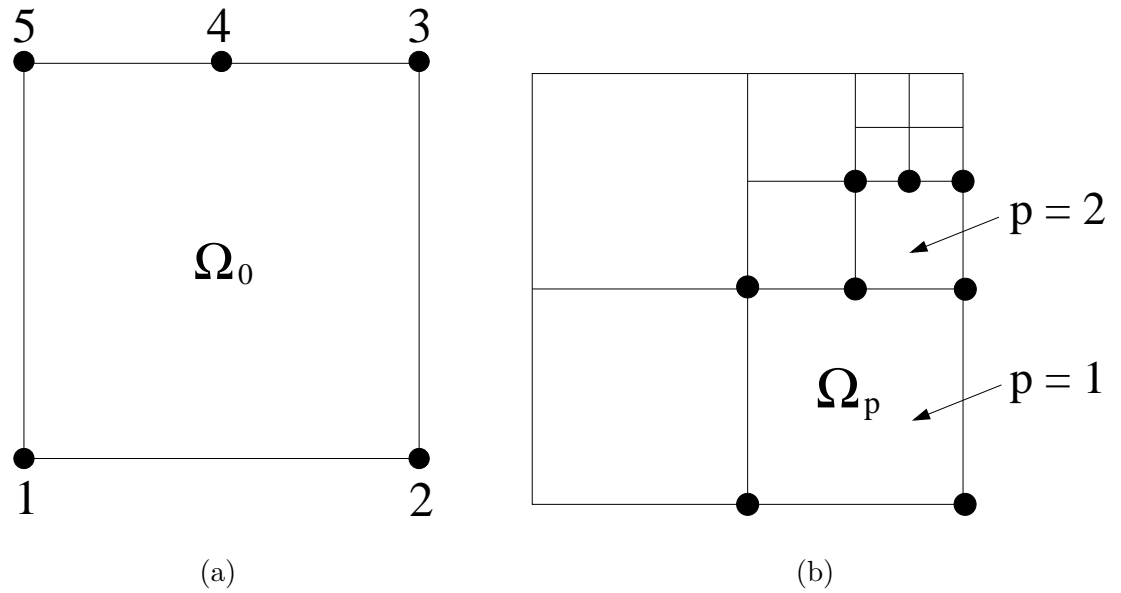


Fig. 15. Fast assembly. (a) Level 0; and (b) Level p . The stiffness matrix \mathbf{K}_0^n for the Laplacian is computed to arbitrary precision in the domain Ω_0 ($n = 5$). The stiffness \mathbf{K} for any self-similar quadtree element with n nodes is equal to \mathbf{K}_0^n .

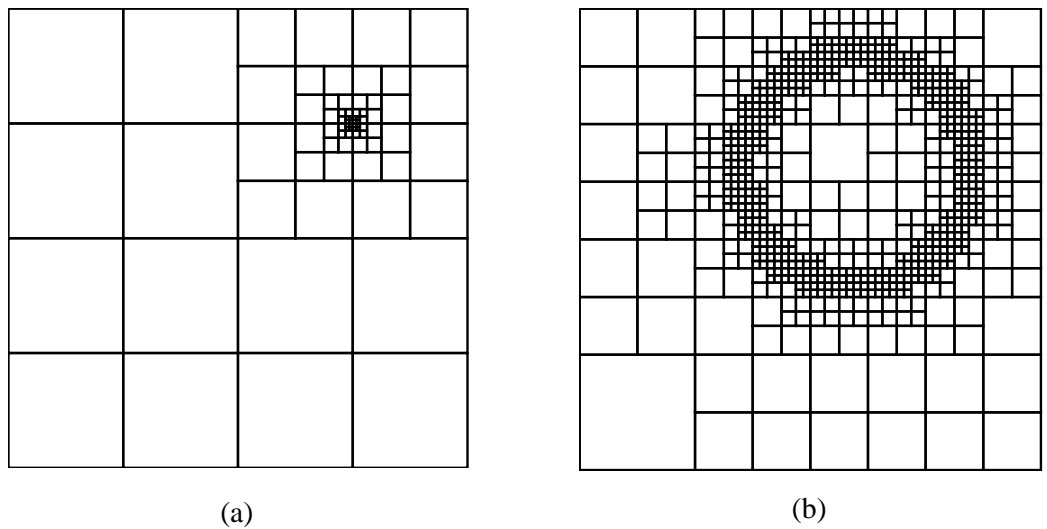


Fig. 16. Patch test on quadtree meshes. (a) Mesh a (105 nodes); and (b) mesh b (876 nodes)

Note that for cases such as nonlinear elasticity or nonlinear Poisson equation with non-constant conductivity tensor, the stiffness matrix is not the same for similar quadtree elements. Therefore, to find the global stiffness matrix, the stiffness matrix of each element must be calculated separately. This can be done by using the classical method described in Section 3.

Table 3
Relative error in the L^2 and H^1 norm for the displacement patch test.

Meshes	Relative error in the L^2 norm		Relative error in the H^1 norm	
	Classical method of assembling stiffness matrix	Fast method of assembling stiffness matrix	Classical method of assembling stiffness matrix	Fast method of assembling stiffness matrix
a	1.62×10^{-9}	1.59×10^{-14}	7.02×10^{-8}	2.43×10^{-13}
b	1.02×10^{-11}	4.99×10^{-14}	1.46×10^{-10}	3.47×10^{-13}

Table 4
Time taken (seconds) to assemble the stiffness matrix of different meshes using classical and fast methods.

Number of nodes	Fast method of assembling	Classical method of assembling
3261	0.84	1.12
7661	2.93	5.51
10021	3.85	9.73

4 Error Estimation

The current state of the art of finite elements employs techniques to assess the reliability of solution and modifications to the mesh during analysis so that the error is equi-distributed over the whole domain. This adaptive strategy generates a sequence of solutions on successively finer meshes. At each stage those elements that contribute most to the error are selected and split into smaller elements. The adaptive process is terminated when the stopping criteria are met. Error estimators are at the heart of mesh adaptive algorithms. Here, two different techniques are used to measure the error. The first one is a residual-based error estimator, which is used to solve Poisson problems. The second method is based on material forces and is used for problems in elastostatics.

4.1 *Explicit Residual-Based Error Estimators*

Explicit residual-based error estimators were first introduced by Babuška and Rheinboldt [37]. This type of error estimators measure the error by evaluating the residual of the system of differential equations as well as the flux jump across the element edges between adjacent elements. Consider the following

elliptic boundary-value problem:

$$-\nabla^2 u(\mathbf{x}) = f(\mathbf{x}) \quad \text{in } \Omega, \quad (11a)$$

$$\frac{\partial u}{\partial n} = g \quad \text{on } \Gamma_N, \quad (11b)$$

$$u = 0 \quad \text{on } \Gamma_D. \quad (11c)$$

In the above equations, Ω is the problem domain, Γ_D and Γ_N are disjoint essential and natural boundary partitions of the domain with $\Gamma_D \cup \Gamma_N = \partial\Omega$, and \mathbf{n} denotes the unit vector normal to the boundary.

In the interest of conciseness, we just state the essential ingredients that pertain to residual-based error estimators; for further details, the interested reader can see Reference [13]. Let τ_h be a regular partition of the domain into finite elements. The energy norm of error of element $k \in \tau_h$ can be approximated by

$$\eta_k^2 = h_k^2 \|r\|_{L^2(\Omega_k)}^2 + h_k \|R\|_{L^2(\partial\Omega_k)}^2. \quad (12)$$

In Eq. (12), r denotes the interior residual, and is defined as

$$r = f(\mathbf{x}) + \nabla^2 u^h(\mathbf{x}) \quad \text{in } \Omega_k. \quad (13)$$

Furthermore, R is the boundary residual representing the jump discontinuity in the normal flux across adjacent elements and is defined as

$$R = \left[\frac{\partial u^h}{\partial n} \right] = \mathbf{n}_k \cdot \nabla u_k^h + \mathbf{n}_{k'} \cdot \nabla u_{k'}^h \quad \text{on } \partial\Omega_k \cap \partial\Omega_{k'}. \quad (14)$$

The global error indicator is the sum over all elements:

$$\eta = \left[\sum_{k \in \tau_h} \eta_k^2 \right]^{\frac{1}{2}}. \quad (15)$$

The quality of error estimator can be measured by the effectivity index defined by

$$\theta = \frac{\eta}{\|e\|}, \quad (16)$$

where $\|e\|$ is the exact energy norm of the error. We expect that by doing mesh refinement the effectivity index approaches unity, but global effectivity indices in the range of 2–3 are acceptable in engineering applications [13]. The adaptive strategy used for mesh refinement consists of the following steps:

- (1) Start with a coarse mesh. Input $\gamma \in (0, 1)$, and the maximum permissible error τ .
- (2) Solve the problem. Evaluate the error of each element η_k and the global error estimate of the domain η .

(3) If $\eta \leq \tau$ stop, else refine all elements such that

$$\eta_k \geq \gamma(\eta_k)_{\max}$$

(4) Go to step 2.

4.2 Material Forces

The concept of energy-momentum tensor was introduced by Eshelby [38]. In Reference [39], Eshelby showed that the forces acting on a defect in an elastic, homogeneous material can be calculated by integrating the energy-momentum tensor over a contour integral around the defect. These forces are known as ‘material’ [15,17] or ‘configurational’ [16] forces. As opposed to physical forces that act over physical space, material forces act on material space and do work in the evolution of defects in material structures; the duality between material and physical forces is presented by Steinmann [40]. In this section, the concept of material forces and a finite element-based technique to evaluate material forces is reviewed. For a detailed discussion on material forces, we point to References [14–17] and to see its application within finite elements the interested reader can refer to References [18, 41, 42].

The energy-momentum tensor is defined as

$$\boldsymbol{\Sigma} = W_0 \mathbf{I} - J \mathbf{F}^T \boldsymbol{\sigma} \mathbf{F}^{-T}, \quad (17)$$

where $W_0(\mathbf{X}, F)$ is the strain energy density per unity volume in the reference configuration, \mathbf{F} denotes the deformation gradient, $J = \det \mathbf{F}$ is the Jacobian, and $\boldsymbol{\sigma}$ is the Cauchy stress tensor. Similar to physical forces, the material force acting over any subdomain $\Omega_0 \subset \mathbf{R}^2$ of the reference configuration must be in equilibrium. The equilibrium equation of material forces can be written as

$$\int_{\partial\Omega_0} \boldsymbol{\Sigma} \cdot \mathbf{N} dS + \int_{\Omega_0} \mathbf{B}^{mat} dV = \mathbf{0}, \quad (18)$$

where \mathbf{N} denotes the normal vector to the boundary and the material body force, \mathbf{B}^{mat} , is given by

$$\mathbf{B}^{mat} = -\frac{\partial W_0}{\partial \mathbf{X}} - \mathbf{F}^T \mathbf{B}^{phy}, \quad (19)$$

where \mathbf{B}^{phy} is the physical body force. The first term in Eq. (18) represents the surface material forces and the second term represents the volume forces acting on Ω_0 . Using Gauss’s theorem and the arbitrariness of the subdomain Ω_0 , the local equilibrium equation of material forces is obtained as

$$\text{Div } \boldsymbol{\Sigma} + \mathbf{B}^{mat} = \mathbf{0}. \quad (20)$$

In the absence of material body forces, i.e., if the material is homogeneous and no physical body force acts on the body, Eq. (20) reduces to

$$\text{Div } \boldsymbol{\Sigma} = \mathbf{0}, \quad (21)$$

which indicates that $\boldsymbol{\Sigma}$ is divergence-free. In other words, when there are no material body forces, the equilibrium of material forces requires the divergence of the energy-momentum tensor to vanish. In this paper, this property of Eshelby's energy-momentum tensor is used to measure the error on quadtree elements.

4.2.1 Finite element discretization

To obtain the weak form of Eq. (21), we multiply it by test functions $\mathbf{v} \in \mathcal{V}_0$ and integrate over the domain to obtain

$$\int_{\Omega_0} \text{Div } \boldsymbol{\Sigma} \cdot \mathbf{v} \, dV = 0, \quad (22)$$

where \mathcal{V}_0 is the space of trial and test functions and is defined as

$$\mathcal{V}_0 = \{\mathbf{v} : \mathbf{v} \in [H^1(\mathbf{X})]^2, \mathbf{v} = \mathbf{0} \text{ on } \Gamma_D\}. \quad (23)$$

Using the product rule, Eq. (22) can be expanded as

$$\int_{\partial\Omega_0} (\boldsymbol{\Sigma} \cdot \mathbf{N}) \cdot \mathbf{v} \, dA - \int_{\Omega_0} \boldsymbol{\Sigma} : \nabla_X \mathbf{v} \, dV = 0. \quad (24)$$

Since \mathbf{v} is zero over Γ_D and $\boldsymbol{\Sigma} \cdot \mathbf{N} = \mathbf{t}^{\text{mat}}$ on Γ_N , Eq. (24) takes the form

$$\int_{\Gamma_N} \mathbf{t}^{\text{mat}} \cdot \mathbf{v} \, dA - \int_{\Omega_0} \boldsymbol{\Sigma} : \nabla_X \mathbf{v} \, dV = 0. \quad (25)$$

The finite element test functions are approximated by

$$\mathbf{v}(\mathbf{X}) = \sum_a \phi_a(\mathbf{X}) \mathbf{v}_a, \quad (26)$$

where $\mathbf{v} = [v_1, v_2, v_3]$, $a = 1, \dots, N$, and N is the number of nodes in an element. From Eq. (26), the gradient of \mathbf{v} can be obtained as

$$\nabla_X \mathbf{v} = \sum_a \mathbf{v}_a \otimes \nabla_X \phi_a. \quad (27)$$

Inserting Eqs. (26) and (27) in Eq. (25) yields

$$\sum_a \mathbf{v}_a \cdot \left[\int_{\Gamma_{N,e}} \phi_a \mathbf{t}^{\text{mat}} \, dS - \int_{\Omega_e} \boldsymbol{\Sigma} : \nabla_X \phi_a \, dV \right] = 0, \quad (28)$$

and on invoking the arbitrariness of \mathbf{v}_a , the term in brackets should be zero, which leads to the following equation for material forces:

$$\mathbf{F}_{e,a}^{mat} = \int_{\Omega_e} \boldsymbol{\Sigma} \cdot \nabla_X \phi_a \, dV, \quad (29)$$

where $\mathbf{F}_{e,a}^{mat}$ is the contribution of element e to the material force of node a . To find the total material force of node a , the material force $F_{e,a}^{mat}$ of all n_{el} elements attached to node a are assembled:

$$\mathbf{F}_a^{mat} = \bigcup_{e=1}^{n_{el}} \mathbf{F}_{e,a}^{mat}. \quad (30)$$

The material force of element e with N nodes is obtained by a direct summation on the material forces of the element nodes:

$$\mathbf{F}_e^{mat} = \bigcup_{a=1}^N \mathbf{F}_{e,a}^{mat}. \quad (31)$$

4.2.2 Adaptive strategy using material forces

From Eqs. (28) and (29), in a homogeneous material with no physical body force, the material force should be zero for all the interior nodes. In deriving Eq. (21), the test functions are assumed to vanish on the boundary of the domain. This is equivalent to imposing essential boundary conditions on the boundary. Owing to fixed boundary conditions, non-zero boundary material forces are generated as reaction forces. Therefore, in a finite element setting, only interior material nodal forces are expected to vanish.

In designing a refinement algorithm, non-vanishing material forces at interior nodes are considered as an indication of insufficient numerical accuracy in the region. To improve the accuracy, greater mesh resolution in such areas is required, which is realized by splitting the quadtree elements into four smaller equal size subelements. The manner in which the mesh refinement process proceeds is as follows:

- (1) First, generate a reasonable mesh using any previous experience available. Set $\gamma \in (0, 1)$, and maximum permissible error τ .
- (2) Obtain the solution on the current mesh and compute the material force of interior nodes using Eq. (30).
- (3) Find the global error indicator by finding the L^2 norm of interior material nodal forces

$$\eta = \left[\sum_{a \in \mathbf{N}_{\text{int}}} (F_a^{mat})^2 \right]^{\frac{1}{2}},$$

where \mathbf{N}_{int} is the set of interior nodes. If $\eta < \tau$ stop, otherwise find the error indicator of each element. The error in an element with n_{int} interior

nodes is:

$$\eta_e = \frac{\bigcup_{a \in \mathbf{N}_{\text{int}}^e} |\mathbf{F}_{e,a}^{\text{mat}}|}{n_{\text{int}}}, \quad (32)$$

where $\mathbf{N}_{\text{int}}^e$ is the set of interior nodes of element e .

(4) Split the elements that satisfy

$$\eta_e \geq \gamma(\eta_e)_{\text{max}}.$$

(5) Go to step 2.

4.2.3 Evaluating the J -integral

The magnitude of the material force evaluated at the crack-tip node should be equal to the J -integral. [40, 41]. The accuracy of the J -integral hinges on the accuracy of the energy-momentum tensor in the elements connected to the crack-tip. Due to the singularity of the stress and strain functions at the crack-tip, the energy-momentum tensor in a finite element analysis is not very accurate in the vicinity of the crack-tip. Therefore, unless special crack-tip elements are used to model the singular field around the crack-tip, the magnitude of the material force at the crack-tip node does not provide a precise approximation of the J -integral. A better approximation of J -integral is achieved by finding the resultant of material nodal forces in a subdomain enclosing the crack-tip [41]. It is shown in Reference [41] that this technique is closely related to the domain formulation of the J -integral.

In this paper, the J -integral in nonlinear elastic materials is computed using either the domain form of the contour J -integral [43] or the improved technique of material forces. For the linear elastic fracture problems, the domain form of interaction integral is adopted to determine the stress intensity factors [43, 44].

5 Model problem for nonlinear elastostatics

In this section, the model problem and governing equations are presented. Since slight distortion of quadtree elements can lead to a negative Jacobian determinant, it is not practical to use a formulation based on the current configuration. In this paper, the total Lagrangian formulation is adopted to conduct computations on quadtree meshes and all the equations will be derived in the initial configuration.

In the absence of body forces, the equilibrium equation of physical forces in

the initial configuration is defined as

$$\text{Div } \mathbf{P} = \mathbf{0} \quad \text{in } \Omega_0 \quad (33a)$$

$$\mathbf{u} = \mathbf{0} \quad \text{on } \Gamma_D, \quad (33b)$$

$$\mathbf{P} \cdot \mathbf{N} = \mathbf{t}_0 \quad \text{on } \Gamma_N, \quad (33c)$$

where $\Omega_0 \subset \mathbf{R}^2$ is the problem domain in the initial configuration, Γ_D and Γ_N are the essential and natural boundary partitions such that $\Gamma_D \cup \Gamma_N = \partial\Omega_0$ and $\Gamma_D^0 \cap \Gamma_N^0 = \emptyset$, \mathbf{N} denotes the unit outward normal, \mathbf{P} represents the first Piola-Kirchhoff stress tensor, and \mathbf{u} is the displacement vector. The weak form of these equations in the initial configuration is expressed as [45]

$$\mathcal{F}(\mathbf{u}, \delta\mathbf{u}) = \int_{\Omega_0} \mathbf{S} : \delta\mathbf{E} dV - \int_{\Gamma_N} \mathbf{t}_0 \cdot \delta\mathbf{u} = 0 \quad \forall \delta\mathbf{u} \in \mathcal{U}_0, \quad (34)$$

where \mathbf{S} represents the second Piola-Kirchhoff stress tensor, \mathbf{E} denotes the Green-Lagrange strain tensor, and $\delta\mathbf{u}$ is the virtual displacement field. Since Eq. (34) is nonlinear with respect to geometry and material, a Newton-Raphson procedure is implemented to find the solution. For the Newton-Raphson implementation, Eq. (34) is linearized. On assuming the surface forces are independent of the configuration, the linearized form of Eq. (34) can be written as [45]

$$D\mathcal{F}(\mathbf{u}, \delta\mathbf{u})[\mathbf{u}] = \int_{\Omega_0} \delta\mathbf{E} : \mathcal{C} : D\mathbf{E}[\mathbf{u}] dV + \int_{\Omega_0} \mathbf{S} : [(\nabla_{\mathbf{x}}\mathbf{u})^T \nabla_{\mathbf{x}}\delta\mathbf{u}] dV, \quad (35)$$

where $D\bullet$ denotes the directional derivative of \bullet , and \mathcal{C} is the material elasticity tensor that can be represented in component form as

$$C_{ijkl} = \lambda C_{ij}^{-1} C_{kl}^{-1} + \mu (C_{ij}^{-1} C_{jl}^{-1} + C_{il}^{-1} C_{kl}^{-1}), \quad (36)$$

where λ and μ are the Lamé constants. The material chosen is an isotropic compressible Neo-Hookean type with stored energy function defined as

$$W(\mathbf{C}) = \frac{1}{2}\lambda(\ln J)^2 - \mu \ln J + \frac{1}{2}\mu(I_1 - 3), \quad (37)$$

where $\mathbf{C} = \mathbf{F}^T \cdot \mathbf{F}$ is the right Cauchy-Green tensor, and $I_1 = \text{trace } \mathbf{C}$. On using Eq. (37), the stress tensors can be expressed as [46]

$$\begin{aligned} \mathbf{S} &= 2 \frac{\partial W}{\partial \mathbf{C}} = 2 \left(\frac{\partial W}{\partial I_1} + I_1 \frac{\partial W}{\partial I_2} \right) \mathbf{I} - 2 \frac{\partial W}{\partial I_2} \mathbf{C} + 2I_3 \frac{\partial W}{\partial I_3} \mathbf{C}^{-1} \\ &= \lambda \ln J \mathbf{C}^{-1} + \mu (\mathbf{I} - \mathbf{C}^{-1}), \end{aligned} \quad (38)$$

$$\begin{aligned} \boldsymbol{\sigma} &= \frac{1}{J} \mathbf{F} \cdot \mathbf{S} \cdot \mathbf{F}^T = 2 \left(\frac{\partial W}{\partial I_1} + I_1 \frac{\partial W}{\partial I_2} \right) \mathbf{B} - 2 \frac{\partial W}{\partial I_2} \mathbf{B}^2 + 2I_3 \frac{\partial W}{\partial I_3} \mathbf{I} \\ &= \frac{1}{J} (\lambda \ln J \mathbf{I} + \mu(\mathbf{B} - \mathbf{I})) \end{aligned} \quad (39)$$

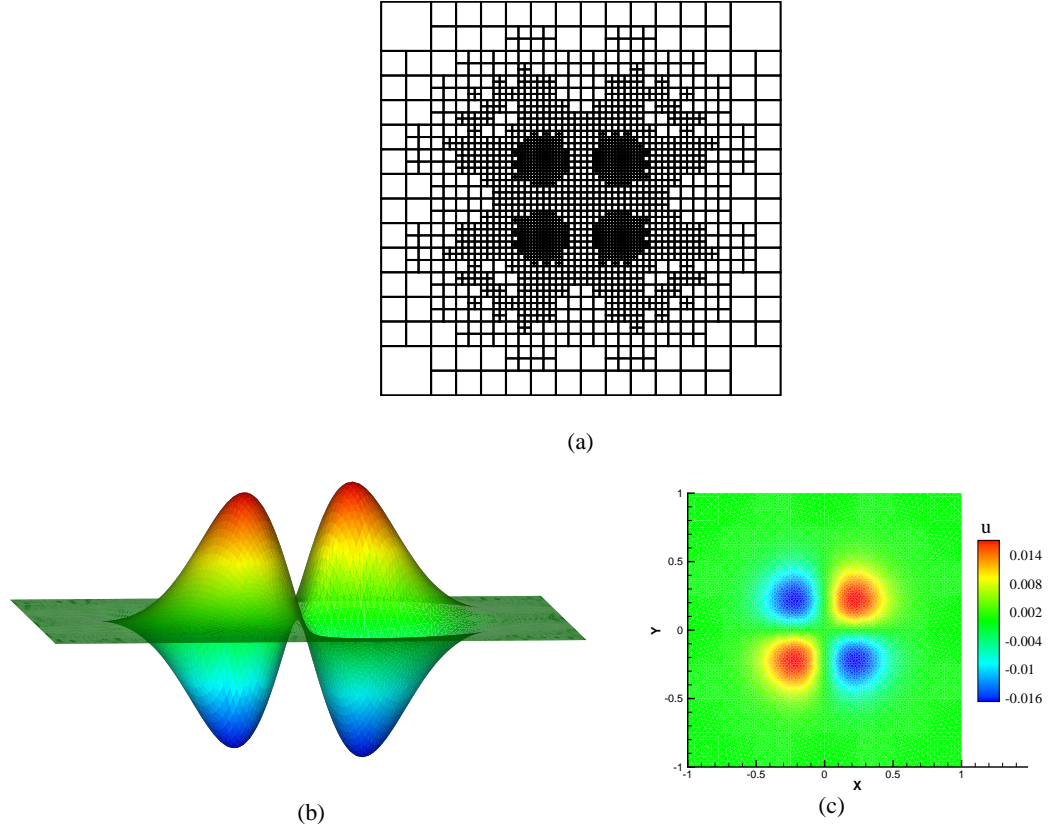


Fig. 17. Surface approximation. (a) Quadtree mesh (step 6); and (b) $u^h(\mathbf{x}) = \sum_i \phi_i(\mathbf{x})u_i$, where $u_i = u(\mathbf{x}_i)$.

where (I_1, I_2, I_3) are the principal invariants of \mathbf{C} , and $\mathbf{B} = \mathbf{F} \cdot \mathbf{F}^T$ denotes the left Cauchy-Green deformation tensor.

6 Numerical examples

As the first example, we use the proposed quadtree interpolant in a surface approximation problem. Quadtree meshes are used to approximate the function $u(x, y) = e^{-10(x^2+y^2)} \sin x \sin y$ in the domain $\Omega = (-1, 1)^2$. As an adaptive strategy, we use the L^2 -norm of the error: $\eta^2 = \int_{\Omega_e} (u - u^h)^2 d\Omega / \int_{\Omega} u^2 d\Omega$. If $\eta > 0.01$, an element is split into four elements (children); otherwise it is not. We begin with a regular 8×8 grid, and the solution after six steps (2653 nodes) reproduces the function well (Fig. 17). The maximum point-wise error and the normalized $L^2(\Omega)$ -error in the domain are both $\mathcal{O}(10^{-3})$ – $\mathcal{O}(10^{-2})$.

The rest of this section is divided into two parts. In the first part, quadtree meshes with residual-based error estimators are used to solve Poisson problems. In the second part, quadtree meshes are used to solve linear and non-linear elastostatics problems. For the purpose of error estimation in elasticity problems, material forces are adopted as an error indicator.

6.1 Poisson problems

Poisson problems with singularity and sharp gradients are solved in this section. To estimate the error, explicit residual-based error estimators are employed. This error estimator is based on evaluating the boundary jumps and the interior residuals. As indicated in Section 2, the first derivatives of Laplace interpolants are square integrable. However, since the singularity of second-order derivatives is stronger, it is not possible to evaluate the interior residuals of Eq. (6.1). Since Laplace interpolant is linearly complete, we appeal to References [47–49] and ignore the interior residuals and just use the boundary jumps to estimate the error.

6.1.1 Example 1

We solve the Poisson equation in a unit square with Dirichlet boundary conditions:

$$-\nabla^2 u = f \quad \text{in } \Omega = (0, 1)^2, \quad (40a)$$

$$u = 0 \quad \text{on } \partial\Omega. \quad (40b)$$

The source term f is chosen such that the exact solution of the problem is [8]:

$$u(\mathbf{x}) = x_1^5 x_2^5 (1 - x_1)(1 - x_2). \quad (41)$$

This problem is solved in Reference [9] using the exact solution to estimate the error. In this paper, the problem is solved using the residual-based error estimator. Although the solution is almost flat over most part of the domain, the gradient of u is large at the top right region, so refinement is expected over this region. The initial mesh consists of a 2×2 quadrilateral mesh. The successive mesh refinements are shown in Fig. 18, and the high gradient region is discretized by smaller elements. The estimated error and the exact error are shown in Fig. 18g. By increasing the number of nodes, the effectivity index converges to 1.15.

6.1.2 Example 2

The Laplacian equation is solved in the domain shown in Fig. 19a. The Dirichlet boundary consists of edge OA , and g is defined such that

$$u = 0.0700754 \sqrt{r} \sin\left(\frac{\theta}{2}\right),$$

where r is the distance to point O and θ is the angle of r with edge OA . The derivatives of u are singular at the origin so elements adjacent to point O are expected to be refined. The initial mesh and the refined meshes are shown in Figs. 19b–19e. Application of the explicit method leads to mesh refinement

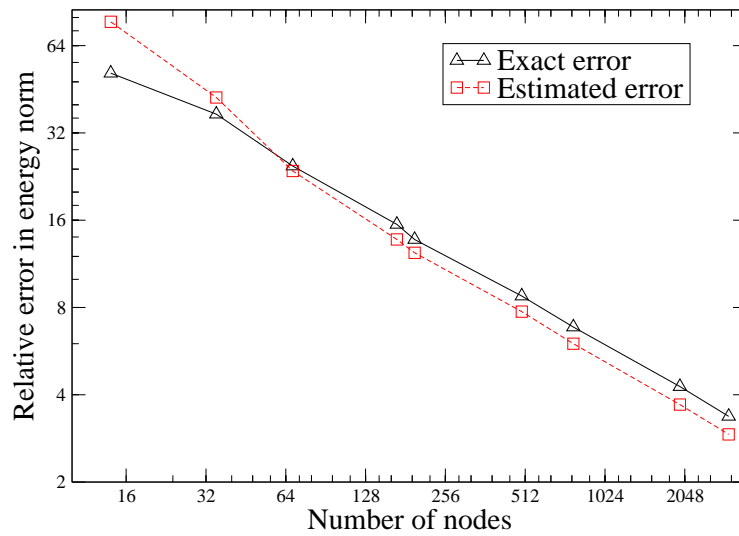
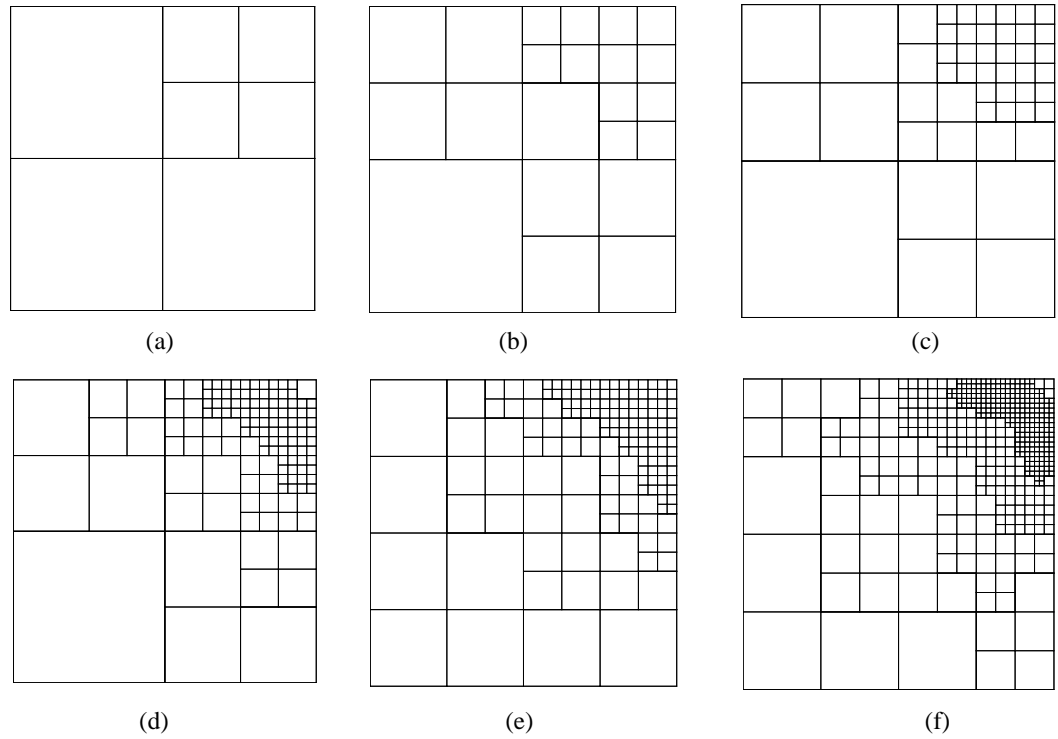


Fig. 18. Successive adaptive refinement over the region with high gradient (Example 1). (a) Level 1 (14 nodes); (b) Level 2 (35 nodes); (c) Level 3 (68 nodes); (d) Level 4 (168 nodes); (e) Level 5 (196 nodes); (f) Level 7 (497 nodes); and (g) Estimated error and exact error.

in the vicinity of O . The logarithmic plot of exact and estimated error versus number of nodes is shown in Fig. 19f. It can be seen that the estimated error converges to the exact error.

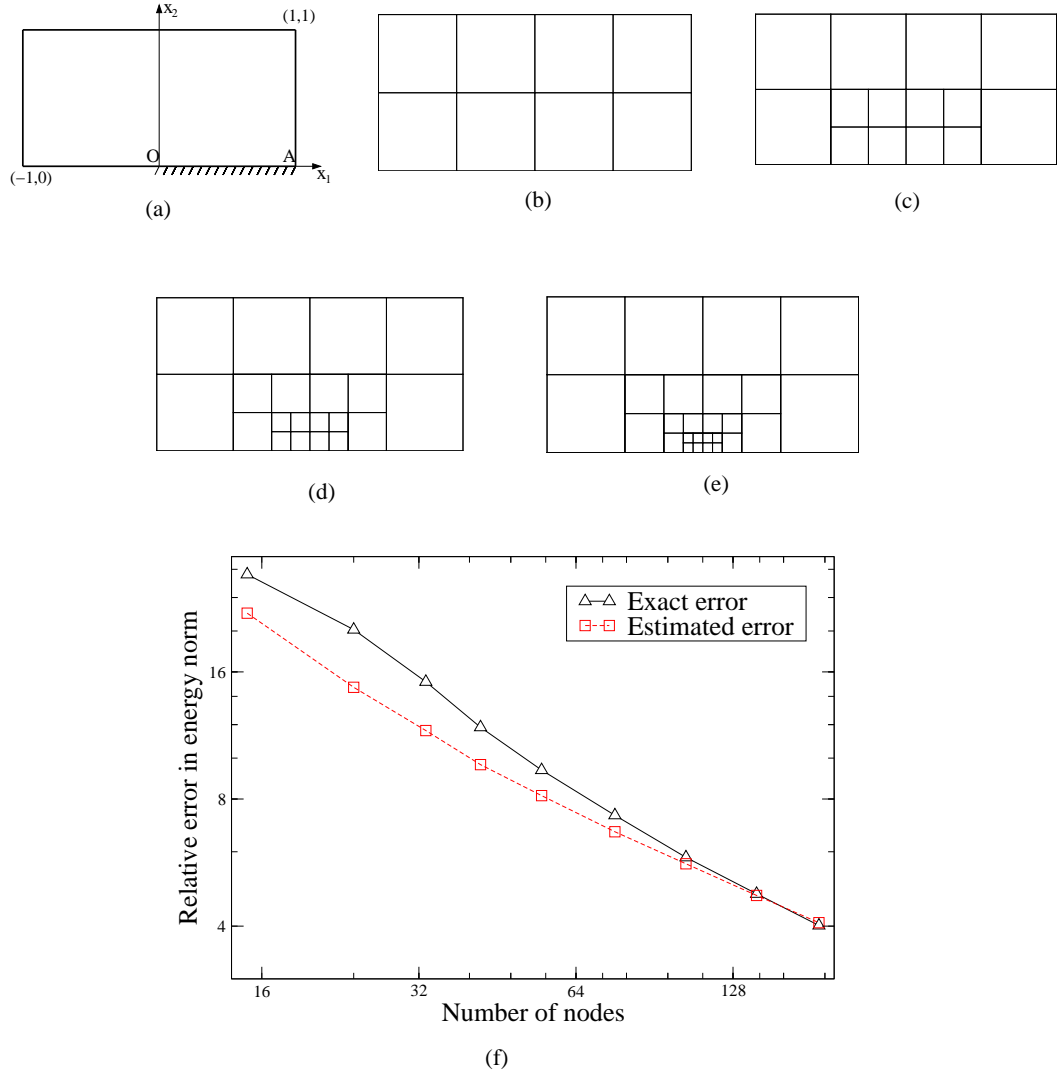


Fig. 19. Example 2, (a) Model and boundary condition; (b)-(e) Successive adaptive refinement near the corner singularity; and (f) Estimated error and exact error.

6.1.3 Example 3

Two Poisson problems with sharp gradients are solved. The localized (model pseudopotential) solution $u(\mathbf{x})$, which appears in electronic structure calculations is given by [50]

$$u(\mathbf{x} - \mathbf{x}_0) = u_0 e^{-\frac{\|\mathbf{x} - \mathbf{x}_0\|^2}{r_c^2}}, \quad (42)$$

with $u_0 = -16$, $r_c = 0.5$. On choosing $\mathbf{x}_0 = (0, 0)$, steep gradients are generated at the origin. The refined meshes and the numerical solution are shown in Fig. 20. As is observed, the numerical solution is able to capture the steep gradients and smaller elements are generated in the vicinity of high gradients. The exact energy norm error and the calculated energy norm error are shown in Fig. 21, and the effectivity index converges to 1.18. The exact solution of

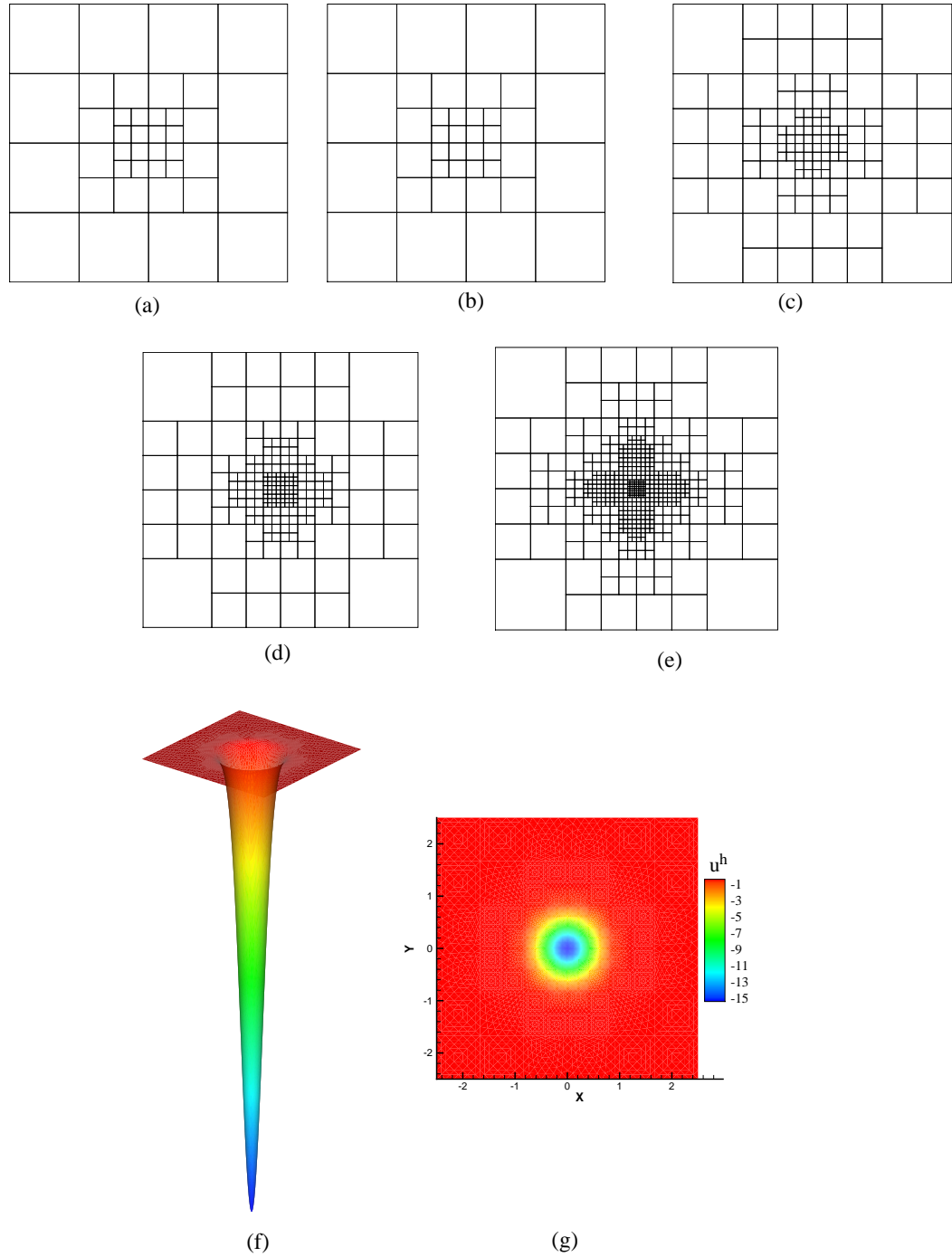


Fig. 20. Adaptive refinement and numerical solution (Example 3a). (a) Level 1 (25 nodes); (b) Level 2 (41 nodes); (c) Level 3 (157 nodes); (d) Level 4 (253 nodes); (e) Level 5 (585 nodes); (g) and (h) the numerical solution.

the second problem is taken to be the sum of two potentials:

$$u(\mathbf{x}) = u(\mathbf{x} - \mathbf{x}_1) + u(\mathbf{x} - \mathbf{x}_2), \quad (43)$$

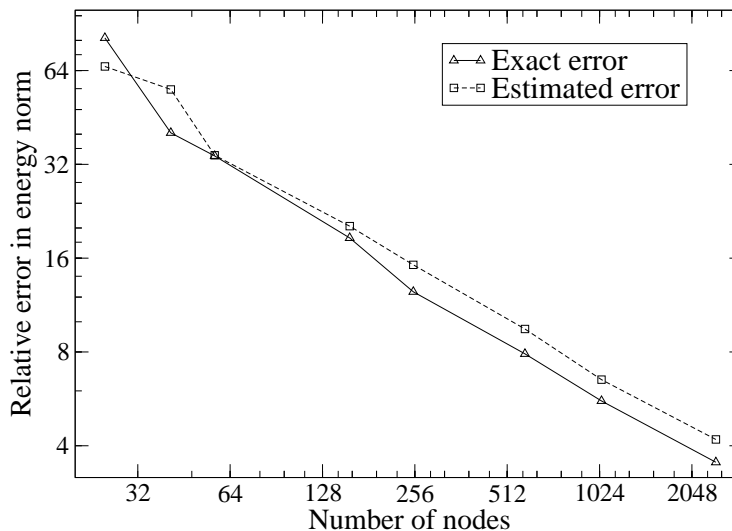


Fig. 21. Estimated error and exact error of example 3a.

with $\mathbf{x}_1 = (-1, 0)$ and $\mathbf{x}_2 = (1, 0)$. The refined meshes and the numerical solution are shown in Fig. 22. The plot in Fig. 23 reveals that the effectivity index converges to 1.17.

6.1.4 Example 4: L-shaped domain

As the final application of residual-based error estimators, the Poisson equation is solved in an L-shaped domain. The exact solution is chosen to be $u(r, \theta) = r^{2/3} \sin\left(\frac{2\theta}{3}\right)$, where r is the distance to the origin and θ is measured from the positive direction of x -axis (Fig. 24) [51]. The initial and refined meshes are shown in Fig. 25. Due to the singularity of the derivatives of u at the origin, the meshes are refined in the vicinity of the re-entrant corner. The estimated and exact energy norm of error are shown in Fig. 25g. It can be seen that the exact and estimated error are proximal.

6.2 Elasticity problems

In this section, quadtree meshes are employed to solve problems in linear and nonlinear elasticity. Material forces are used as an error indicator in the mesh refinement of quadtree meshes. In the numerical computations, we assume $\lambda = 1000$ and $\mu = 400$.

6.2.1 Stress intensity factor computations for an edge-crack

The adaptive strategy is employed to solve an edge-crack in a semi-infinite domain. The problem is solved for three cases: specimen subjected to pure

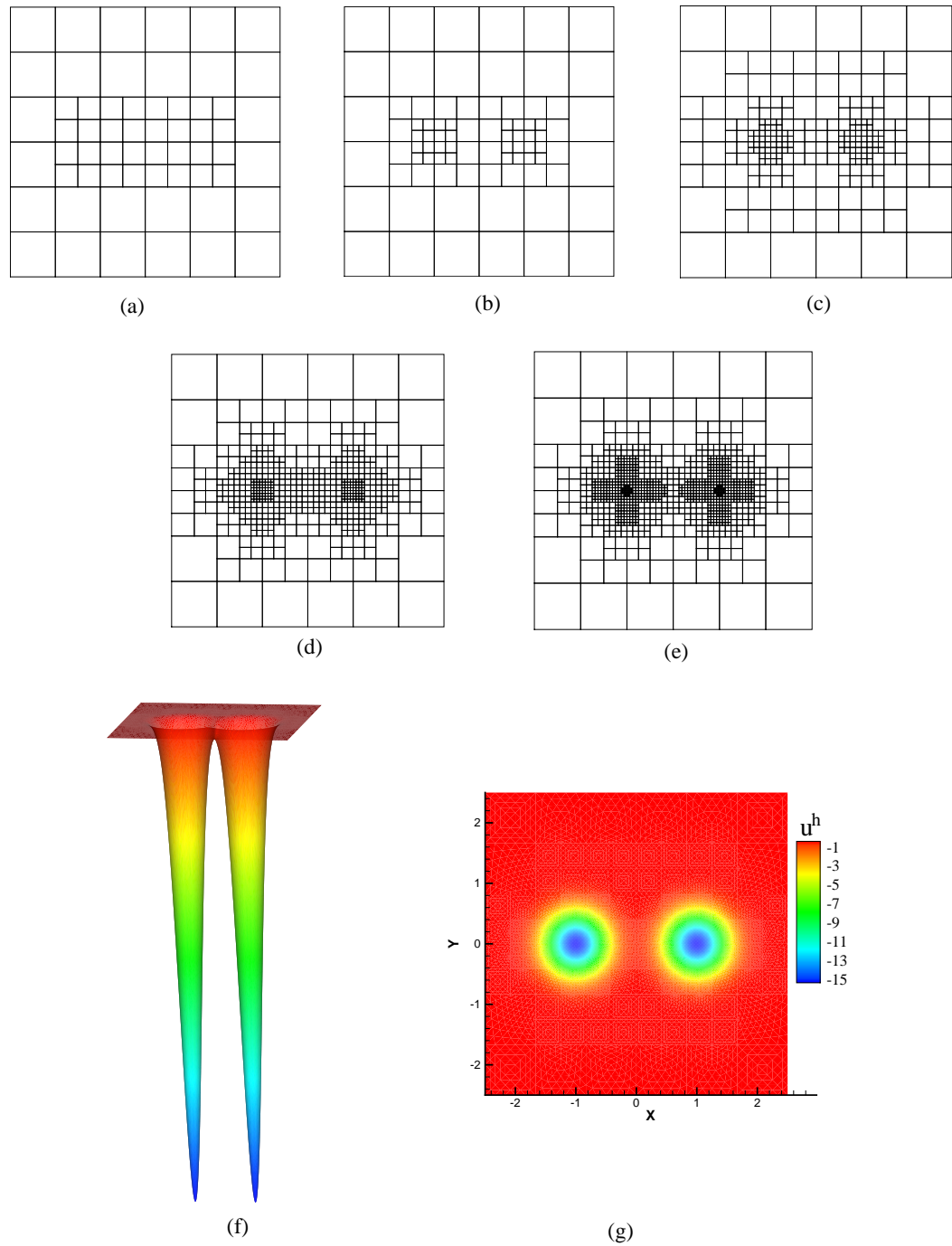


Fig. 22. Successive adaptive refinement over the region with high gradient (Example 3b). (a) Level 1 (79 nodes); (b) Level 2 (111 nodes); (c) Level 3 (293 nodes); (d) Level 4 (661 nodes); (e) Level 5 (1239 nodes); (f) Level 6 (2539 nodes); and (g) and (h) the numerical solution.

mode I, pure mode II, and combination of mode I and mode II deformation states. A region close to the crack-tip is studied. The exact displacement field is applied over the boundary with: (a) $K_I = 1.0, K_{II} = 0.0$ (pure mode I); (b) $K_I = 0.0, K_{II} = 1.0$ (pure mode II); and (c) $K_I = 1.0, K_{II} = 1.0$ (mixed

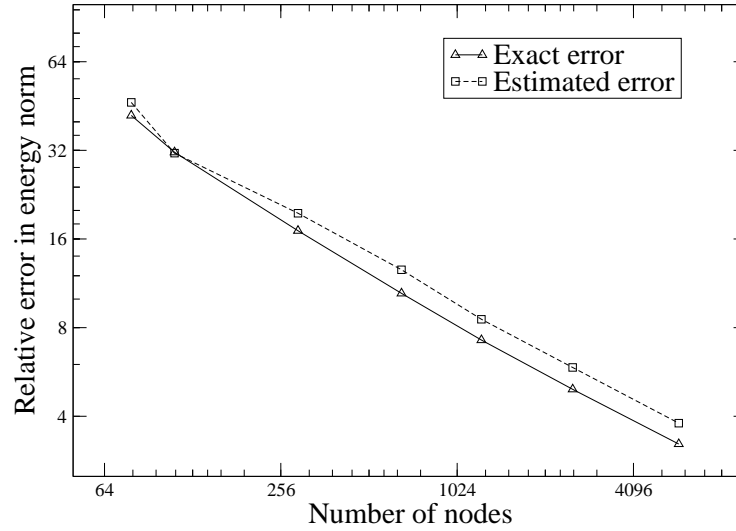


Fig. 23. Estimated error and exact error of Example 3b.

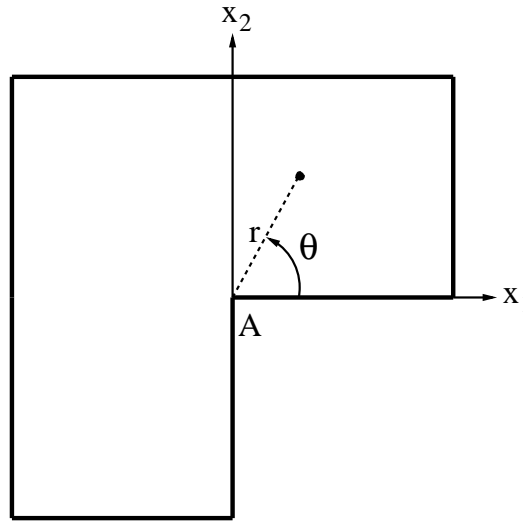


Fig. 24. L-shaped domain.

mode). The material forces on the final mesh for the mixed-mode problem are shown in Fig. 26. To evaluate the stress intensity factors, the domain form of interaction integral is employed [43,44]. The computed stress intensity factors are listed in Table 5.

6.2.2 Linear elastic edge-crack specimen

As the second linear elastic problem, a finite-dimensional plate in plane stress condition under the action of uniform tension on the top edge is analyzed. The sketch of the problem is shown in Fig. 27a, and the refined meshes are presented in Figs. 27b–27e. For the case shown in Fig. 27a, the exact normalized stress intensity factor is $\frac{K_I^{\text{ref}}}{\sigma\sqrt{\pi a}} = 2.8264$ [52]. The domain form of interaction

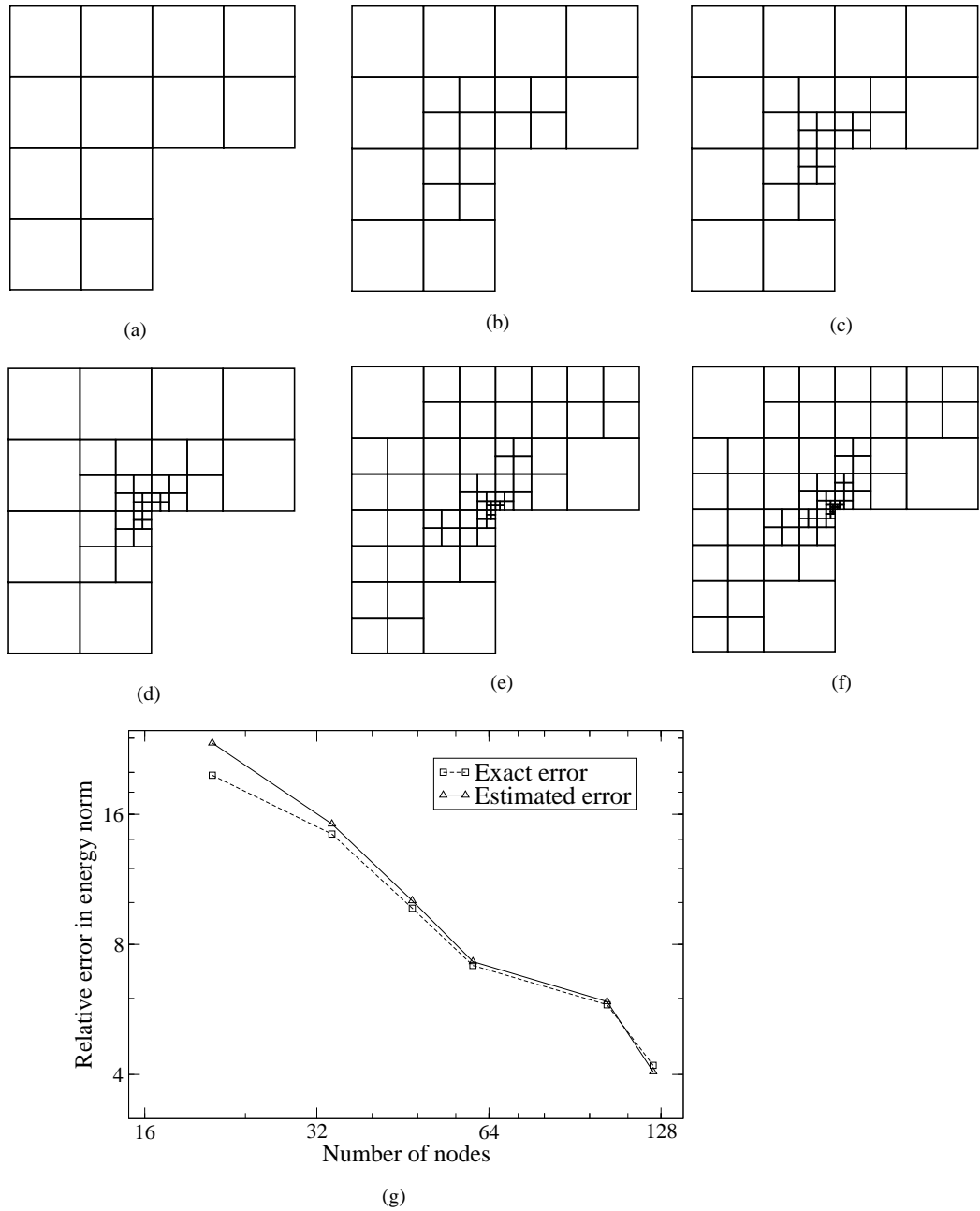


Fig. 25. Successive adaptive refinement on the L-shape domain. (a) Level 1 (21 nodes); (b) Level 2 (34 nodes); (c) Level 3 (47 nodes); (d) Level 4 (60 nodes); (e) Level 5 (103 nodes); (f) Level 6 (124 nodes); (g) Estimated error and exact error of L-shape domain.

integral is used to calculate the stress intensity factors. The calculated stress intensity factors are presented in Table 6.

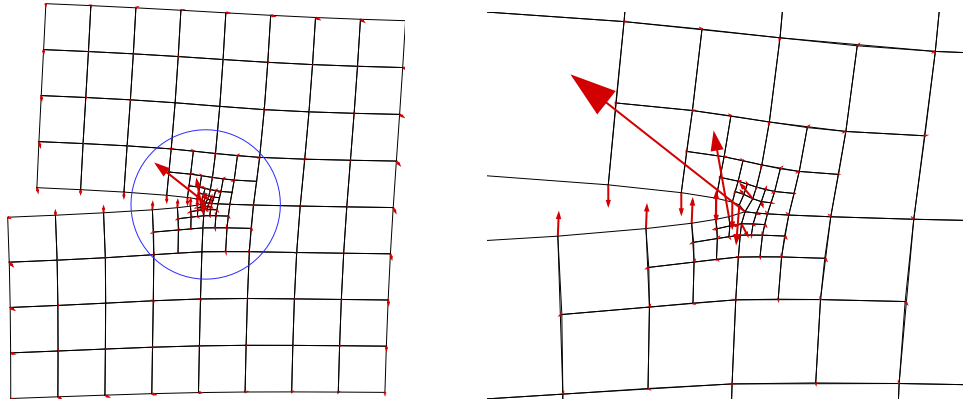


Fig. 26. Mesh refinement and material forces for the mixed-mode problem. The right figure is a zoom of the circled region depicted on the left.

Table 5

Normalized SIF: Semi-infinite edge crack problem.

Number of nodes	Pure Mode I	Pure Mode II	Combination of mode I and II	
	K_I	K_{II}	K_I	K_{II}
102	1.041	1.050	1.050	1.050
119	1.022	1.031	1.030	1.028
136	1.012	1.021	1.023	1.017
153	1.008	1.011	1.019	1.013

Table 6

Normalized SIF: Edge crack under tension.

Mesh	Number of nodes	$\frac{K_I^{\text{ref}}}{\sigma\sqrt{\pi a}}$	% Error in interaction integral	L^2 norm of interior material nodal force
b	104	2.7189	3.80	0.552
c	121	2.7600	2.34	0.396
d	138	2.7786	1.69	0.226
e	155	2.7845	1.48	0.052

6.2.3 Edge-cracked beam

An edge cracked beam is solved as the next example. The schematic of this problem is shown in Fig. 28 with the following parameters: $H = 2$ inches, $L = 8$ inches and $\frac{a}{H} = 0.5$. A total displacement of half inch is applied to the middle point of the top edge. The refined meshes are shown in Fig. 28. Large material

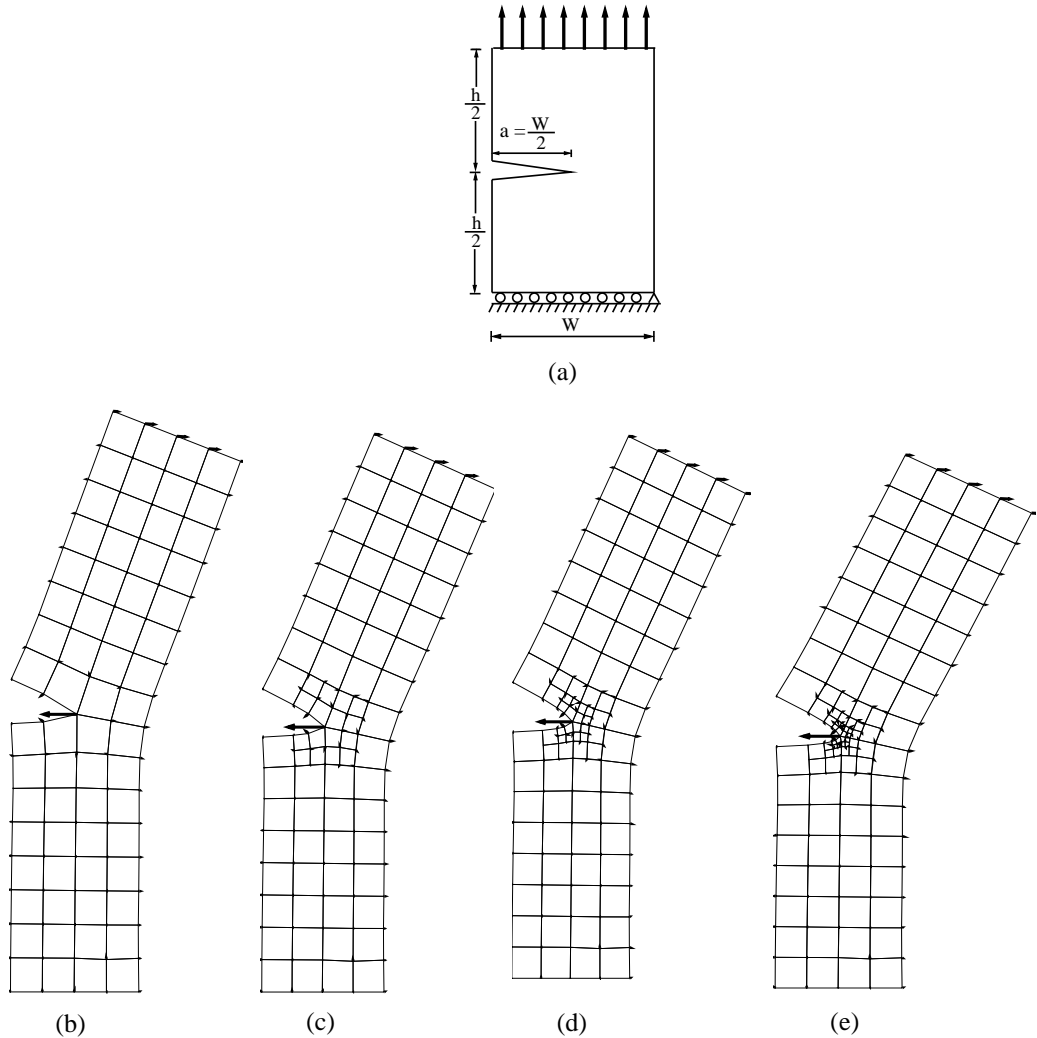


Fig. 27. The linear plane stress edge-crack specimen. (a) Geometry and boundary conditions and (b) to (e) refined meshes.

forces are observed at the crack-tip and at the points where displacement boundary conditions are prescribed. The J -integral (domain form of contour integral) and the improved material forces as a function of mesh refinement are shown in Fig. 28f, whereas in Fig. 28g, the L^2 norm of the interior material nodal forces is plotted. The two methods are found to be in good agreement.

6.2.4 Compact-tension specimen

In this example, we consider a nonlinear plane strain compact-tension specimen. The geometry and boundary conditions are shown in Fig. 29a. Due to the singularity at the crack-tip, we expect the mesh to be refined around the crack-tip. The refined meshes and the calculated material forces are shown in Figs. 29b–29f. These plots indicate that the material forces vanish at all the interior nodes except at the crack tip. Since the supports act as inho-

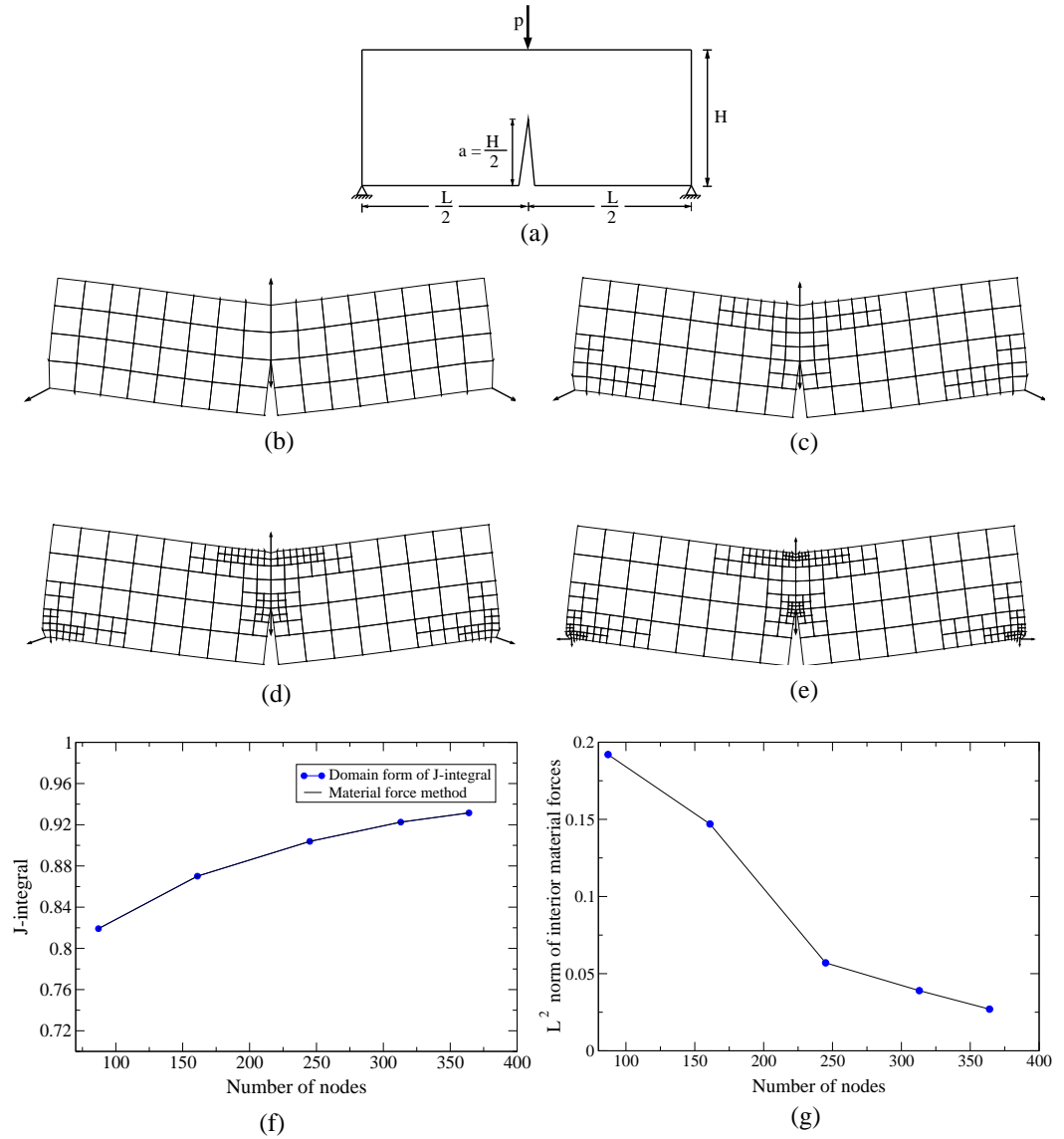


Fig. 28. The nonlinear plane strain edge-cracked beam. (a) Geometry and boundary conditions; (b) to (e) refined meshes; (f) J -integral and (g) L^2 norm of interior material nodal forces.

mogeneities, material forces are generated where the loads are applied (see Figs. 29b–29f). The J -integral and the L^2 norm of the interior material nodal forces are plotted for different meshes in Figs. 29g and 29h, respectively.

6.2.5 Nonlinear edge-crack specimen

A plane strain edge-crack specimen under tensile loading is considered. The geometry and boundary conditions of the problem are shown in Fig. 30a. The height to width ratio is $H/W = 4$ and crack length to width ratio is $a/W = 0.5$. A uniform displacement of 50% is applied to the top edge of the

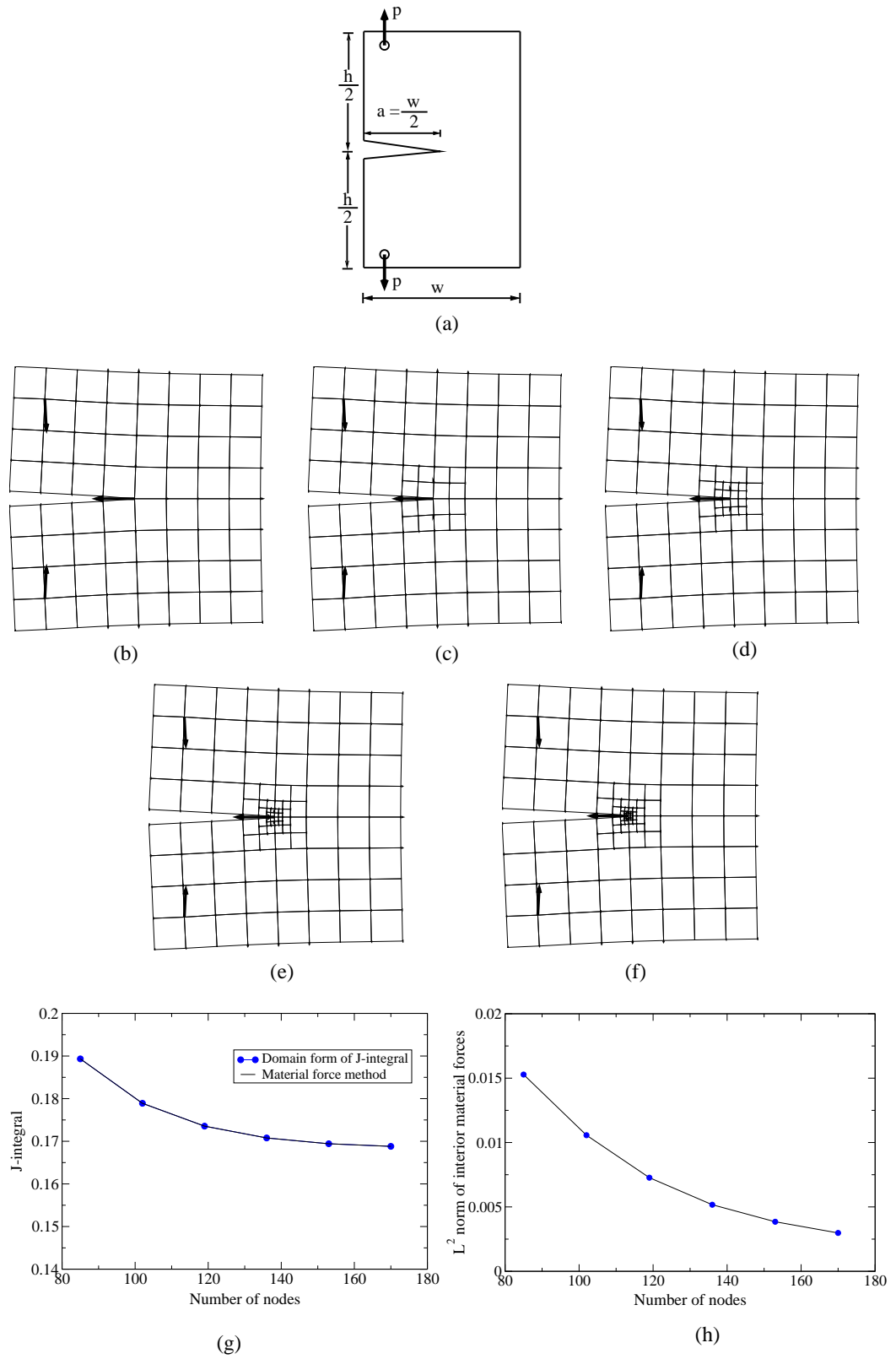


Fig. 29. The nonlinear elastic compact-tension specimen. (a) Geometry and boundary conditions; (b) to (f) refined meshes; (g) J -integral and (h) L^2 norm of interior material nodal forces.

specimen using ten load increments. The refined meshes and the calculated material forces are shown in Fig. 30. The refined meshes have greater resolution in the vicinity of the crack-tip. The J -integral and the L^2 norm of the material forces of the interior nodes are plotted for different meshes in Figs. 30f and 30g, respectively.

7 Conclusion

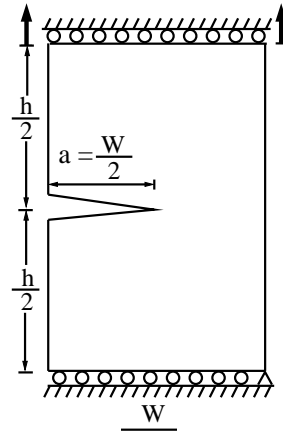
In this paper, an h -adaptive technique using quadtree meshes was presented. The hierarchical property of quadtree meshes renders it to be a desirable mesh data structure for h -adaptivity. On using quadtree meshes for h -adaptivity certain difficulties related to mesh quality after refinement were avoided, and the use of the Laplace interpolant resolved the issue of element incompatibility. An efficient technique for stiffness matrix assembly on quadtree meshes was introduced. Numerical results for the Poisson equation, and linear and nonlinear elastostatic problems were presented. The improved accuracy and significant advantages of the proposed adaptive method were revealed in the numerical modeling of problems involving sharp gradients, singularities, and crack discontinuities.

8 Acknowledgment

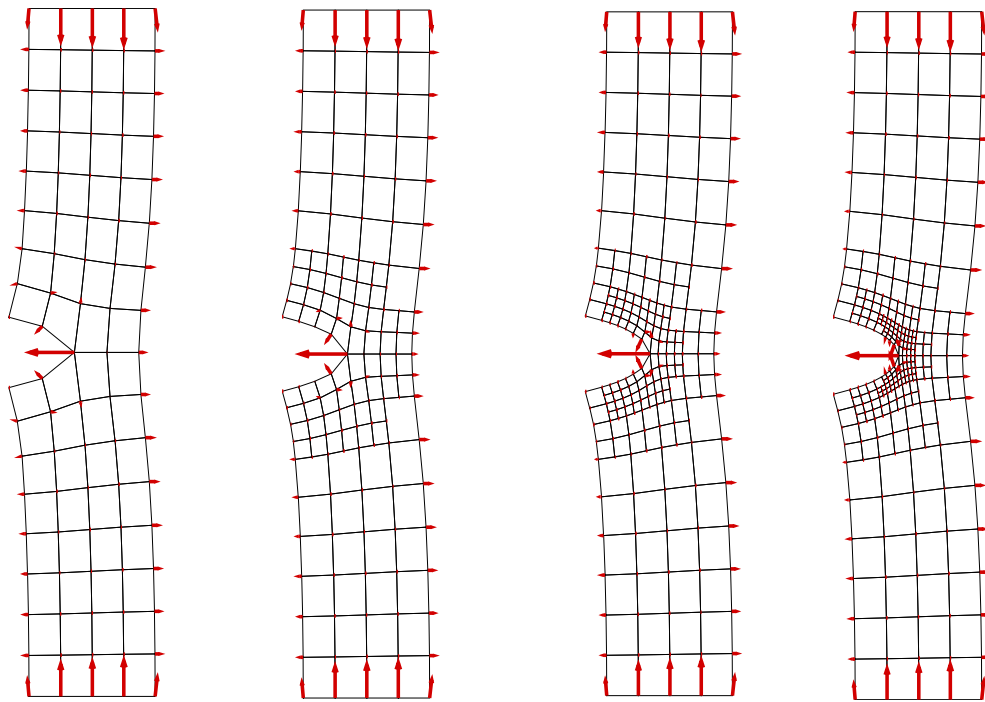
The authors thank John Bolander and John Burkardt for providing the codes to generate the Voronoi and centroidal Voronoi tessellations, respectively.

References

- [1] M. Ainsworth, B. Senior, Aspects of an adaptive hp -finite element method: Adaptive strategy, conforming approximation and efficient solvers, *Computer Methods in Applied Mechanics and Engineering* 150 (1997) 65–87.
- [2] N. Palle, J. A. Dantzig, An adaptive mesh refinement scheme for solidification problems, *Metallurgical and Materials Transactions* 27A (1996) 707–718.
- [3] N. Provatas, N. Goldenfeld, J. Dantzig, Adaptive mesh refinement computation of solidification microstructures using dynamic data structures, *Journal of Computational Physics* 148 (3) (1999) 265–290.
- [4] P. Hansbo, C. Lovadina, I. Perugia, G. Sangalli, A Lagrange multiplier method for the finite element solution of elliptic interface problems using non-matching meshes, *Numerische Mathematik* 100 (2005) 91–115.



(a)

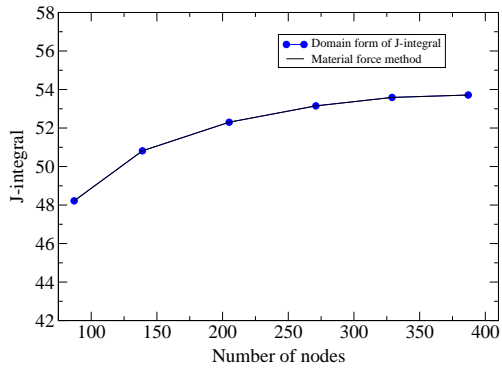


(b)

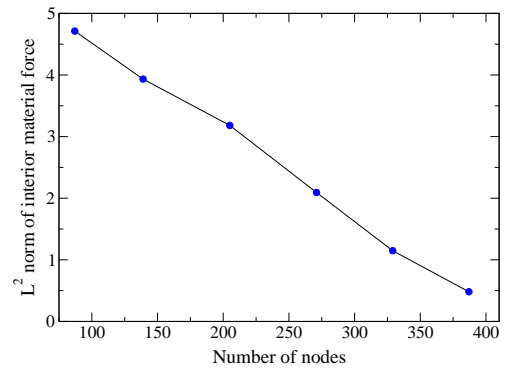
(c)

(d)

(e)



(f)



(g)

Fig. 30. The nonlinear plane strain edge-crack specimen. (a) Geometry and boundary conditions; (b) to (e) refined meshes; (f) J -integral and (g) L^2 norm of interior material nodal forces.

- [5] A. Hansbo, P. Hansbo, An unfitted finite element method, based on Nitsche's method, for elliptic interface problems, *Computer Methods in Applied Mechanics and Engineering* 191 (2002) 5537–5552.
- [6] P. Krysl, A. Trivedi, B. Zhu, Object-oriented hierarchical mesh refinement with CHARMS, *International Journal for Numerical Methods in Engineering* 60 (8) (2004) 1401–1424.
- [7] P. Krysl, P. Grinspun, E. Schröder, Natural hierarchical refinement for finite element methods, *International Journal for Numerical Methods in Engineering* 56 (8) (2003) 1109–1124.
- [8] P. Kagan, A. Fischer, P. Z. Bar-Yoseph, Mechanically based models: Adaptive refinement for B-spline finite element, *International Journal for Numerical Methods in Engineering* 57 (2003) 1145–1175.
- [9] A. Tabarraei, N. Sukumar, Adaptive computations on conforming quadtree meshes, *Finite Elements in Analysis and Design* 41 (2005) 686–702.
- [10] J. J. Laguardia, E. Cueto, M. Doblaré, A natural neighbour Galerkin method with quadtree structure, *International Journal for Numerical Methods in Engineering* 63 (2005) 789–812.
- [11] N. Sukumar, A. Tabarraei, Conforming polygonal finite elements, *International Journal for Numerical Methods in Engineering* 61 (2004) 2045–2066.
- [12] A. Tabarraei, N. Sukumar, Application of polygonal finite elements in linear elasticity, *International Journal of Computational Methods*, in press (2006).
- [13] M. Ainsworth, J. T. Oden, *A Posteriori Error Estimation in Finite Element Analysis*, John Wiley and Sons, New York, N.Y., 2000.
- [14] G. A. Maugin, *Material Inhomogeneities in Elasticity*, Chapman & Hall, New York, N.Y., 1993.
- [15] P. Steinmann, G. A. Maugin, *Mechanics of Material Forces*, Springer Verlag, New York, N.Y., 2005.
- [16] M. E. Gurtin, *Configurational Forces as Basic Concept of Continuum Physics*, Springer Verlag, New York, N.Y., 2000.
- [17] R. Kienzler, G. Herrmann, *Mechanics of Material Space with Applications in Defect and Fracture Mechanics*, Springer Verlag, New York, N.Y., 2000.
- [18] R. Muller, D. Gross, G. A. Maugin, Use of material forces in adaptive finite element methods, *Computational Mechanics* 33 (2004) 421–434.
- [19] P. Heintz, F. Larsson, P. Hansbo, K. Runesson, Adaptive strategies and error control for computing material forces in fracture mechanics, *International Journal for Numerical Methods in Engineering* 60 (2004) 1287–1299.
- [20] E. L. Wachspress, *A Rational Finite Element Basis*, Academic Press, N. Y., 1975.

- [21] M. Meyer, H. Lee, A. H. Barr, M. Desbrun, Generalized barycentric coordinates on irregular polygons, *Journal of Graphics Tools* 7 (1) (2002) 13–22.
- [22] M. S. Floater, Mean value coordinates, *Computer Aided Geometric Design* 20 (1) (2003) 19–27.
- [23] G. Dasgupta, Interpolants within convex polygons: Wachspress’ shape functions, *Journal of Aerospace Engineering* 16 (1) (2003) 1–8.
- [24] G. Dasgupta, Integration within polygonal finite elements, *Journal of Aerospace Engineering* 16 (1) (2003) 9–18.
- [25] E. A. Malsch, G. Dasgupta, Interpolations for temperature distributions: A method for all non-concave polygons, *International Journal of Solids and Structures* 41 (2004) 2165–2188.
- [26] N. Sukumar, Construction of polygonal interpolants: A maximum entropy approach, *International Journal for Numerical Methods in Engineering* 61 (2004) 2159–2181.
- [27] N. Sukumar, E. A. Malsch, Recent advances in the construction of polygonal finite element interpolants, *Archives of Computational Methods in Engineering* 13 (2006) 129–163.
- [28] A. Okabe, B. Boots, K. Sugihara, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, John Wiley & Sons, Chichester, England, 1992.
- [29] N. H. Christ, R. Friedberg, T. D. Lee, Weights of links and plaquettes in a random lattice, *Nuclear Physics B* 210 (3) (1982) 337–346.
- [30] N. Sukumar, B. Moran, A. Y. Semenov, V. V. Belikov, Natural neighbor Galerkin methods, *International Journal for Numerical Methods in Engineering* 50 (1) (2001) 1–27.
- [31] Q. Du, V. Faber, M. Gunzburger, Centroidal Voronoi tessellations: Applications and algorithms, *SIAM Review* 41 (1999) 637–676.
- [32] Q. Du, M. Gunzburger, L. Ju, Meshfree, probabilistic determination of point sets and regions for meshless computing, *Computer Methods in Applied Mechanics and Engineering* 191 (2002) 1349–1366.
- [33] J. Burkardt, CCVT_BOX: Centroidal Voronoi Tessellation Constrained to a Box, Available at http://www.csit.fsu.edu/~burkardt/f_src/ccvt_box/ccvt_box.html, 2004.
- [34] W. H. Press, S. A. Teukolsky, , W. T. Vetterling, B. P. Flannery, *Numerical Recipes in Fortran. The Art of Scientific Computing*, Cambridge University Press, New York, N.Y., 1992.
- [35] P. Lancaster, K. Salkauskas, Surfaces generated by moving least squares methods, *Mathematics of Computation* 37 (1981) 141–158.

- [36] K. Hormann, Barycentric Coordinates for Arbitrary Polygons in the Plane, Technical Report, Clausthal University of Technology, September 2004.
- [37] I. Babuška, W. C. Rheinboldt, A-posteriori error estimates for the finite element method, *International Journal for Numerical Methods in Engineering* 12 (1978) 1597–1615.
- [38] J. D. Eshelby, The force on an elastic singularity., *Phil. Trans. Roy. Soc. London A* 244 (1951) 87–112.
- [39] J. D. Eshelby, The elastic energy-momentum tensor, *Journal of Elasticity* 5 (1975) 321–335.
- [40] P. Steinmann, Application of material forces to hyperelastostatic fracture mechanics. I. Continuum mechanical setting, *International Journal of Solids and Structures* 37 (2000) 7371–7391.
- [41] P. Steinmann, D. Ackermann, F. J. Barth, Application of material forces to hyperelastostatic fracture mechanics. II. Computational setting, *International Journal of Solids and Structures* 38 (2001) 5509–5526.
- [42] R. Muller, G. A. Maugin, On material forces and finite element discretizations, *Computational Mechanics* 29 (2002) 52–60.
- [43] B. Moran, C. F. Shih, Crack tip and associated domain integrals from momentum and energy balance, *Engineering Fracture Mechanics* 27 (1987) 615–642.
- [44] J. Yau, S. Wang, H. Corten, A mixed-mode crack analysis of isotropic solids using conservation laws of elasticity, *Journal of Applied Mechanics* 47 (1980) 335–341.
- [45] J. Bonet, R. D. Wood, *Nonlinear Continuum Mechanics for Finite Element Analysis*, Cambridge University Press, New York, N.Y., 1997.
- [46] T. Belytschko, W. K. Liu, B. Moran, *Nonlinear Finite Elements for Continua and Structures*, John Wiley and Sons, New York, N.Y., 2000.
- [47] I. Babuška, D. Yu, Asymptotically exact a posteriori error estimator for biquadratic elements, *Finite Elements in Analysis and Design* 3 (1987) 341–354.
- [48] I. Babuška, A. Miller, A feedback finite element method with a posteriori error estimation: part I. the finite element method and some basic properties of the a posteriori error estimator, *Computer Methods in Applied Mechanics and Engineering* 61 (1987) 1–40.
- [49] C. Carstensen, R. Verfürth, Edge residuals dominate a-posteriori error estimates for low order finite element methods., *SIAM Journal on Numerical Analysis* 36 (1999) 1571–1587.
- [50] J. E. Pask, Private communication, 2006.

- [51] I. Babuška, T. Strouboulis, *The Finite Element Method and Its Reliability*, Oxford Science Publications, New York, N.Y., 2001.
- [52] H. Tada, P. C. Paris, G. R. Irwin, *The Stress Analysis of Cracks Handbook*, ASME Press, New York, N.Y., 2000.