Comput. Methods Appl. Mech. Engrg. 237–240 (2012) 152–165

Contents lists available at SciVerse ScienceDirect

Comput. Methods Appl. Mech. Engrg.

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

The partitioned element method in computational solid mechanics

M.M. Rashid [a,*], A. Sadri [a,1]

[a] Department of Civil and Environmental Engineering, University of California at Davis, Davis, CA 95616, USA

ARTICLE INFO

Article history:
Received 1 December 2011
Received in revised form 8 May 2012
Accepted 9 May 2012
Available online 24 May 2012

Keywords:
Finite element method
Polygonal elements
Partitioned element method

ABSTRACT

A finite-element-like approximation method is proposed for solid-mechanics applications, in which the elements can take essentially arbitrary polygonal form. A distinguishing feature of the method, herein called the "partitioned element method," is a partitioning of the elements into quadrature cells, over which the shape functions are taken to be piecewise linear. The gradient and constant values for each cell are determined by minimizing a quadratic function which represents a combined smoothness and compatibility measure. Linear completeness of the shape-function formulation is proved. Robustness in the presence of element non-convexity and geometric degeneracy (e.g. nearly coincident nodes) are particular goals of the method. Convergence for various 2D linear elasticity problems is demonstrated, and results for a finite-deformation elastic–plastic problem are compared to those of the standard FEM.

1. Introduction

As a general-purpose method for approximating the solutions to elliptic variational BVPs, the conventional isoparametric finite element method enjoys many powerful features. These include straightforward enforcement of both natural and essential boundary conditions, a high degree of quadrature efficiency, and a robust, flexible, and compact software expression. These advantages come, however, at the expense of having to construct a suitable mesh, and this requirement is usually cited as the conventional FEM's main drawback.

Any Galerkin approximation method consists of two essential elements: a mechanism to generate an arbitrary number of suitable basis functions, and a means of evaluating the weak-form integrals over the problem domain. An extensive array of variations on, and alternatives to, the standard FEM exists in the literature, in which various approaches are taken to these two main tasks. A method can be considered "meshless" if its basis functions are defined with reference to a set of nodes only, without the need for a partition of the domain into elements. Methods that instead require a domain partition for purposes of basis-function construction might be categorized as "finite-element-like".

All Galerkin methods must envisage a domain partition in some form, even if only to effect numerical quadrature. The topological requirements incumbent on quadrature partitions are, however, mild; the construction of such partitions is therefore less burdensome, algorithmically, than in cases where the partition is also used to define shape functions. Specifically, quadrature requires only that the d-dimensional problem domain be tessellated using d-dimensional shapes that are acceptable to the quadrature rule. For a tessellation to serve as a conventional finite element mesh, on the other hand, certain rules must be obeyed relating to how k-dimensional faces can be shared by the $k+1$-dimensional objects on whose boundaries they lie. For example, in a 3D finite element mesh, the intersection of the closure of two finite elements must either be null, or it must consist of a full facet of both elements. Partial overlaps are not allowed. These requirements are in addition to those on the shape and connectivity of the individual elements themselves, which derive from the isoparametric mapping to a canonical element.

Much of the early work on meshless methods emphasized the basis-function aspect, and indeed it is on the task of constructing basis functions that meshless methods have a particular strength. More recent work has illuminated the role of effective and efficient quadrature as an equally important contributor to the overall utility of an approximation method [1–4]. It seems clear that high quadrature efficiency, i.e. accurate evaluation of the weak-form integrals using only a few quadrature points, is promoted by basis-function supports that coincide with the quadrature cells. Accordingly, in the hope of approaching both the quadrature efficiency of FE methods and the discretization flexibility of meshless methods, meshless-type shape-function constructions have been adapted to finite-element-like discretizations by applying them locally, to individual polygonal/polyhedral elements [5–10]. Another, similar approach is to employ a boundary-integral-type formulation locally, within each polygonal element, to produce conforming shape functions [11,12]. A variation on this theme is found in [13],

[*] Corresponding author. Tel.: +1 530 752 7013; fax: +1 530 752 7872

E-mail addresses: mmrashid@ucdavis.edu (M.M. Rashid), alipasha.celeris@gmail.com (A. Sadri).

[1] Celeris LLC, San Diego, CA, USA.

0045-7825/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.cma.2012.05.014

where radial basis functions are used to construct harmonic shape functions on polyhedral elements, with inter-element compatibility being enforced approximately, via collocation.

It is clear that shape functions of finite-element type can be constructed on polygons and polyhedra in various ways, each with its own attributes regarding inter-element continuity, interior smoothness, and tolerance of element geometric pathologies such as non-convexity and nodal degeneracy. In all cases, the non-mapped character of the shape functions does bring a measure of freedom from the geometric and connectivity requirements attending conventional finite elements. But it also incurs a requirement that quadrature be carried out on irregular polytopes, on which the usual Gaussian product rules do not, in general, apply. To meet this challenge, a variety of approaches to quadrature on irregular domains has recently been proposed. In [14], the usual Gaussian approach is extended to irregular polygons by seeking weights and point locations that exactly integrate a polynomial basis. In later work [15], the same authors propose a class of quadrature rules on convex polytopes in which the points fall on the edges only. This latter approach uses a theorem by Lasserre [16] which reduces the integral of a homogeneous function over a convex polytope to a linear combination of integrals of the *same* function over the facets. An additive decomposition of the integrand into homogeneous terms must therefore be available. In [17], arbitrary (non-convex in general) polygons are mapped to regular polygons via Schwarz–Christoffel conformal mapping, where standard product rules are used in each triangular wedge.

In the standard FEM, the elements serve to define both the basis-function supports and the quadrature cells. A further attribute of the FEM also contributes to its remarkable quadrature efficiency, and that is the nature of the basis functions themselves: they are typically low-order polynomials in the mapped coordinates. Such functions can be accurately integrated using Gaussian product rules of very low order, even when the physical element is quite distorted.

In the present paper, a Galerkin method is proposed which attempts to achieve this same synergy, but without using a mapping to a canonical element domain where Gaussian product rules would apply. The new method, here termed the "partitioned element method" (PEM), aspires to the same goal as do most other FEM alternatives: FEM-like performance, particularly with regard to quadrature, without FEM-like restrictions on the construction of the spatial discretization. The method is finite-element-like rather than meshless, in the sense that the domain is first partitioned into "elements," which are used to facilitate both shape-function construction and quadrature. A distinguishing feature of the PEM is that the elements are further subdivided into quadrature cells. This subdivision can be done in many different ways, with corresponding differences in the performance characteristics of the overall method.

The crux of the PEM is that the shape functions and their gradients are formulated in terms of their discrete values relative to the quadrature cells, instead of as continuous functions of position on the entire element. In this sense, the quadrature rule and the shape functions are formulated in a symbiotic way. As a result, the shape-function formulation is tolerant of degeneracies in the element geometry, such as non-convexity (even if extreme), very small aspect ratios, and node pairs that are arbitrarily close together. Under these circumstances, the PEM shape-function formulation remains free of numerical pathologies, and renders results that support high-quality global approximations despite the presence of troublesome element geometry. This robustness is valuable in the context of automatic mesh-generation strategies, in which well-proportioned polygonal or polyhedral elements generally cannot be guaranteed. Stated another way, if the goal of a polygonal/polyhedral finite element method is to make automatic meshing easier, then this goal is better realized if the method can effectively work with whatever polygons/polyhedra a simple domain-partitioning algorithm produces.

In previous work, polygonal [18] and polyhedral [19] methods were developed in which the shape functions take the form of low-order polynomials (typically 3rd or 4th) in the physical Cartesian coordinates. This method is non-conforming, but was proven to converge based on the fact that the polynomial coefficients are constrained to produce zero-mean jumps at inter-element boundaries [20]. This and other constraints, along with an optimal smoothness criterion, were invoked to uniquely determine the polynomial coefficients. This method is reasonably tolerant of non-convexity if not too severe, but ultimately breaks down as coincident-node degeneracy is approached.

A conforming polygonal element method is proposed in [5], in which an isoparametric mapping to a regular n-gon is used to effect shape-function construction and quadrature. A natural-neighbors approach is used to define rational basis functions on the elements. As with the conventional FEM, the isoparametric mapping engenders a restriction to convex elements. A similar approach was taken in 3D, although without the isoparametric mapping to a canonical element, in [6]. These methods can be considered as extensions of the conventional FEM to n-sided polygonal/polyhedral elements, in the sense that the elements define both the shape-function supports and the quadrature-cell boundaries. It is interesting to note, however, the relatively large number of integration points required to achieve reasonable solution accuracy. This is likely due to the rational nature of the shape functions.

The error introduced by inexact numerical quadrature can be studied, in relation to a particular BVP, by defining an "integration error" as a suitable norm (e.g. the energy norm) of the difference between the solution obtained using exact integration, and that obtained with approximate quadrature [21]. A central question is, how accurate must the quadrature rule be in order that the integration error approach zero under mesh refinement? Further, how accurate must the quadrature be so that the overall solution converges to the exact one at an undiminished rate? In the case of second-order elliptic BVPs with constant coefficients, the answer to the first question is that the first derivatives of all basis functions must be exactly integrated by the quadrature formula. This condition is satisfied, somewhat fortuitously, by isoparametric finite elements with polynomial bases, even when the order of the quadrature formula is very low. In other cases, it must be checked explicitly. This can be done via a "quadrature consistency" test, wherein the divergence theorem is applied to individual elements. In this test, the quadrature rule is used to compute the domain integral, while the boundary integral is evaluated exactly. Passage of the quadrature-consistency test is necessary if patch tests are to be passed; its absence explains why strict passage of patch tests is not observed when rational basis functions are used. Quadrature consistency is discussed in [19] in the context of the above-mentioned polynomial-based polygonal/polyhedral FE method; and in [2] in the context of Galerkin mesh-free methods.

Geometric complexity of the domain is usually cited as the main motivation for polygonal or polyhedral FE-type methods. Another approach to analysis on geometrically complex domains exploits the idea that a conventional FE mesh need not conform to the domain boundary, provided that care is taken to integrate only over the portions of the elements that lie inside the domain. This notion is generally associated with the terms "generalized finite element method" (GFEM) [22] or "extended finite element method" (XFEM) [23], although it is obviously distinct from a second important feature of these methods – use of an enriched nodal basis to capture special solution features. Integration over "partial elements" was also used in [24] in the context of arbitrary crack paths in 2D domains, and in [25] in the "implicit boundary method".

Finally we mention a collection of recently-proposed approximation methods called the "smoothed finite element method" (SFEM) [26]. This method is an adaptation of the technique used in [2] and in [1] to devise a stabilized nodal integration scheme in meshless methods. The essence of the SFEM is a partition of the elements into quadrature cells, over which the shape-function gradients are averaged. The piecewise-constant cell-averaged shape-function gradients are then used to form the weak-form integrals. In the averaging process, the divergence theorem is used to reduce the gradient averages to integrals over the cell boundaries. In the original conception of the SFEM [26], otherwise-conventional isoparametric elements were used, with later extensions to polygonal elements [27] and to node-centered [28] and edge-centered [29] quadrature cells. In [27], for example, n-sided polygonal elements are partitioned into n triangular wedges, with the mean vertex coordinates used as the common apex. If one then assumes that the shape functions vary linearly over the interior segments, only a single value remains unknown for each shape function – the value at the common apex. In [27], this value is simply taken to be $1/n$, regardless of element geometry. Similar constructions are used in the node-based [28] and edge-based [29] variants of the method. In the context of thin-plate theory, a similar idea is used in [30] in connection with a reproducing-kernel approximation. Specifically, the domain is partitioned into nodal domains, which are then further subdivided for the purpose of smoothing the curvatures. The smoothing step entails integration of the RK shape-function gradients over the subdomains.

The method proposed herein shares with the SFEM the idea of partitioning the element into quadrature cells, and then calculating cell-averaged shape-function gradients via assumed piecewise-linear variation on a network of segments. Where the methods differ is in the manner in which the shape-function values at the network's interior vertices are calculated. In short, the PEM accommodates essentially arbitrary cell-boundary networks with multiple interior vertices, thus allowing for considerable flexibility in the quadrature-cell design. A combined smoothness and inter-cell compatibility criterion is used to form a minimization problem, from which the interior-vertex shape-function values emerge. The approach is explained in full detail in the Section 2. An algorithm with which to carry out the associated element-level calculations is presented in Section 3. Section 4 is devoted to numerical exposition of the performance of the method, with concluding remarks given in Section 5.

2. Element formulation

The partitioned element method (PEM) is a finite element method in the sense that it begins with a partition of the domain into non-overlapping elements, which are then used to define shape functions and to effect quadrature. The PEM departs from the conventional FEM in the manner in which the shape functions and quadrature rule are defined. The PEM does not use the notion of a parent element or isoparametric transformation; instead, the quadrature rule is constructed via a geometric partition of the element in its physical configuration. The shape functions are then generated in a manner which is closely entwined with the quadrature cells. Specifically, the shape functions and their gradients are characterized by sets of discrete values associated with the quadrature cells. This contrasts with the conventional approach, in which the shape functions are specified as analytical functions over the entire element, with the gradients evaluated at the quadrature points for purposes of evaluating weak-form integrals.

Conceptually, the approach taken here can be seen from two distinct points of view. In the first, it is imagined that conventional conforming shape functions exist, but are not explicitly defined pointwise over the element. Instead, in recognition of the fact that only the quadrature-point values are ever needed in actual computations, approximations to these discrete values are sought without actually defining, analytically, the shape functions over the whole element. These discrete values should reflect certain properties possessed by the continuous (but unknown) shape functions, such as consistency with prescribed boundary values and smoothness on the interior of the element.

In the second point of view, the shape functions are taken to be defined pointwise over the entire element, with linear variation over each quadrature cell. These piecewise-linear functions are, in general, discontinuous at inter-cell boundaries, as well as at inter-element boundaries. In this conception, the PEM is viewed as a non-conforming method, although a somewhat unconventional one: shape-function discontinuities occur not only at the inter-element boundaries, but within the element interiors as well. As with all non-conforming methods, proof of convergence hinges on approximability of the approximation space, as well as on a consistency requirement [31]. Which of these two points of view is adopted is largely a matter of taste, as the choice does not influence the development itself. However, the non-conforming piecewise-linear framing facilitates a convergence analysis that draws upon well-established results.

In the development that follows, we consider a single polygonal "element," which need not be convex nor simply connected (see Fig. 1). The elements will be partitioned into polygonal "cells" as explained in Section 2.1 below. Both the element and its cell structure are defined by straight "segments," which are joined together at their endpoints. Segment endpoints will be called "vertices;" all vertices are retained in the geometric definition of an element, even those that are shared only by two colinear segments. Some of the vertices on the boundary of the element will be associated with element shape functions. Such vertices are additionally referred to as "nodes." In the present paper, we confine ourselves to the case where a shape function is associated with each vertex on the element's boundary whose included angle $\neq \pi$. I.e., for the present purposes, nodes occur at all the "corners." There are reasons to generalize this picture to the case where some corners do not have associated shape functions, and therefore are not nodes as the term is used herein. For example, should two corners be very close together, it may be desirable to place a node at only one of them, for reasons of computational efficiency. This is especially true if the closely-spaced corners occur merely as an artifact of the mesh-generation process, and are not intentionally introduced to resolve some expected feature of the solution. One scenario in

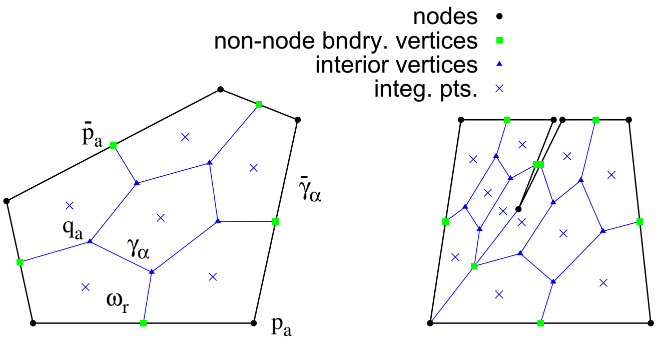

Fig. 1. Two representative PEM elements, showing quadrature cells and integration points. The element at right is severely non-convex due to the crack-like feature; this element was subdivided into two macro-cells.

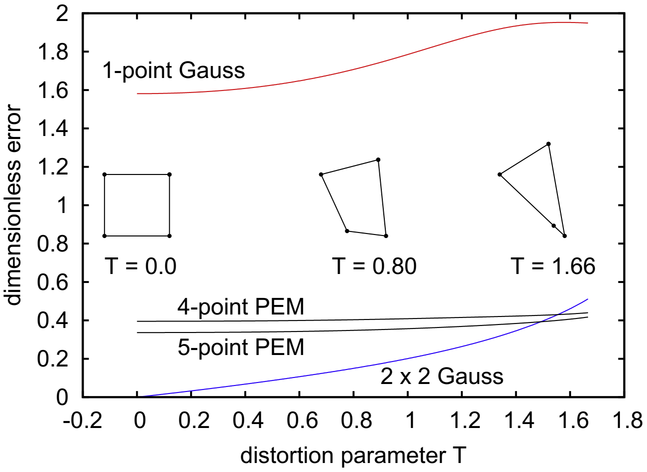

Fig. 2. Dimensionless error e (Eq. 3) for integration of cubic polynomials over a distorted quadrilateral, for 5-point PEM ($\xi = 0.5$), 4-point PEM (no centroidal cell), 2×2 Gauss, and 1-point Gauss quadrature rules.

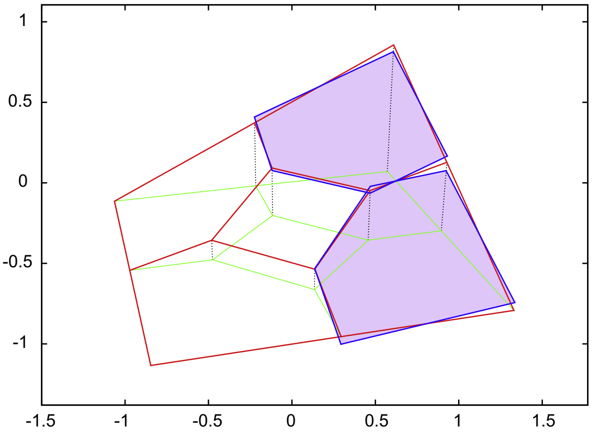

Fig. 3. Plot of the piecewise-linear variation of a shape function on a square PEM element, over two of the element's integration cells.

which this is likely to occur is where the mesh is generated via Boolean geometric operations on, say, a structured hex mesh, producing elements in the form of "cut cells." In any case, we defer such generalizations to a future communication, while observing that, even in the case of closely-spaced nodes, no ill-conditioning in the shape-function formulation occurs.

In Sections 2.1–3, calculation of the quadrature rule, shape function gradients, and shape-function constant terms, respectively, is explained. Then, in Section 2.4 it is proved that PEM shape functions exhibit linear completeness.

2.1. Quadrature rule

Consider a polygonal element Ω with nodes $p_a,\ a = 1,\ldots,N$, each of which has an associated shape function of nominally finite element type. The element need not be convex, nor even simply connected. We take a geometric approach to the formulation of the element's quadrature rule, whereby the element is partitioned into subregions that constitute quadrature cells. The integration weights are taken to be the cell areas, while the locations of the integration points are the cell centroids.

Qualitatively, the quadrature cells should be designed so that interpolating nodal shape functions can be represented reasonably accurately with discontinuous-piecewise-linear functions on the cell complex. We emphasize that an element's cell complex serves to define not only the quadrature rule for the element, but also the piecewise-linear representation of the shape functions. A cell design's effectiveness in relation to the later function is at least as important to the overall performance of the method as its effectiveness in relation to the former. An examination of the conventional bilinear quadrilateral element suggests that interpolating nodal shape functions can be reasonably-well represented by associating a cell with each of the nodes, with an additional cell that contains the element's centroid (Fig. 1, left). A plot of a piecewise-linear representation of a nodal shape function on a square element can be seen in Fig. 3. For elements with obtuse or even reflex (i.e. $\geqslant \pi$) angles at some nodes, a better approximation to the node's "actual" shape function can be had by ensuring that a cell boundary passes through these nodes.

With these observations in mind, we define the cell boundaries as follows. First, the element is subdivided into a minimal or near-minimal number of convex macro-cells. Elements that are convex to begin with require no macro-cell subdivision. Also, strict convexity of the macro-cells is not important; the point of the macro-cell subdivision is to mitigate severe non-convexity in the element geometry. Then, each macro-cell is partitioned into quadrature cells by connecting the mid-sides of the macro-cell to its centroid. Also, nodes at which large angles occur can be connected to the centroid, effectively splitting cells that were formed by the mid-side-to-centroid connectors. Interior vertices are placed at points given by the fraction ξ of the distance from mid-side (or node) to centroid, and these interior vertices are then linked together to form a cell that contains the centroid. ξ is a parameter in this cell design; its selection potentially influences the accuracy of the overall method. Some insight into the effects of ξ is provided later in this subsection and in Section 4.2, in relation to quadrature accuracy and overall solution accuracy, respectively. In general, we have found the influence of ξ to be mild; accordingly, a nominal value of 0.5 is recommended for general use. This value of ξ was used in all example computations reported in Section 4. The partitioning method is illustrated in Fig. 1 for one convex and one non-convex element. The method guarantees convex quadrature cells, provided a suitable convex subdivision of the element into macro-cells can be found. However, it is noted that convexity of the cells is not strictly necessary.

With reference to Fig. 1, the cells are denoted as $\omega_r,\ r = 1,\ldots,K$, where $\cup_r\ \omega_r = \Omega$ and $\omega_r \cap \omega_s = \emptyset$ for $r \neq s$. The partition of the element into cells introduces interior vertices $q_a \in \Omega$, as well as additional vertices on the element boundary $\bar{p}_a \in \partial\Omega$. The three sets of points $\{p_a,\ a = 1,\ldots,N\}$, $\{\bar{p}_a,\ a = 1,\ldots,\overline{N}\}$, and $\{q_a,\ a = 1,\ldots,n\}$ constitute the full set of endpoints of the boundary segments of the cells. These boundary segments are denoted $\{\gamma_\alpha,\ \alpha = 1,\ldots,M\}$ in the case of cell-cell interfaces interior to the element, and $\{\bar{\gamma}_\alpha,\ \alpha = 1,\ldots,\overline{M}\}$ in the case of segments that lie on $\partial\Omega$. The set $\{\bar{\gamma}_\alpha\}$ constitutes a partition of $\partial\Omega$, while $\{\gamma_\alpha\}$ contains all interfaces between the cells. Further, we define sets of indices $\overline{Q}^r$ and Q^r such that

$$\partial\omega_r = \{\cup_{\alpha\in\overline{Q}^r}\bar{\gamma}_\alpha\}\ \cup\ \{\cup_{\alpha\in Q^r}\gamma_\alpha\}. \tag{1}$$

The normal vectors to each $\bar{\gamma}_\alpha$ and γ_α are given by $\bar{\mathbf{n}}_\alpha$ and $\mathbf{n}_\alpha$ respectively, where the $\bar{\mathbf{n}}_\alpha$ point outward from Ω. The outward sense of $\mathbf{n}_\alpha$ relative to ω_r is determined by the signed indicator $z^r_\alpha = -1, 0,$ or 1, such that if γ_α lies on the boundary of ω_r, then $z^r_\alpha\mathbf{n}_\alpha$ (no sum on α) is directed away from ω_r. If γ_α does not lie on the boundary of ω_r, then $z^r_\alpha = 0$. These and other notational conventions are summarized in Table 1.

With the partition of the element in hand, the quadrature rule is effected by placing a single integration point at the centroid of each cell, and assigning its weight to be the area of the cell. This rule is exact for linear functions on the element, and otherwise results in

Table 1
Element-formulation notation.

Entity	Symbol	Vector length	Associated location
Node	p_a	N	On $\partial\Omega$
Non-node boundary vertex	$\bar{p}_a$	$\overline{N}$	On $\partial\Omega$
Interior vertex	q_a	n	Interior of Ω
Quadrature cell	ω_r	K	Interior of Ω
Interface segment	γ_α	M	Interior of Ω
Boundary segment	$\bar{\gamma}_\alpha$	$\overline{M}$	$\partial\Omega$
Interface normal	$\mathbf{n}_\alpha$	M	γ_α
Boundary normal	$\bar{\mathbf{n}}_\alpha$	$\overline{M}$	$\bar{\gamma}_\alpha$
Normal dir. indicator	z_α^r	MK	γ_α
Nodal function value	$\mathbf{v}$ (components v_a)	N	p_a
Bndry.-point func. value	$\bar{\mathbf{v}}$ (components $\bar{v}_a$)	$\overline{N}$	$\bar{p}_a$
Interior-point func. value	$\mathbf{u}$ (components u_a)	n	q_a
Cell gradient	$\mathbf{g}_r$ (components g_{ir})	K	Average over ω_r
Cell function value	φ_r	K	Selected point in ω_r
$\int_{\gamma_\alpha} \varphi ds$	Φ_α	M	γ_α
$\int_{\bar{\gamma}_\alpha} \varphi ds$	$\bar{\Phi}_\alpha$	$\overline{M}$	$\bar{\gamma}_\alpha$

accurate estimates for functions that are well-represented by piecewise-linear approximants on the assemblage of cells. The integration-point weight $|\omega_r|$ and location $\mathbf{x}_r$ are computed using the divergence theorem; with the foregoing notation, the formulae are

$$|\omega_r| = \int_{\omega_r} da = \frac{1}{2}\int_{\partial\omega_r} \mathbf{x}\cdot\mathbf{n}ds = \frac{1}{2}\sum_{\alpha\in\overline{Q}^r}\int_{\bar{\gamma}_\alpha} \mathbf{x}\cdot\bar{\mathbf{n}}_\alpha ds + \frac{1}{2}\sum_{\alpha\in Q^r}\int_{\gamma_\alpha} z_\alpha^r\mathbf{x}\cdot\mathbf{n}_\alpha ds,$$
$$|\omega_r|\mathbf{x}_r = \int_{\omega_r} \mathbf{x}da = \frac{1}{3}\int_{\partial\omega_r} \mathbf{x}\mathbf{x}\cdot\mathbf{n}ds = \frac{1}{3}\sum_{\alpha\in\overline{Q}^r}\int_{\bar{\gamma}_\alpha} \mathbf{x}\mathbf{x}\cdot\bar{\mathbf{n}}_\alpha ds + \frac{1}{3}\sum_{\alpha\in Q^r}\int_{\gamma_\alpha} z_\alpha^r\mathbf{x}\mathbf{x}\cdot\mathbf{n}_\alpha ds. \tag{2}$$

Some insight into the accuracy of this quadrature rule can be had by considering the dimensionless quadrature error e, defined by

$$e^2 = \sup_{w\in V}\frac{1}{|\Omega|\|w\|^2}\left[\int_\Omega w dv - Q(w)\right]^2, \tag{3}$$

where Ω is the domain of integration, V is a space of square-integrable functions defined on Ω, $\|w\|$ is the L_2-norm on V, and $Q(w)$ is the result of applying the quadrature rule to the function w. This measure emerges by considering the quadrature rule to be a functional defined on the function space from which integrands are drawn; e is essentially the norm of this functional. Accordingly, the supremum in (3) can be computed by finding the minimum eigenvalue of a certain matrix. Quadrature-error measures of this type are advocated in [32]. In essence, the error e measures the "worst-case" error incurred by using the subject quadrature rule on functions selected from the space V. Though e is dimensionless, it cannot be regarded as a "percentage error," strictly speaking, because the denominator is not equal to the square of the exact integral.

The error e was computed for a one-parameter family of quadrilateral domains Ω using two versions of the above-described cell-based PEM integration rule, and, for comparison, 2 × 2 Gaussian integration, and one-point Gaussian integration. The quadrilateral distorts continuously with the parameter T, varying from the bi-unit square at $T = 0$ to incipient non-convexity at T's maximum value. Specifically, the nodal coordinates for the element's four nodes are given by: $(1-0.3T, 1+0.6T)$, $(-1,1)$, $(-1+T,-1+0.2T)$, $(1,-1)$. Gaussian integration was effected by employing the usual isoparametric transformation to a bi-unit square, whereas the PEM integration rule was applied in the domain's physical configuration. For purposes of illustration, V was selected as the complete space of third-order polynomials in two dimensions; i.e. $V = span\{1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2, y^3\}$. Third order was chosen because this is the highest polynomial order that is exactly integrated by 2 × 2 Gaussian quadrature on the undistorted bi-unit square. Departure from zero error as the element distorts is therefore entirely due to the effects of the isoparametric mapping.

The results of the error calculation are shown in Fig. 2 for 2 × 2 and 1-point Gaussian quadrature, for a 5-point PEM quadrature rule with a centroidal cell using $\xi = 0.5$, and for a 4-point PEM rule that lacks a centroidal cell. The progression of the element's distortion is shown in the figure, at three discrete values of T. Although the PEM integration rules remain effective after the element becomes non-convex, the plot is truncated at the point where the jacobian first goes to zero at a vertex, because the transformation used in the Gaussian integration cases becomes non-invertible at this point. The conventional 2 × 2 rule is exact for cubic functions on an undistorted element, but with nonzero distortion the jacobian brings in factors that are linear in each coordinate, causing the integrand to become fourth-order in the parent-domain coordinates. The plot suggests that the PEM rules described here, both with and without a centroidal cell, exhibit an accuracy that is comparable to quadrature rules typically used with conventional finite elements. It is further remarked that the PEM rule is applicable to polygons of rather general shape, whereas Gaussian product rules only apply to elements that are mappable to a square or triangle.

Although the error measure (3) is useful for comparing integration rules in a consistent manner, it bears emphasis that the suitability of an integration rule for a given application strongly depends on the nature of the functions to be integrated. In the PEM case, as will be seen below, typically the integrands are piecewise-constant functions on the collection of quadrature cells. For such functions, the PEM rule is of course exact. The design of the quadrature-cell partition is, therefore, not so much a question of quadrature accuracy as it is of accurate representation of the element's shape functions and gradients.

2.2. Shape-function gradients

The problem of constructing the shape functions can be stated as follows. Given an N-vector $\mathbf{v}$ of function values at the element's nodes p_a, find the discrete values c_r and gradients $\mathbf{g}_r$, $r = 1, \ldots, K$, associated with the element's quadrature cells. The relationship between $\mathbf{v}$ and $\{c_r, \mathbf{g}_r\}$ should be linear, and should reflect optimal smoothness on the element interior in an appropriately defined sense. The cell values should also be consistent with the boundary variation of the underlying function φ, which is controlled by the nodal values $\mathbf{v}$. We further impose the requirement that, when $\mathbf{v}$ is consistent with a globally linear function, then $\{\varphi_r, \mathbf{g}_r\}$ should be consistent with the same linear function. This linear-completeness condition implies approximability, which is required for convergence of the overall finite element method. If $\mathbf{v}$ is set consistent with the shape function associated with node b, i.e. $v_b = 1$ and $v_a = 0$ for $a \neq b$, then the resulting output $\mathbf{g}_r$ values will be the average shape-function gradients over each cell, for shape function b.

The boundary-variation consistency condition is formulated by applying the divergence theorem to each quadrature cell:

$$|\omega_r|\mathbf{g}_r = \int_{\omega_r} \nabla\varphi da = \int_{\partial\omega_r} \varphi\mathbf{n}\, ds. \tag{4}$$

Introducing the symbols

$$\bar{\Phi}_\alpha = \int_{\bar{\gamma}_\alpha} \varphi\, ds, \quad \Phi_\alpha = \int_{\gamma_\alpha} \varphi\, ds, \tag{5}$$

Eq. (4) can be rewritten

$$|\omega_r|\mathbf{g}_r = \sum_{\alpha\in\bar{Q}^r} \bar{\Phi}_\alpha \bar{\mathbf{n}}_\alpha + \sum_{\alpha\in Q^r} z^r_\alpha \Phi_\alpha \mathbf{n}_\alpha, \tag{6}$$

The function φ for a given vector $\mathbf{v} = (v_1, \ldots, v_N)$ of nodal values is taken to be continuous and piecewise linear on $\partial\Omega$, i.e. $\varphi\rfloor_{\partial\Omega} = \sum_a v_a \psi_a$, where ψ_a are 1D finite-element-type "hat functions," one for each node p_a. Defining an $\bar{N}$-vector $\bar{\mathbf{v}}$ (with components $\bar{v}_a$) of function values at the non-node boundary vertices $\bar{p}_a$, and recalling that all corners on the element boundary are nodes, it is evident that $\bar{\mathbf{v}}$ is determined by linear interpolation from $\mathbf{v}$. This leaves only the n-vector $\mathbf{u}$ of interior-vertex values u_a as unknowns. The calculation of these unknowns forms the crux of the element formulation, and is explained below.

The segments $\{\gamma_\alpha, \bar{\gamma}_\alpha\}$ and vertices $\{p_a, \bar{p}_a, q_a\}$ define a planar network; the function φ is taken to be continuous and piecewise linear on this network. From this assumption, the integrals Φ_a and $\bar{\Phi}_a$ (see Eq. 5) can be expressed in terms of the vertex values as

$$\Phi_\alpha = \sum_{a=1}^{N} P_{\alpha a} v_a + \sum_{a=1}^{n} R_{\alpha a} u_a, \quad \bar{\Phi}_\alpha = \sum_{a=1}^{N} \bar{P}_{\alpha a} v_a. \tag{7}$$

These relations reflect the facts that interior segments can have endpoints of any of the three vertex types, while boundary-segment endpoints can be nodes and non-node boundary vertices only; and further that non-node boundary-vertex values can be eliminated in favor of nodal values by linear interpolation. The matrices $P_{\alpha a}, R_{\alpha a}$, and $\bar{P}_{\alpha a}$ are sparse; specifically, for a fixed α each matrix will have nonzeros for at most two values of a.

Substitution of (7) into (6) results in cell-averaged gradient values given in terms of the vector $\mathbf{v}$ of known nodal values, and the vector $\mathbf{u}$ of unknown interior-vertex values:

$$\mathbf{g}_r = \sum_{a=1}^{N} \mathbf{k}^r_a v_a + \sum_{a=1}^{n} \mathbf{m}^r_a u_a, \tag{8}$$

where

$$|\omega_r|\mathbf{k}^r_a = \sum_{\alpha\in\bar{Q}_r} \bar{\mathbf{n}}_\alpha \bar{P}_{\alpha a} + \sum_{\alpha\in Q_r} z^r_\alpha \mathbf{n}_\alpha P_{\alpha a}, \quad |\omega_r|\mathbf{m}^r_a = \sum_{\alpha\in Q_r} z^r_\alpha \mathbf{n}_\alpha R_{\alpha a}. \tag{9}$$

The present objective is to select, for a given $\mathbf{v}$, the vector $\mathbf{u}$ such that the resulting cell-averaged gradients (8) have desirable qualities, in some sense. In the present context, "desirability" can be associated with freedom from oscillation or other excursions within the element that are unnecessary for purposes of maintaining consistency with φ's boundary variation. This seems appropriate, in light of the fact that the intended use of the method is approximation of solutions to elliptic BVPs, whose solutions are typically smooth. The selection of polynomial shape functions for conventional isoparametric finite elements represents a parallel: in the conventional method, usually the lowest-order polynomials needed to meet the required boundary variation are used. In any case, the interior-vertex values $\mathbf{u}$ are, so far, arbitrary. In order that they may be uniquely determined, a condition of some kind must be imposed. Based on these considerations, for a given $\mathbf{v}$ we seek to set $\mathbf{u}$ according to a discrete version of the variational problem

$$\text{minimize} \int_\Omega \nabla\varphi \cdot \nabla\varphi da \text{ subject to given } \varphi\rfloor_{\partial\Omega}, \tag{10}$$

which constitutes a measure of the interior "smoothness" of φ. The Euler–Lagrange equation for (10) is the Laplace equation with Dirichlet boundary condition. A discrete version of (10) can be had by approximating $\nabla\varphi$ by a piecewise-constant field of cell-averaged values $\mathbf{g}_r$. The functional in (10) then becomes

$$\mathfrak{F} = \sum_{r=1}^{K} |\omega_r| \mathbf{g}_r \cdot \mathbf{g}_r. \tag{11}$$

Substitution of (8) into (11) leads to a small, well-posed quadratic minimization problem for the interior-vertex values u_a in terms of the nodal values v_a.

Unfortunately, proceeding on the basis of minimization of (11), in its present form, often leads to unsatisfactory cell gradients $\mathbf{g}_r$. Specifically, the $\mathbf{g}_r$ can all turn out to be identical for given nodal values $\mathbf{v}$, even when these nodal values depart from linearity. This behavior can be understood by considering that the average gradient over the entire element, which can be determined from the boundary values alone via the divergence theorem, provides an absolute lower bound for $\mathfrak{F}$ (and also, for that matter, on the functional in Eq. (10)). For given $\mathbf{v}$, then, if there exists a $\mathbf{u}$ such that the cell-averaged gradients $\mathbf{g}_r$ come out to be identical for all cells, then this $\mathbf{u}$ is the one selected through minimization of (11). Depending on the cell geometry, such a $\mathbf{u}$ often does exist, but usually exhibits rather large fluctuations in the interior-point values u_a, especially for closely-spaced interior vertices.

A solution to this problem can be had by modifying the objective function (11) to include a measure of the inconsistency, on $\partial\omega_r$, between the cell gradient and the piecewise-linear variation of φ on the segment network. Specifically, (11) is replaced by

$$\mathfrak{F} = \beta\sum_{r=1}^{K} |\omega_r| \mathbf{g}_r \cdot \mathbf{g}_r + (1-\beta)\sum_{r=1}^{K} |\partial\omega_r| \int_{\partial\omega_r} \left(\frac{d\varphi_r}{ds} - \frac{d\varphi}{ds}\right)^2 ds, \tag{12}$$

where $\beta \in [0, 1]$ is a numerical parameter. In practice, the end-result cell gradients are rather insensitive to β over most of its range. In particular, values in the range $[0.1, 0.9]$ were tried with the example problems presented in Section 4, with negligible effect on the energy error. The nominal value $\beta = 0.5$ is therefore recommended for general use. In (12), φ_r represents the piecewise-linear approximation to φ on ω_r (see Eq. 19 below), while s is an arc parameter on $\partial\omega_r$.

Eq. (12) represents a quadratic function of the nodal values v_a and the interior-point values u_a, which is to be minimized with respect to u_a for given v_a. The first term of (12) can be written as an explicit function of these point values by using (8). To treat the second term, we first define unit tangent vectors λ_α and $\bar{\lambda}_\alpha$ on the segments γ_α and $\bar{\gamma}_\alpha$ respectively, where for definiteness the directions are taken such that $\lambda_\alpha \times \mathbf{e}_3 = \mathbf{n}_\alpha$ and $\bar{\lambda}_\alpha \times \mathbf{e}_3 = \bar{\mathbf{n}}_\alpha$. Thus $d\varphi_r/ds$ in (12) can be written as $\mathbf{g}_r \cdot \lambda$, where λ stands for λ_α or $\bar{\lambda}_\alpha$, depending on the segment of $\partial\omega_r$. Next, in a similar fashion to (5), coefficients $p_{\alpha a}, \bar{p}_{\alpha a}$, and $r_{\alpha a}$ are introduced such that the derivative of φ on the segments of the network can be written

$$\begin{aligned} \frac{d\varphi}{ds} &= \sum_{a=1}^{N} p_{\alpha a} v_a + \sum_{a=1}^{n} r_{\alpha a} u_a \text{ on } \gamma_\alpha, \\ &= \sum_{a=1}^{N} \bar{p}_{\alpha a} v_a \text{ on } \bar{\gamma}_\alpha, \end{aligned} \tag{13}$$

where the derivative is taken in the direction of the corresponding λ_α or $\bar{\lambda}_\alpha$. Eqs. (8) and (13) then lead to

$$\int_{\partial\omega_r}\left(\frac{d\varphi_r}{ds}-\frac{d\varphi}{ds}\right)^2 ds = \sum_{\alpha\in Q_r}|\gamma_\alpha|\left[\sum_a(\mathbf{k}_a^r\cdot\lambda_\alpha - p_{\alpha a})v_a + \sum_a(\mathbf{m}_a^r\cdot\lambda_\alpha - r_{\alpha a})u_a\right]^2 + \sum_{\alpha\in\overline{Q}_r}|\bar{\gamma}_\alpha|\left[\sum_a(\mathbf{k}_a^r\cdot\bar{\lambda}_\alpha - \bar{p}_{\alpha a})v_a + \sum_a(\mathbf{m}_a^r\cdot\bar{\lambda}_\alpha)u_a\right]^2. \tag{14}$$

By using (14) and (8), (12) can be recast as

$$\mathfrak{F} = \sum_{a=1}^{n}\sum_{b=1}^{n}u_a A_{ab} u_b + 2\sum_{a=1}^{n}\sum_{b=1}^{N}u_a B_{ab} v_b + C, \tag{15}$$

where C is a constant independent of u_a, and where

$$A_{ab} = \beta\sum_r|\omega_r|\mathbf{m}_a^r\cdot\mathbf{m}_b^r + (1-\beta)\overline{A}_{ab},$$
$$B_{ab} = \beta\sum_r|\omega_r|\mathbf{m}_a^r\cdot\mathbf{k}_b^r + (1-\beta)\overline{B}_{ab}. \tag{16}$$

In (16), $\overline{A}_{ab}$ and $\overline{B}_{ab}$ are assembled from the contributions (14) from individual cell boundaries.

The coefficient matrix A_{ab} in (15) is symmetric positive definite, thus $\mathfrak{F}$ has a unique minimizer given by

$$u_a = \sum_{b=1}^{N}D_{ab}v_b, \quad D_{ab} = -\sum_{c=1}^{n}A_{ac}^{-1}B_{cb}. \tag{17}$$

Combining (17) and (8) leads to the desired linear relation between nodal function values and the corresponding cell gradients:

$$g_{ir} = \sum_{a=1}^{N}L_{ia}^r v_a, \quad L_{ia}^r = k_{ia}^r + \sum_{b=1}^{n}m_{ib}^r D_{ba}. \tag{18}$$

For fixed index a, L_{ia}^r gives the gradient for cell r of the finite element shape function associated with node a. These are precisely the quantities needed for evaluation of finite-element residual and stiffness contributions in solid mechanics applications.

2.3. Shape-function values

With the shape-function gradients for each quadrature cell in hand, the final step in the PEM element formulation is to set the constant terms for each shape function, within each cell. To this end, we associate, with a vector $\mathbf{v}$ of nodal values, the discontinuous, piecewise-linear function $\varphi_r(\mathbf{x}), \mathbf{x}\in\Omega$, given by

$$\varphi_r = c_r + \mathbf{g}_r\cdot(\mathbf{x}-\mathbf{a}_r), \quad \mathbf{x}\in\omega_r, \tag{19}$$

where the cell gradients $\mathbf{g}_r$ are given by (18), and where $\mathbf{a}_r$ is a convenient but arbitrary fixed location associated with ω_r (e.g. a node, or the cell's centroid). The discrete values c_r are chosen to minimize the decoupled functionals

$$\mathfrak{H}_r = \int_{\partial\omega_r}(\varphi_r^- - \varphi)^2 ds, \tag{20}$$

where the superscript $()^-$ is meant as a reminder that (19) is to be evaluated on the inside of the relevant ω_r. In (20), φ is the piecewise-linear and continuous function defined on the segment network.

Each $\mathfrak{H}_r$ is a function of c_r only; consequently its minimization leads, with the help of (8), (5) and (7), to the explicit expression

$$c_r = \sum_{a=1}^{N}(P_a^r - \mathbf{y}_r\cdot\mathbf{k}_a^r)v_a + \sum_{a=1}^{n}(R_a^r - \mathbf{y}_r\cdot\mathbf{m}_a^r)u_a, \tag{21}$$

where

$$|\partial\omega_r|P_a^r = \sum_{\alpha\in Q_r}P_{\alpha a} + \sum_{\alpha\in\overline{Q}_r}\overline{P}_{\alpha a}, \quad |\partial\omega_r|R_a^r = \sum_{\alpha\in Q_r}R_{\alpha a}, \tag{22}$$

and where

$$|\partial\omega_r|\mathbf{y}_r = \int_{\partial\omega_r}(\mathbf{x}-\mathbf{a}_r)ds. \tag{23}$$

Finally, using (18) in (21) leads to the linear mapping

$$c_r = \sum_{a=1}^{N}K_a^r v_a, \quad K_a^r = P_a^r - \mathbf{y}_r\cdot\mathbf{k}_a^r + \sum_{b=1}^{n}(R_b^r - \mathbf{y}_r\cdot\mathbf{m}_b^r)D_{ba}. \tag{24}$$

This equation, together with (18), furnishes the constant and linear terms, respectively, for a piecewise-linear representation of the shape functions within each quadrature cell. Specifically, K_a^r and L_{ia}^r represent the constant and the gradient, respectively, of the shape function associated with node a, within cell r. These matrices depend exclusively on the geometry of the element, and on its quadrature-cell partition. A plot of representative piecewise-linear shape functions is shown in Fig. 3.

The above development pertains specifically to the 2D case. The analogous formulation for a 3D PEM element is entirely similar. In strict analogy with the 2D case, 3D PEM elements are first partitioned into polyhedral quadrature cells. The partition's vertices are grouped into four sets (instead of three as in the 2D case): nodes, non-node edge vertices, non-node facet vertices, and interior vertices. The main technical issue in the 3D case is computation of the integrals (5), where γ_α and $\bar{\gamma}_\alpha$ are now polygonal facets of the quadrature cells, instead of line segments. Again consistent with the 2D case, the interpolating function φ is taken to be autonomous on each cell facet, in the sense that φ on a given facet depends only on the values at the vertices of that facet. A formula, whose detailed derivation is to be given elsewhere, is then invoked to estimate the zero-order moments (5), on polygons, of harmonic functions whose boundary values are known. This additional step aside, the 3D element formulation is very close to the 2D one, with the final results also expressible in the forms (18) and (24).

2.4. Linear completeness

We have in mind application to second-order elliptic variational BVPs, such as those found in solid mechanics. In that context, a necessary condition for convergence is that the sequence of approximation spaces exhibits an approximability property with respect to the full (H_0^1) solution space. This means, in essence, that an arbitrary member of H_0^1 can be approximated arbitrarily closely, under suitable norm, by a member of an approximation space that is associated with a sufficiently refined mesh. However, the PEM generates approximation spaces that are not strictly contained in H_0^1 (i.e. the method is non-conforming), hence the discrete variational problem statement is not simply the continuous one, restricted to a subspace of H_0^1. An additional consistency condition is therefore relevant, which holds that the discrete problem itself must approach the continuous problem. Consistency is expressed by the quantity Δ in [21], and by the term $\|e_\iota\|_{E\prime_\iota}^2$ in the two-sided error estimate (12.i) in [31]. In this article we defer the consistency issue and focus on approximability, while relying on numerical experimentation (Section 4) to establish the likelihood of convergent behavior.

We take the standard approach to approximability by showing that the PEM element formulation exhibits linear completeness, which in the present notation can be stated as

$$v_a = c + \mathbf{b}\cdot\bar{\mathbf{x}}_a \Rightarrow c_r = c, \quad \mathbf{g}_r = \mathbf{b} \quad \forall c, \mathbf{b}, \tag{25}$$

where $\bar{\mathbf{x}}_a$ is the position of node p_a. Eq. (25) says that, when the nodal values are set consistent with a globally linear function, the shape-function formulation of Sections 2.2 and 2.3, interpreted as a generator of general interpolation functions, reproduces the same globally linear function.

We proceed by showing that, under the supposition $v_a = c + \mathbf{b}\cdot\bar{\mathbf{x}}_a$ (i.e. the left-hand element of Eq. 25), each term of (12) individually attains its absolute minimum value when inter-vertex values are consistent with the same globally linear variation, i.e. when

$$u_a = c + \mathbf{b}\cdot\mathbf{x}_a, \quad a = 1,\ldots,n, \tag{26}$$

where $\mathbf{x}_a$ is the position of interior vertex q_a. Indeed, under (26), the second term of (12) attains its absolute minimum value of zero, by inspection. Thus it remains only to show that, when u_a is given by (26) and v_a by the left-hand element of (25), the first term in (12), or, identically, (11), attains its absolute minimum value. This can be shown with the help of the following elementary lemma.

Lemma. *Let* $\mathbf{v}$, with components $v_i \in L_2(\Omega)$, be a vector-valued function defined on the bounded domain Ω, and let $\{\omega_1,\ldots,\omega_K\}$ be a partition of Ω, i.e. $\bigcup_{r=1}^{K}\omega_r = \Omega$ and $\omega_r \cap \omega_s = \emptyset,\ r \neq s.$

Define

$$\bar{\mathbf{v}} = \frac{1}{|\Omega|}\int_\Omega \mathbf{v}\,dv, \quad \mathbf{v}_r = \frac{1}{|\omega_r|}\int_{\omega_r}\mathbf{v}\,dv. \tag{27}$$

Then,

$$\int_\Omega \mathbf{v}\cdot\mathbf{v}dv \geqslant \sum_{r=1}^{K}|\omega_r|\mathbf{v}_r\cdot\mathbf{v}_r \geqslant |\Omega|\bar{\mathbf{v}}\cdot\bar{\mathbf{v}}. \tag{28}$$

Proof. Consider the case of a binary partition of Ω, i.e. $K = 2$. On ω_r, any $\mathbf{v}$ can be written as $\mathbf{v} = \mathbf{v}_r + \mathbf{a}$, where $\int_{\omega_r}\mathbf{a}dv = \mathbf{0}$. Then,

$$\int_{\omega_r}\mathbf{v}\cdot\mathbf{v}dv = |\omega_r|\mathbf{v}_r\cdot\mathbf{v}_r + \int_{\omega_r}\mathbf{a}\cdot\mathbf{a}dv \geqslant |\omega_r|\mathbf{v}_r\cdot\mathbf{v}_r. \tag{29}$$

Summing this over $r = 1, 2$ establishes the left-hand inequality of (28) for the case $K = 2$. The general case follows by induction, upon recursively subdividing the subdomains ω_r.

Returning to the binary-subdivision case and dropping the magnitude bars $||$ for the remainder of the proof, the first of (27) can be written $\bar{\mathbf{v}} = (\omega_1\mathbf{v}_1 + \omega_2\mathbf{v}_2)/(\omega_1+\omega_2)$, from which one easily obtains

$$\Omega\bar{\mathbf{v}}\cdot\bar{\mathbf{v}} = \frac{\omega_1^2\mathbf{v}_1\cdot\mathbf{v}_1 + \omega_2^2\mathbf{v}_2\cdot\mathbf{v}_2 + 2\omega_1\omega_2\mathbf{v}_1\cdot\mathbf{v}_2}{\omega_1+\omega_2}. \tag{30}$$

Now let $\mathbf{v}_2 = s\mathbf{v}_1 + \mathbf{p}$, where $\mathbf{p}\cdot\mathbf{v}_1 = 0$ and $s \in \mathbb{R}$. Then,

$$\Omega\bar{\mathbf{v}}\cdot\bar{\mathbf{v}} = \mathbf{v}_1\cdot\mathbf{v}_1\left[\frac{\omega_1^2 + s^2\omega_2^2 + 2\omega_1\omega_2 s}{\omega_1+\omega_2}\right] + \mathbf{p}\cdot\mathbf{p}\frac{\omega_2^2}{\omega_1+\omega_2}. \tag{31}$$

On the other hand,

$$\sum_{r=1}^{2}\omega_r\mathbf{v}_r\cdot\mathbf{v}_r = \mathbf{v}_1\cdot\mathbf{v}_1(\omega_1 + s^2\omega_2) + \mathbf{p}\cdot\mathbf{p}\,\omega_2. \tag{32}$$

Obviously $\omega_2 \geqslant \omega_2^2/(\omega_1+\omega_2)\forall\omega_1,\omega_2 \geqslant 0$, so in order to establish the right-hand inequality of (28), it remains only to observe that

$$\omega_1 + s^2\omega_2 \geqslant \frac{\omega_1^2 + s^2\omega_2^2 + 2\omega_1\omega_2 s}{\omega_1+\omega_2} \quad \forall\omega_1,\omega_2 \geqslant 0 \text{ and } s \in \mathbb{R}. \tag{33}$$

This is verified by multiplying by $(\omega_1+\omega_2)$ and then noting that $1 + s^2 \geqslant 2s\forall s \in \mathbb{R}$. This establishes the right-hand inequality for the case $K = 2$, with the general case again following by induction. □

With the foregoing lemma in hand, let φ be such that $\varphi\rfloor_{\partial\Omega} = c + \mathbf{b}\cdot\mathbf{x}$, as mandated by the left-hand element of (25), and let $\mathbf{v} = \nabla\varphi$. Then, $\bar{\mathbf{v}} = \mathbf{b}$ by (27) and the divergence theorem. The above lemma therefore implies that

$$\int_\Omega \nabla\varphi\cdot\nabla\varphi dv \geqslant \sum_{r=1}^{K}|\omega_r|\mathbf{g}_r\cdot\mathbf{g}_r \geqslant |\Omega|\mathbf{b}\cdot\mathbf{b}. \tag{34}$$

This equation implies that the absolute minimum value of (11) is obtained by taking $\varphi = c + \mathbf{b}\cdot\mathbf{x}$ throughout Ω, which in turn implies (26). Under (26), the corresponding function φ is globally linear on the network of segments, and from (8) we have $\mathbf{g}_r = \mathbf{b}$. Further, under (26), minimization of each $\mathfrak{H}_r$ in (20) with respect to c_r results in $c_r = c$. The shape functions defined by (19), (18), (24) therefore exhibit linear completeness. In essence, whenever these shape functions are used to interpolate nodal data that is consistent with a globally linear function, the interpolant reproduces this linear function.

3. Algorithm

The element-level calculations in the PEM can be organized into the steps enumerated below. The input to the calculation consists of coordinates of all vertices, identification of the nodes, and connectivity of the element's cell complex. The complete set of element-level calculations consists of computation of the following quantities: the quadrature-point weights and locations, the shape-function gradients for each node and each quadrature cell, and the shape-function values (constant terms) for each node and each quadrature cell. These computations can be carried out as follows:

1. If the input element is non-convex, then subdivide the element into a near-minimal number of convex macro-cells.
2. Partition each macro-cell into quadrature cells, as described in Section 2.1 above. This results in the following data: coordinates of non-node boundary vertices $\bar{p}_a$ and interior vertices q_a, connectivity information for cell-boundary segments $\bar{\gamma}_\alpha$ and γ_α, cell-boundary membership information $\overline{Q}^r$ and Q^r, normal vectors $\bar{\mathbf{n}}_\alpha$ and $\mathbf{n}_\alpha$, and normal-direction indicators z^r_α.
3. Compute the quadrature cell areas $|\omega_r|$ and centroids $\mathbf{x}_r$ (Eq. 2). This requires that normal distances from the origin to the lines containing the segments be computed, along with integrals of x_i on each segment. If no spatially-varying problem data (e.g. a body force or graded material property) is present, then the locations of the quadrature points are not needed. However, segment integrals of x_i are needed in step 11 below, and so should be computed and temporarily stored anyway.
4. Compute the sparse matrices $P_{\alpha a}, R_{\alpha a}$, and $\overline{P}_{\alpha a}$ (see Eqs. 5 and 7) and the vectors $\mathbf{k}^r_a$ and $\mathbf{m}^r_a$ (9).
5. Set the unit tangents λ_α and $\bar{\lambda}_\alpha$ on γ_α and $\bar{\gamma}_\alpha$ respectively; then form $p_{\alpha a}, \bar{p}_{\alpha a}$, and $r_{\alpha a}$ (see Eq. 13).
6. Calculate the matrices $\overline{A}_{ab}$ and $\overline{B}_{ab}$. These matrices can be inferred from (15), in conjunction with (12), (14), and (16). Specifically, $\overline{A}_{ab}$ appears in the $u_a u_b$ terms on the right-hand side of (15), whereas $2\overline{B}_{ab}$ multiplies the $u_a v_b$ terms.
7. Form A_{ab} and B_{ab} (Eq. 16), and solve for D_{ab} as given by the second of (17).
8. Compute L^r_{ia} as given by the second of (18). This completes the cell values of the shape-function gradients. If the problem requires only shape-function gradients, then the element formulation is finished at his point. If on the other hand full shape-function information is needed, e.g. to form a consistent mass matrix or to accommodate a nonzero body force, then continue to the next step.
9. Perform the column sums indicated in (22) on the sparse matrices computed in step 3 above to form P^r_a and R^r_a.

10. Select a reference point $\mathbf{a}_r$ for each cell. This point is arbitrary, but should be stored for each cell. Convenient choices are the cell centroid, or the first vertex stored for each cell. Alternatively, it can even be the same point for all cells in an element; e.g. the first node for the element.
11. Compute vectors $\mathbf{y}_r$ as indicated in (23). This is essentially a matter of retrieving the values previously computed in step 3 above.
12. Form K_a^r as given in (24), which constitutes the constant term c_r in cell ω_r for shape function number a. The piecewise-linear shape functions are now complete; they take the form.

$$\varphi_a\rfloor_{\omega_r} = K_a^r + L_{ia}^r(x_i - a_{ia}). \tag{35}$$

It is worth noting that all of the matrix manipulations required in the foregoing procedure, specifically step 7, involve small matrices. For example, the symmetric matrix A_{ab} has dimension equal to the number of interior vertices q_a for the element, which for a convex element is the number of nodes. Thus, while the required operations may appear complicated, they involve only very modest computational effort. Further, it has been observed in computational experiments that the computations remain well-conditioned for essentially arbitrary element geometry and arrangements of nodes and quadrature cells. This holds even in the case of spatially coincident nodes, as would occur in an element that is partially penetrated by a zero-thickness crack. It is required, however, that the quadrature cells all have non-vanishing area.

4. Numerical exposition

In this section, the behavior of the PEM is explored via a number of computational experiments. Specifically, patch-test behavior is shown, and convergence in the energy norm is compared to the conventional FEM for two linear-elasticity problems. The section concludes with results for a finite-deformation elastic–plastic problem. See Fig. 4 for a pictorial depiction of the elasticity and elasto-plasticity problems detailed in Sections (4.2)–(4.5). In all problems involving traction boundary conditions, the domain was minimally restrained with displacement BCs to prevent rigid-body motion.

Comparisons are shown between PEM and conventional FEM calculations, where standard four-node bilinear quadrilateral elements with 2 × 2 Gaussian quadrature are used in the latter case. In both the PEM and FEM element formulations, a finite-deformation version of the “enhanced assumed strain” idea [33] is used, whereby the volume change at all integration points is replaced by an average value for the element as a whole. This formulation is free of volumetric locking in the presence of nearly incompressible material behavior.

The energy norm of the error was computed for all elastic test problems below. In all cases, an “exact” reference solution was generated using FEM/Q4, with meshes consisting of more than 5×10^4 elements. The error in the stresses was then computed at each integration point of the test mesh via a least-squares fit of the reference solution’s integration-point data, weighted by the integration weights. The test mesh’s integration rule was then used to form the energy norm of the error.

4.1. Patch tests

A variety of patch-test configurations is illustrated in Fig. 5. For each mesh, both traction- and displacement-boundary-condition cases were run. In this connection, it is worth mentioning that the nodal data in the PEM interpolates the displacement field on element boundaries. Hence, displacement boundary conditions are imposed in the same way they are in the conventional FEM – by simply specifying nodal values on the domain boundary, consistent with the boundary conditions. In all cases, the PEM solution produced nodal displacements and quadrature-cell stresses that are correct to the expected level of numeric precision, i.e. 12 digits of accuracy in double-precision computations, indicating that the method passes the engineering patch test for the meshes shown.

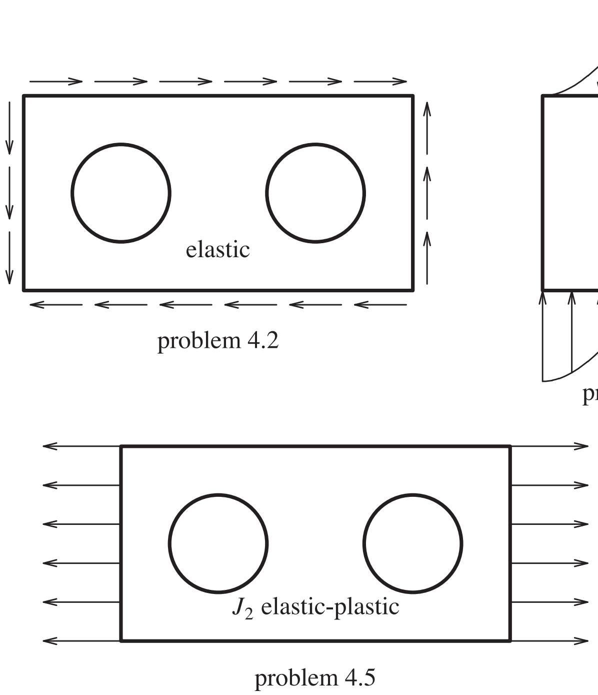

Fig. 4. Geometry and traction boundary conditions for all example problems. All problems were subject to plane-strain conditions. The figures are labeled with the subsection numbers in which the corresponding problems are presented.

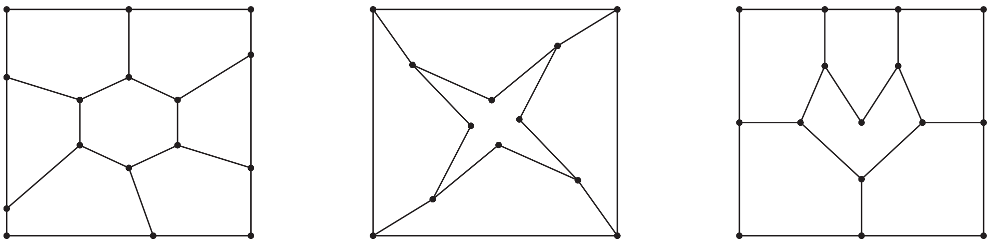

Fig. 5. Example PEM patch-test meshes.

4.2. Shear stress on a domain with two holes

In this linear elasticity problem, a rectangular domain with two circular holes was subjected to all-around shear traction consistent with the stress components $T_{12} = T_{21} = T,\ T_{11} = T_{22} = 0$. Two representative meshes, one consisting of conventional four-node quadrilateral finite elements, and the other of more general polygonal PEM elements, are illustrated in Fig. 6. The domain was subject to the plane-strain constraint, and was assigned the material properties $E = 3600 \times T,\ \nu = 0.3$.

The convergence performance of the PEM for this problem was compared to that for the FEM/Q4 via the energy norm of the error. The comparison is shown in Fig. 7, in which the natural logarithm of the energy norm of the error, normalized by the energy norm of the solution, is plotted vs. $\log(h)$. Here, h is the maximum element diameter in the mesh. The results for eight FEM/Q4 analyses are shown in the figure, corresponding to eight levels of uniform mesh refinement. Fig. 7 also shows the results for two sets of 64 PEM analyses. The first set was obtained using a centroidal quadrature cell in each element, with $\xi = 0.5$ (see Section 2.1). In the second set of PEM analyses, designated “PEM_NC” in the plot, no centroidal quadrature cell was used in the element formulation. Each PEM set of 64 analyses can be grouped into eight groups of eight analyses each, where the analyses within a group correspond to meshes with a similar level of mesh refinement.

The PEM meshes were generated using a recursive-subdivision strategy, as follows. Polygonal regions are divided into three subregions, unless the region has a large aspect ratio and is bounded by four or more segments, in which case it is cut into two subregions. Regions are subdivided based on a priority queue, in which the highest priority is accorded the region whose size differs from its target size by the largest amount. The target size can vary spatially, providing a means of grading the mesh. Subdivision then occurs by connecting the region’s centroid to vertices and/or segment midsides with up to three new segments. This procedure is applied to the mesh recursively, until all regions are at or below a specified tolerance of their target sizes. The subdivision procedure supports the introduction of some random variation; hence, any number of

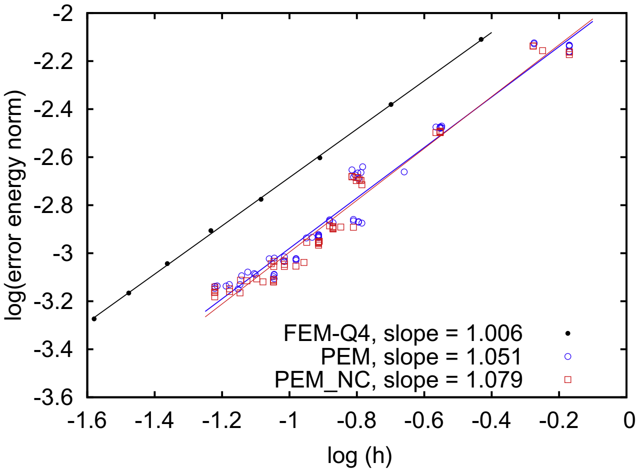

Fig. 7. Energy norm of the error vs. element diameter in log–log format, for the two-hole elastic domain loaded in shear. The results for a conventional FE analysis using four-node quadrilateral elements are compared to those of the PEM, both with (PEM) and without (PEM_NC) a centroidal quadrature cell.

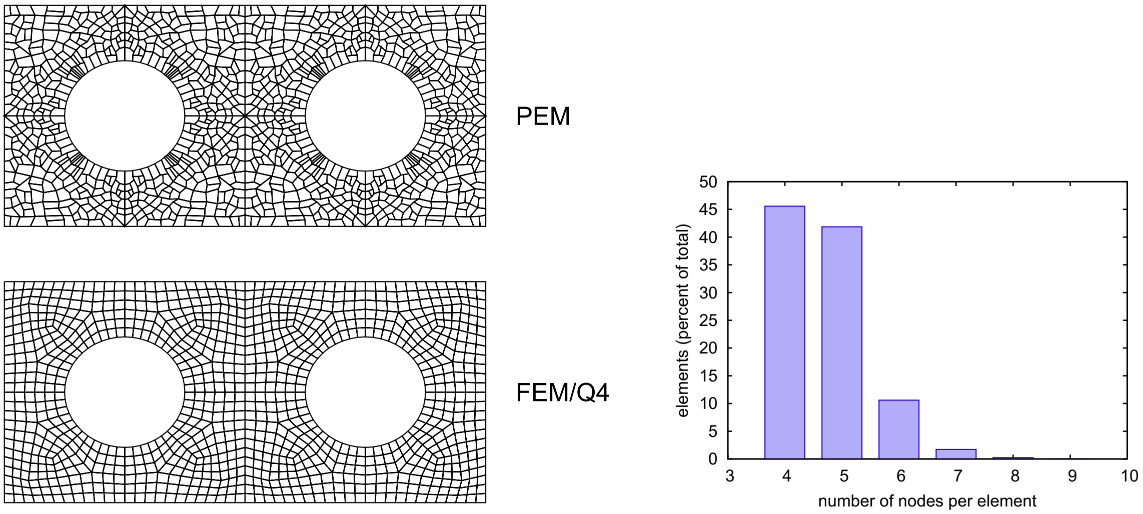

Fig. 6. Example PEM and FEM/Q4 meshes for the two-hole domain, at similar levels of mesh refinement. The distribution of numbers of nodes per element in a typical PEM mesh used for this problem is shown at right.

meshes can be generated automatically at a given approximate level of refinement.

The recursive-subdivision polygonal mesher produces quite satisfactory PEM meshes on any 2D domain that has been subdivided into a small number of convex, or nearly convex, regions. It produces meshes that contain a diversity of (not necessarily convex) elements, as illustrated by the representative histogram in Fig. 6. In the PEM mesh in Fig. 6, some non-convex elements can be seen. The effects of PEM-element non-convexity are studied more systematically in Section 4.3.

The error plot (Fig. 7) suggests that, for this problem, the PEM achieves an accuracy comparable to, or even slightly better than, that of the FEM with conventional Q4 elements, at similar levels of mesh refinement. Interestingly, for this problem the PEM quadrature-cell design without the centroidal cell appears to perform as well as the design that uses the extra cell. Both the PEM and FEM/Q4 methods exhibit the theoretical $O(h)$ convergence rate in the energy norm. Although the PEM appears to do somewhat better in this regard, this observation may not be definitive, in light of the scatter evident in the error plot.

4.3. Non-convex elements

In an effort to isolate and study the effects of element non-convexity on the performance of the PEM, FEM/Q4 and PEM solutions were compared for a simple non-uniform compression problem on a rectangular domain (see Fig. 4). The traction distributions acting on the top and bottom edges are sinusoidal. Once again, the energy norm of the error was used to characterize solution accuracy. The Q4 meshes consisted of Cartesian $3n \times 2n$ arrays of square elements. The PEM meshes were built from interlocking pairs of 8-node L-shaped elements, as shown in Fig. 8. Each L-shaped PEM element was subdivided into 15 quadrature cells in a pattern corresponding to an array of three "standard" square PEM elements, which have five cells each. This was done to isolate, to the greatest extent possible, the influence of element non-convexity on solution accuracy via the shape-function formulation. Although the 8-node L-shaped elements inherited their quadrature rule from the square 4-node PEM element, this does not mean that the 8-node element amounts to a complex of three square elements: in the 8-node element, the shape-function gradient for each of the element's eight shape functions is nonzero at all 15 integration points. This is, of course, not the case in an assemblage of multiple elements.

The problem was analyzed under plane-strain conditions, again with $\nu = 0.3$. Fig. 9 shows the log of the (normalized) energy norm of the error vs $\log(h)$ for three scenarios: conventional Q4-FEM elements, square PEM elements with four nodes and five integration points each, and 8-node L-shaped PEM elements. For consistency, in the plot h is taken to be the edge length of the 4-node square in all three cases. It is noted that the nodal locations, and the total numbers of dof, are identical across the three scenarios for a particular value of h. Also, the three cases exhibit similar levels of integration-point density, with 24, 30, and 30 (respectively) integration points occurring in each 3×2 complex of squares. Fig. 9 shows nearly identical accuracy in the FEM/Q4 and 4-node PEM cases, while the 8-node L-shaped element results in a modest degradation in the energy norm of the error. In all three cases, the expected $O(h)$ convergence rate is evident. The results suggest that pervasive non-convexity in PEM elements, while perhaps not lending itself to optimal accuracy, does not lead to pathologies in the shape functions, as it does e.g. in conventional isoparametric element formulations.

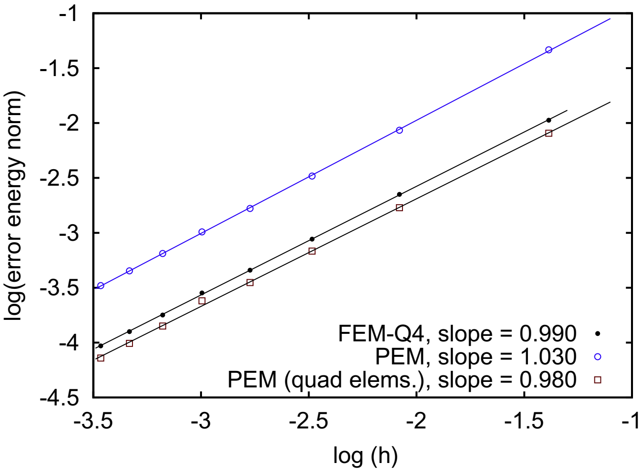

Fig. 9. Energy norm of the error vs. mesh parameter in log–log format, for the non-uniform compression problem illustrated in Fig. 4. Results are shown at eight levels of mesh refinement for three element formulations: conventional 4-node quad, a 4-node PEM quadrilateral with five integration points per element, and interlocking L-shaped 8-node PEM elements.

4.4. Near-degenerate nodal arrangements

In this calculation, a second type of exceptional feature in the element geometry is isolated, and its effects on the approximating

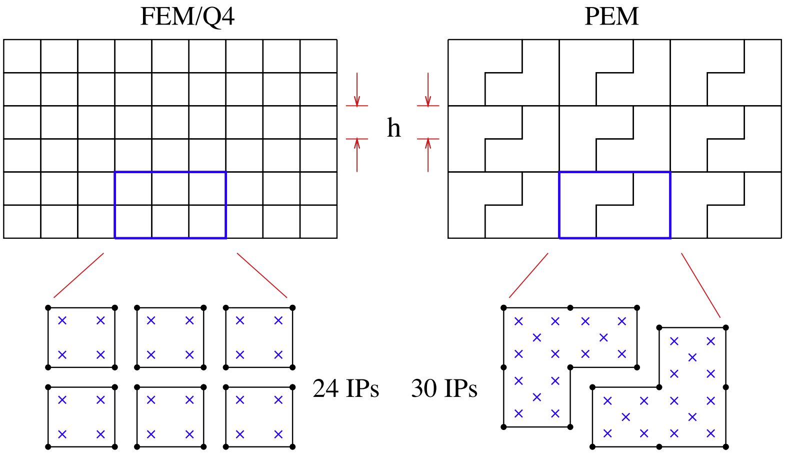

Fig. 8. Representative FEM/Q4 and PEM meshes used in the non-uniform-compression problem. The meshes shown have identical mesh parameter h. The arrangements of integration points for a 3×2 complex of squares is illustrated for both meshes.

power of the PEM are studied. As with the previous problem, the motivation here stems mainly from the reasoning that a polygonal finite element method is most useful if it can tolerate element shapes that are, in one way or another, pathological or exceptional, without suffering undue degradation in solution accuracy. In the present case, we consider PEM meshes consisting of square elements, but where many instances of side nodes occur. These side nodes are, in many cases, placed extremely close to a corner node.

The specifics of this scheme can be explained with reference to Fig. 10. The figure shows an instance of a PEM mesh used to solve the same problem as in the previous subsection, namely, the non-uniform-compression problem of Fig. 4. Dots are used to indicate the locations of the additional (i.e. element-side) nodes; there are nodes at each element corner as well, but these are not illustrated with dots for clarity. In placing the element-side nodes, 50% of all inter-element sides were selected at random. For a selected side with endpoints $\mathbf{x}_a$ and $\mathbf{x}_b$, the extra node was placed at $(1-r^8)\mathbf{x}_a + r^8\mathbf{x}_b$, where r is a random number in the range $(0,1)$. In cases where r^8 lies outside the range $(0.001, 0.999)$, the r value is discarded and another random number is generated. The $()^8$ has the effect of strongly biasing the locations of the element-side nodes toward one endpoint or the other, as can be seen in Fig. 10. Owing to the randomness in both the element-side selection and in the node placement on each selected side, an arbitrary number of instances of meshes of this kind can be generated automatically. This scheme results in meshes with a diversity of elements, where each element has at least four, and no more than eight, nodes in a random arrangement – see the histogram accompanying the mesh in Fig. 10. Finally, we note that the integration rule for each element is as described in Section 2.1. Two cases were considered, namely, a centroidal quadrature cell with $\xi = 0.5$, and no centroidal cell.

Fig. 11 shows a comparison of the error decay, in log–log format, for FEM/Q4 analyses vs. PEM analyses using the above-described mesh scheme. Eight values of h were used, with one FEM/Q4 and 16 random PEM meshes (eight for the centroidal-cell and eight for the no-centroidal-cell cases) used at each level of mesh refinement. The comparison shows nearly identical levels of accuracy across all the analyses. An additional set of eight PEM analyses was performed in which element-side nodes were placed at a randomly-selected 50% of the inter-element sides, but with the nodes placed at the middle of the sides in all cases, instead of being biased toward an endpoint. Error results for these analyses are shown in Fig. 11 as well, from which it can be concluded that near-degenerate nodal locations have little influence on the accuracy of the PEM, at least for this problem.

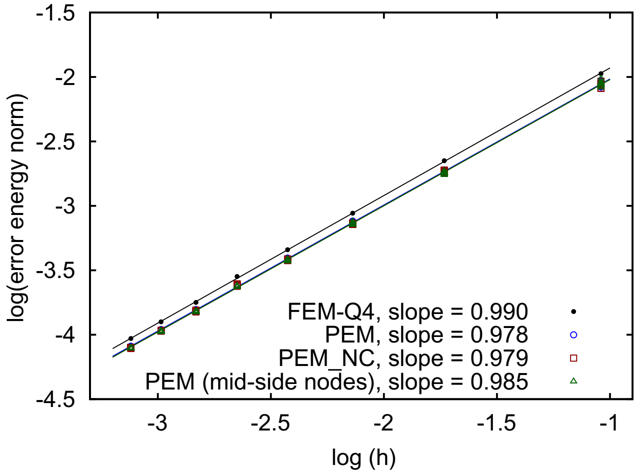

Fig. 11. Energy norm of the error vs. element diameter in log–log format, for the rectangular elastic domain loaded in non-uniform compression as shown in Fig. 4. The results for a conventional FE analysis using four-node quadrilateral elements are compared to those of the PEM, where side nodes have been added at random to the PEM elements.

The foregoing results suggest that the presence of element-side nodes does little to enhance accuracy, at least under the "generic" integration rule outlined in Section 2.1. This finding can be explained by recalling that the shape-function gradients are approximated by constants within each integration cell. In the case of an element-side node, the associated "continuous" shape function clearly has a gradient that varies markedly in the neighborhood of the node. Therefore, in such situations it may be advisable to replace the integration-cell design of Section 2.1 with one where a cell boundary passes through each element-side node, and indeed through each node at which the included angle is greater than, say, 135°.

4.5. Finite-deformation elastic–plastic problem

In this final demonstration problem, we return to the two-hole geometry of Section 4.2, but with an elastic–plastic material, and with the shear-stress boundary condition replaced by horizontal tension applied on the left and right sides (see Fig. 4). The constitutive model is J_2 flow theory of elasto-plasticity, with isotropic hardening given by the uniaxial stress–strain law

$$\sigma = \sigma_0\left[1 + (\epsilon/\epsilon_0)^n\right], \tag{36}$$

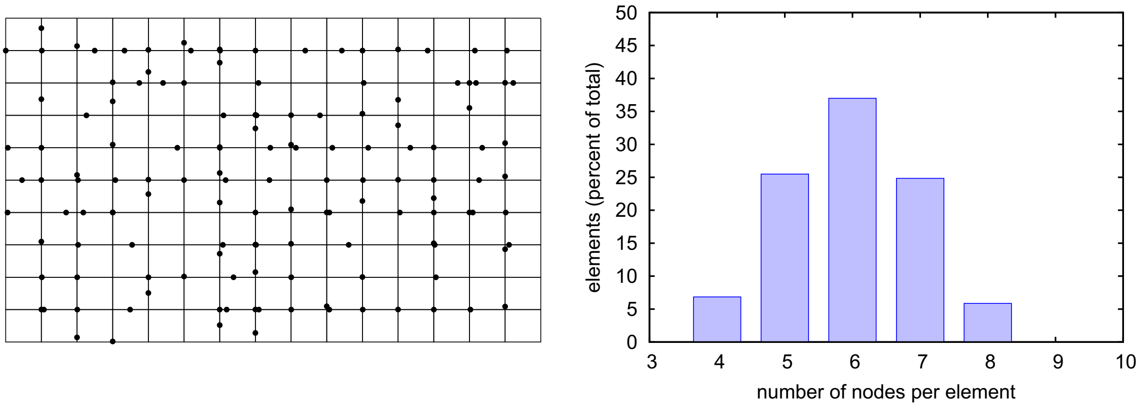

Fig. 10. Representative PEM mesh of a rectangular domain, where side nodes have been randomly placed at 50% of the interior element edges. The side nodes are indicated with dots. Nodes also occur at the quadrilateral vertices. A histogram showing the numbers of nodes per element for a typical mesh of this type is shown at right.

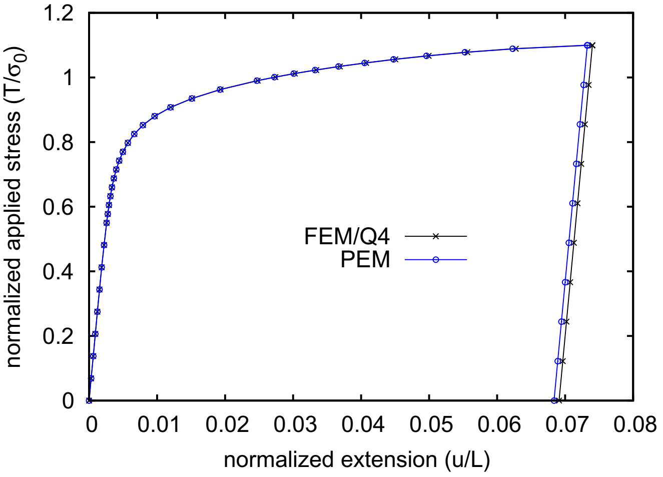

Fig. 12. Applied tension (normalized by the initial yield stress) vs. normalized centerline displacement for the two-hole elastic–plastic analysis. Results for the conventional FEM/Q4 and PEM analyses are shown.

where σ is the true stress in a uniaxial stress–strain test, ϵ is the equivalent plastic strain, and σ_0, ϵ_0, and n are material constants with values 140 MPa, 0.1, and 0.2, respectively. The elastic response is characterized by an isotropic hypoelastic relation, with $E = 72$ GPa and $\nu = 0.3$.

This problem was analyzed with the conventional FE method using 4032 Q4 elements, and the PEM with 4768 polygonal elements having between 4 and 7 nodes per element (see Fig. 13). The PEM mesh was generated using the same recursive-subdivision method as was used in problem (4.2) above. Element diameters fell in the range (0.1926 m, 0.3381 m) for the FEM/Q4 mesh, and (0.1247 m, 0.3844 m) for the PEM mesh. Volumetric locking was avoided through the use of an enhanced-assumed-strain-type formulation similar to that presented in [33].

The applied stress was increased monotonically in 34 load steps, up to a maximum of 1.1 times the material's initial yield stress (σ_0). This resulted in an axial displacement along the domain's symmetry line of about 0.074 times the initial length. The applied stress was then unloaded to zero in 10 additional load steps, leaving a residual state of stress and deformation. This loading program is shown in Fig. 12, where the applied stress vs. axial deformation is compared for the FEM/Q4 and PEM cases. Fig. 13 shows both the FEM and PEM meshes in their undeformed and deformed configurations, the latter following the unloading. Fig. 14 compares the final (after unloading) distributions of equivalent plastic strain for the FEM (above the symmetry line) and PEM (below) analyses. In this figure, 14 identical contour levels were used to facilitate the comparison. These results indicate a very close match between the two solution methods for this problem.

5. Concluding remarks

The principal attraction of the PEM is that its element formulation allows for considerable flexibility in the element geometry and connectivity, and thus in mesh generation. The PEM element formulation consists of two main elements: a partition of the element into quadrature cells, and the calculation of shape-function gradients and values for each cell. A distinguishing feature of the PEM is that the shape functions are defined only by discrete data, rather than as analytic functions over the whole of the element, as they are in the traditional FEM. As such, the shape functions can be visualized as piecewise-linear, discontinuous functions over the element's assemblage of quadrature cells. The approximating power of the shape functions depends on the quadrature-cell partition; here, we have associated one cell with each node, with an additional cell at the element's centroid.

The numerical results presented herein suggest that $O(h)$ error in the stresses can be expected of the PEM, as with conventional quasi-linear elements (quad-4 in 2D and hex-8 in 3D). For a fixed mesh, the error magnitude depends on the number and arrange-

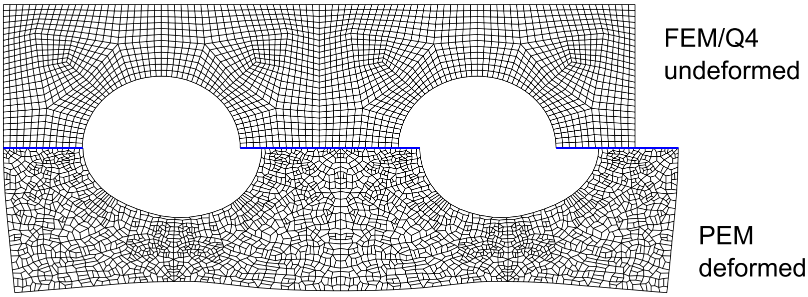

Fig. 13. FEM mesh with four-node quadrilateral elements (top) and PEM mesh generated using recursive subdivision, as used in the two-hole elastic–plastic analysis. The PEM mesh is shown in its deformed configuration, following unloading.

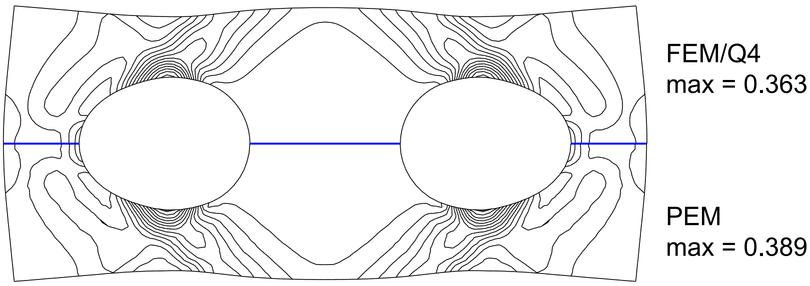

Fig. 14. Contour plot of the equivalent plastic strain following unloading. The FEM/Q4 result is shown above the symmetry line, and the PEM result below. Identical contour levels are used in both cases.

ment of the quadrature cells within each element, again as might be expected. It might be said that the PEM functions to replace the burden of generating a conventional FE mesh with the task of designing high-quality partitions of each element into quadrature cells. However, the latter job is decidedly simpler: the topological and geometric constraints attending the quadrature-cell design are few and mild compared to those of FE mesh generation. For example, a facet shared by two adjacent elements need not have partitions that coincide.

A simple, easy-to-implement quadrature-cell design is presented here, which appears to render overall performance that is competitive with well-designed conventional FE meshes at similar levels of mesh density. Elements that are severely non-convex may require subdivision into macro-cells before applying the cell subdivision scheme. In three dimensions, Voronoi cells with the nodes and the element centroid as nuclei appear to hold promise as quadrature cells. In all cases, the main objective is to choose the cell partitioning so that smooth, interpolating shape functions, which are never specified analytically in the PEM, are well-approximated by piecewise-linear fields over the cells. The minimum value of $\mathfrak{F}$ (see Eq. 12) could, in assessing alternative cell designs, serve as an objective measure of the extent to which this holds. The two partitioning schemes considered here – node-centered, with and without a centroidal cell – appear to give similar levels of solution accuracy, at least for the problems considered here. An ideal partitioning scheme is the subject of ongoing research.

The computational effort associated with the PEM element formulation itself can be many times that of the conventional FEM at similar levels of mesh refinement. Indeed, the element-level computations in the traditional isoparametric formulation are extremely simple; the traditional FEM's economy at the element level is difficult to approach using any alternative method. However, in all elastic demonstration problems presented herein, the PEM element-level computations still consumed only 1–3% of the total run time for the problem, depending on the problem size (with larger percentages associated with smaller problems). This percentage decreases as the problem size increases, and is much smaller in the case of nonlinear problems, where much more extensive computations occur with a fixed set of shape-function data.

Acknowledgement

The support of the US National Science Foundation through Grant No. CMMI-0726180 to the University of California, Davis is gratefully acknowledged.

References

[1] J. Yoo, B. Moran, J. Chen, Stabilized conforming nodal integration in the natural-element method, Int. J. Numer. Methods Engrg. 60 (2004) 861–890.
[2] J. Chen, C. Wu, S. Yoon, Y. You, A stabilized conforming nodal integration for galerkin mesh-free methods, Int. J. Numer. Methods Engrg. 50 (2001) 435–466.
[3] W. Quak, A. van den Boogaard, D. Gonzalez, E. Cueto, A comparative study on the performance of meshless approximations and their integration, Comput. Mech. 48 (2011) 121–137.
[4] Q. Zhang, Theoretical analysis of numerical integration in galerkin meshless methods, BIT Numer. Math. 51 (2011) 459–480.
[5] N. Sukumar, A. Tabarraei, Conforming polygonal finite elements, Int. J. Numer. Meth. Engrg. 61 (2004) 2045–2066.
[6] S. Idelsohn, E. Onate, B. Calvo, F. del Pin, The meshless finite element method, Int. J. Numer. Meth. Engrg. 58 (2003) 893–912.
[7] J. Bishop, Simulating the pervasive fracture of materials and structures using randomly close packed voronoi tessellations, Comput. Mech. 44 (2009) 455–471.
[8] M. Wicke, M. Botsch, M. Gross, A finite element method on convex polyhedra, Eurographics 26 (2007) 355–364.
[9] C. Talischi, G. Paulino, A. Pereira, I. Menezes, Polygonal finite elements for topology optimization: a unifying paradigm, Int. J. Numer. Methods Engrg. 82 (2010) 671–698.
[10] J. Chen, C.-J. Li, W.-J. Chen, Construction of n-sided polygonal spline element using area coordinates and b-net method, Acta Mech. Sin. 26 (2010) 685–693.
[11] S. Weisser, Residual error estimate for bem-based fem on polygonal meshes, Numer. Math. 118 (2011) 765–788.
[12] C. Hofreither, U. Langer, C. Pechstein, Analysis of a non-standard finite element method based on boundary integral operators, Electronic Trans. Numer. Anal. 37 (2010) 413–436.
[13] S. Martin, P. Kaufmann, M. Botsch, M. Wicke, M. Gross, Polyhedral finite elements using harmonic basis functions, Eurographics 27 (2008) 1521–1529.
[14] S. Mousavi, H. Xiao, N. Sukumar, Generalized gaussian quadrature rules on arbitrary polygons, Int. J. Numer. Methods Engrg. 82 (2010) 99–113.
[15] S. Mousavi, N. Sukumar, Numerical integration of polynomials and discontinuous functions on irregular convex polygons and polyhedrons, Comput. Mech. 47 (2011) 535–554.
[16] J. Lasserre, Integration on a convex polytope, Proc. Amer. Math. Soc. 126 (1998) 2433–2441.
[17] S. Natarajan, S. Bordas, D. Mahapatra, numerical integration over arbitrary polygonal domains based on Schwarz–Christoffel conformal mapping, Int. J. Numer. Methods Engrg. 80 (2009) 103–134.
[18] M. Rashid, P. Gullet, On a finite element method with variable element topology, Comp. Methods Appl. Mech. Engrg. 190 (2000) 1509–1527.
[19] M. Rashid, M. Selimotic, A three-dimensional finite element method with arbitrary polyhedral elements, Int. J. Numer. Methods Engrg. 67 (2006) 226–252.
[20] Z. Shi, The F-E-M test for convergence of nonconforming finite element, Math. Comp. 49 (1987) 391–405.
[21] G. Strang, G. Fix, An Analysis of the Finite Element Method, Prentice-Hall, Englewood Cliffs, NJ, 1973.
[22] T. Strouboulis, K. Copps, I. Babuska, The generalized finite element method, Comp. Methods Appl. Mech. Engrg. 190 (2001) 4081–4193.
[23] T. Belytschko, C. Parimi, N. Moes, N. Sukumar, S. Usui, Structured extended finite element methods for solids defined by implicit surfaces, Int. J. Numer. Methods Engrg. 56 (2003) 609–635.
[24] M. Rashid, The arbitrary local mesh replacement method: an alternative to remeshing for crack propagation analysis, Comp. Methods Appl. Mech. Engrg. 154 (1998) 133–150.
[25] A. Kumar, S. Padmanabhan, B. Ravi, Implicit boundary method for finite element analysis using non-conforming mesh or grid, Int. J. Numer. Methods Engrg. 74 (2007) 1421–1447.
[26] G. Liu, K. Dai, T. Nguyen, A smoothed finite element method for mechanics problems, Comput. Mech. 39 (2007) 859–877.
[27] K. Dai, G. Liu, T. Nguyen, An n-sided polygonal smoothed finite element method (nsfem) for solid mechanics, Finite Elements Anal. Des. 43 (2007) 847–860.
[28] G. Liu, T. Nguyen-Thoi, H. Nguyen-Xuan, K. Lam, A node-based smoothed finite element method for upper-bound solution to solid problems (ns-fem), Comp. Struct. 87 (2009) 14–26.
[29] T. Nguyen-Thoi, G. Liu, H. Nguyen-Xuan, An n-sided polygonal edge-based smoothed finite element method (nes-fem) for solid mechanics, Int. J. Numer. Methods Biomed. Engrg. 27 (2011) 1446–1472.
[30] D. Wang, J.-S. Chen, A Hermite reproducing kernel approximation for thin-plate analysis with sub-domain stabilized conforming integration, Int. J. Numer. Methods Engrg. 74 (2008) 368–390.
[31] F. Stummel, The generalized patch test, SIAM J. Numer. Anal. 16 (1979) 449–471.
[32] S. Sobolev, V. Vaskevich, The Theory of Cubature Formulas, Kluwer, Dordrecht; Boston, 1997.
[33] E. de Souza Neto, D. Peric, M. Dutko, D. Owen, Design of low order finite elements for large strain analysis of nearly incompressible solids, Int. J. Solids Stuct. 33 (1996) 3277–3296.