Contents lists available at SciVerse ScienceDirect



Applied Mathematics and Computation

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

Mesh based construction of flat-top partition of unity functions



Won-Tak Hong^a, Phill-Seung Lee^{a,b,*}

^a KAIST Institute for Design of Complex Systems, 291 Daehak-ro, Yuseong-gu, Daejeon 305-701, Republic of Korea ^b KAIST Ocean Systems Engineering, 291 Daehak-ro, Yuseong-gu, Daejeon 305-701, Republic of Korea

ARTICLE INFO

Keywords: Partition of unity Flat-top Generalized finite element method (GFEM)

ABSTRACT

A novel idea to construct flat-top partition of unity functions in a closed form on a general (structured or unstructured) finite element mesh is presented. An efficient and practical construction method of a flat-top partition of unity function is important in the generalized finite element method (GFEM). Details on how to construct flat-top partition of unity functions on a provided mesh are given. The generalized finite element approximation with the use of the new flat-top partition of unity function is presented with various numerical examples that demonstrate the effectiveness of proposed flat-top partition of unity functions.

© 2013 Elsevier Inc. All rights reserved.

1. Introduction

Partition of unity function is an essential component of the generalized finite element method (GFEM) [1,2], and in a number of meshless methods [3–6]. These methods, based on the partition of unity, have been successfully applied to solve significant engineering problems (see [1–7], and references therein). In the meshfree (or meshless) community, Shepard type partition of unity functions and their variants have been popular owing to the flexibility of allowing non-polygonal support and the ability to control the smoothness of the approximation functions, see for example [7]. In general, non-polygonal support and smoothness of the approximation function incur a high computational cost to achieve accurate numerical integrations. As a result, several researchers have started to use conventional linear hat functions in this way, the smoothness of approximation is lost. A singular or nearly singular stiffness matrix can also result from the use of linear polynomials as enrichment functions [9,8,10]. Nevertheless, using the conventional hat function was shown to be successful in the application of the partition of unity enrichment technique in the context of the generalized finite element method (GFEM) [11–18]. The key to the success is the enhanced degree of practicality; that is, the method can very effectively construct the partition of unity functions.

A simple solution, creating a flat-top region in the partition of unity function was proposed to avoid an ill-conditioned system with the partition of unity enrichment [8–10,19–21]. It was shown to be effective to reduce the matrix condition number with higher order polynomial enrichments. Recently, Oh et al. [21] introduced a general framework to create flat-top partition of unity in 2D and 3D on a polygonal domain. The powerful aspect of the general framework is that it utilizes closed-form one-dimensional partition of unity function to create partition of unity function via a simple product. The majority of the efforts are focused to improve meshless (meshfree) partition of unity method. Although it may be possible, it is not a straightforward to create flat-top partition of unity functions on a (structured or unstructured) finite element mesh. This fact motivates us to develop an efficient and practical method to construct flat-top partition of unity functions.

0096-3003/\$ - see front matter @ 2013 Elsevier Inc. All rights reserved. http://dx.doi.org/10.1016/j.amc.2013.02.055

^{*} Corresponding author at: KAIST Institute for Design of Complex Systems, 291 Daehak-ro, Yuseong-gu, Daejeon 305-701, Republic of Korea. *E-mail address:* phillseung@kaist.edu (P.-S. Lee).

In this paper, we present a novel algorithm that constructs flat-top partition of unity functions on a unstructured finite element mesh. We list some of the important features that distinguish the proposed partition of unity construction method from existing methods:

- A locally adjustable flat-top size: The existence of the flat-top region and the size of the flat-top area in the formulation of the partition of unity function are important when patch-wise (elemental) higher-order polynomial enrichment is utilized. The ability to adjust the size of the flat-top region in a patch-wise manner is not an issue when the enrichment order is chosen uniformly over the entire computational domain, as in several earlier studies [20,9,21]. However, a locally adjustable flat-top size becomes important when a different order of polynomial enrichment is used. We will demonstrate this effect in the following section.
- Piecewise linear polynomial partition of unity function in a closed form: We build the flat-top partition of unity function on a unstructured mesh with standard finite element shape functions. With the help of mesh, the support of partition of unity functions can be identified easily. Also, unlike the popular Shepard partition of unity, which is defined as rational functions, the proposed partition of unity is a piecewise linear polynomial including flat-top. This significantly reduces the amount of work for numerical integration.

In the conventional finite element framework, elemental shape functions are defined on a reference coordinate system and mapped to a physical coordinate system to create approximation functions. The mapped shape functions need to be assembled at common vertices along the common edge/face of neighboring elements to obtain continuity of the basis functions. Hence, more vertices may appear along the element edge/face when the order of the interpolation function is increased on the element. As a result, the neighboring elements needs to be altered or a special treatment becomes necessary to maintain continuity across the elements in the conventional approach. On the other hand, the basis functions for the partition of unity method, automatically satisfy the continuity requirement as long as the local enrichment functions are continuous. Therefore, with the proposed partition of unity function, we can fully control the mesh size *h* as well as the patch-wise enrichment order *p* on a conventional finite element mesh.

2. Partition of unity functions

In the partition of unity method [19], one creates a collection of open covering of the given domain Ω and constructs a partition of unity function ϕ_i subordinated to each cover. A formal requirement for ϕ_i to be a partition of unity is given below.

Definition 1. { $\phi_i : i \in \Lambda$ } is called a **partition of unity** subordinate to the covering { $Q_i : i \in \Lambda$ } if there is a family { $\phi_i : i \in \Lambda$ } of Lipschitz functions in Ω that satisfy the following three conditions:

(i) $\exists C$ such that $\|\phi_i\|_{\infty,\mathbb{R}^d} \leq C$ for all *i*.

- (ii) $\operatorname{supp}(\phi_i) \subseteq \overline{Q_i}$, for each $i \in \Lambda$.
- (iii) $\sum_{i \in \Lambda} \phi_i(x) = 1$, $\forall x \in \Omega$.

where $\{Q_i : i \in \Lambda\}$ is a point finite open covering of a domain Ω , and Λ is an index set.

The open covering is sometimes called *clouds* [11,18], *spheres* [7] or *patches* [9,20,21]. In this paper we will adopt the notion of *patches*.

A well-known and popular partition of unity function in meshless community is the Shepard partition of unity function [22]. The Shepard partition of unity function needs an open covering of the given domain. However, the shape of the covering does not have to be polygon and can be overlapped many times. Thus has less restriction on geometric constraints compared to conventional finite element methods. Hence, suitable for meshless approximation and has been widely used in many partition of unity methods. The Shepard partition of unity is defined as follows:

Definition 2. Let $W_i : \mathbb{R} \to \mathbb{R}$, $W_i \in C^k$, $k \ge 0$ denote a bubble function with compact support ω_i . Suppose we have a open covering $\{Q_i\}$ of a domain Ω in \mathbb{R}^n and bubble function is built at every ω_i , i = 1, ..., N of an open covering $\{Q_i\}$ of a domain Ω . Then the Shepard partition of unity subordinate to the covering $\{Q_i\}$ is defined as follows:

$$\phi_i(\mathbf{x}) = \frac{W_i(\mathbf{x})}{\sum_{\beta(\mathbf{x})} W_\beta(\mathbf{x})} \quad \beta(\mathbf{x}) \in \{\gamma | W_\gamma(\mathbf{x}) \neq \mathbf{0}\} \quad i = 1, \dots, N.$$
(1)

However, the versatility of the Shepard partition of unity function comes at the price. The difficulty in using the Shepard function, Eq. (1), is the high computational cost to achieve accurate numerical integration.

Definition 3. A partition of unity $\{\phi_i : i \in \Lambda\}$ subordinate to the covering $\{Q_i : i \in \Lambda\}$ that has non empty open set ω_i within the support of ϕ_i for each $i \in \Lambda$ such that $\phi_i(x) = 1$, $\forall x \in \omega_i$ is called a flat-top partition of unity.

The partition of unity function can only reproduce constant function exactly on the entire computational domain. Thus, local enrichment is necessary in most cases to have higher reproducing order. The flat-top partition of unity function has been known to be effective to reduce the matrix condition number when polynomial enrichment is used. As pointed out a partition of unity without flat-top property may lead to a singular or nearly singular system of linear equations even with linearly independent local enrichment functions [8–10]. Therefore, having flat-top region in the partition of unity function is important to avoid ill-conditioned system of equations.

There is no unique way to create a flat-top partition of unity on a given computational domain. One can create partition of unity with or without a mesh. To obtain stable h and p refinement, it is recommended to construct a partition of unity function that has flat-top property. On the other hand, maintaining flat-top region within the support of partition of unity function seems challenging without a mesh, and also systematic h refinement is impossible which is a standard technique in the finite element method. Thus, we propose to use a mesh to build flat-top partition of unity function. In the following section, we will introduce how to build a flat-top partition of unity functions effectively on a given mesh.



Fig. 1. Procedure to build the mesh based flat-top partition of unity: (a) partitioned domain (mesh) with elements E_i ; (b) shrunken mesh (dotted lines); (c) flat-top regions (grayed area); (d) interconnections between flat-top regions; (e) Q_i , the support of the partition of unity function ϕ_i ; (f) partition of unity function $z = \phi_i$ subordinated to Q_i .

3. Construction of flat-top partition of unity on a mesh

To ameliorate the cost of numerical integration and to reduce the implementation difficulties, we propose a new flat-top partition of unity function on a mesh by mapping elemental shape functions. The resulting partition of unity function is piecewise linear polynomial with an additional constant function.

We utilize a conventional finite element mesh to construct *patches* (open covering of the domain). In this paper, the term *patches* is different to the commonly used term *elements* in conventional finite element method. The *patches* are allowed to be overlapped and the *i*-th patch Q_i encloses the *i*-th *element*. The flat-top region of the partition of unity function corresponding to a specific element is always within the element.

3.1. Construction procedure

The essential procedure to build a flat-top partition of unity on a general two dimensional domain is illustrated in Fig. 1. The procedure is outlined as follows:

- Step 1: A given domain Ω is partitioned into elements E_i , i = 1, ..., N, see for example Fig. 1(a). When an element has at least two vertices that falls on the boundary Γ of the given domain, we call *boundary element* and if not, we call *interior element*. The set of elements $\{E_i\}$ can be obtained from the conventional finite element mesh generator.
- Step 2: The flat-top region of partition of unity function is determined by simply shrinking the elements as shown in Fig. 1(b). This can be done efficiently by using the mapping concept of the isoparametric finite element procedure as follows:

Let $({}^{i}x_{j}, {}^{i}y_{j})$ be the coordinates for the vertices of element E_{i} and $h_{j}(r, s)$ be the standard linear interpolation functions. We also define a mapping **T** from a reference element \hat{E} to a physical element E_{i} as follows:

$$\mathbf{T}(r,s) = (\mathbf{x}(r,s), \mathbf{y}(r,s)),$$

(2)



Fig. 2. Shrinking elements to obtain flat-top regions for (a) an interior element and (b) a boundary element. \hat{E} is the reference element and E_i is the physical element. **T** is the mapping from \hat{E} to E_i .

where $x(r,s) = \sum_{i=1}^{i} x_i h_i(r,s), y(r,s) = \sum_{i=1}^{i} y_i h_i(r,s)$, see Fig. 2.

Let us define a *flat-top parameter* σ that takes a value between zero and one. The flat-top parameter σ *directly* controls the size of flat-top region in the partition of unity function as illustrated in Fig. 2. Note that σ is defined in the reference coordinate system.

The procedure to obtain flat-top region of boundary element is slightly different to the procedure for the interior element. The difference is necessary to impose boundary conditions correctly using the Kronecker delta property. Let an interior element E_i to be a quadrangle. See for example Fig. 2(a). Then the following four points define the flat-top region of partition of unity function ϕ_i ,

$$\mathbf{T}(-\sigma,-\sigma), \quad \mathbf{T}(+\sigma,-\sigma), \quad \mathbf{T}(+\sigma,+\sigma), \quad \mathbf{T}(-\sigma,+\sigma),$$
(3)

where $0 < \sigma < 1$.

On the other hand, let us consider the case of boundary element E_i that has three vertices on the boundary, see Fig. 2(b). Note that $\mathbf{T}(-\sigma, +\sigma)$ is the only interior point. Then the following four points define the flat-top region of partition of unity function ϕ_i .

$$\mathbf{T}(-\sigma, -1), \quad \mathbf{T}(+1, -1), \quad \mathbf{T}(+1, +\sigma), \quad \mathbf{T}(-\sigma, +\sigma).$$
 (4)

Similar procedure can be applied to a quadrangular element that has two vertices on the boundary.

- Step 3: Step 2 is repeated for all elements E_i , i = 1, ..., N. The shrunken elements are shaded in Fig. 1(c) and each of the shrunken region defines the flat-top region of the partition of unity function ϕ_i , i = 1, ..., N.
- Step 4: After the flat-top regions are identified for each element, the decaying regions can be readily obtained by interconnecting the neighboring flat-top regions. Fig. 1(d) shows the completed interconnections.
- Step 5: Finally, a patch which will be denoted as Q_i is obtained. A patch consists of decaying region and flat-top region and determines the support of a partition of unity function ϕ_i . The patch Q_i is outlined with bold lines in Fig. 1(e).
- Step 6: On each patch Q_i , i = 1, ..., N, we construct flat-top partition of unity function. We map a constant function for the flat-top region, and finite element shape functions and their linear combinations for decaying regions of the partition of unity function as illustrated in Fig. 3. In this step, we also use the mapping technique of the standard isoparametric finite element procedure. The actual partition of unity function $z = \phi_i$ subordinated to the patch Q_i is shown in Fig. 1(f). Note that ϕ_i is compactly supported, piecewise linear, and has a wide flat-top.



Fig. 3. Construction of the piecewise linear flat-top partition of unity function subordinated to a quadrangular element.

3.2. Partition of unity on the boundary patches

On the decaying region of the partition of unity function, at least two partition of unity functions need to be overlapped to fulfill the partition of unity requirement. Next to the convex boundary, however, the decaying region of the partition of unity function contains a triangular area which is overlapped only two times by neighboring partition of unity functions which is insufficient to create partition of unity by the procedure given in Section 3.1. A typical situation is illustrated in Fig. 4. On the triangular decaying region, marked as Δabc in Fig. 4, only two partition of unity functions are defined. Hence, to make $1 = \phi^{\text{left}} + \phi^{\text{right}}$ on the triangular area Δabc , we propose to define two neighboring partition of unity functions ϕ^{left} and ϕ^{right} on Δabc as follows:



(a)



Fig. 4. Partition of unity functions, ϕ^{left} and ϕ^{right} , near the convex boundary: Construction of (a) ϕ^{left} and (b) ϕ^{right} on Δabc .

$$\phi^{\text{left}} = h_a + \frac{1}{2}h_b,$$

$$\phi^{\text{right}} = h_c + \frac{1}{2}h_b,$$
(6)

where h_a , h_b , and h_c are linear shape functions defined on Δabc . Fig. 4(a) and (b) shows how to construct for ϕ^{left} and ϕ^{right} on Δabc . Since the linear shape functions defined on Δabc satisfies $h_a + h_b + h_c = 1$, from Eqs. (5) and (6), the fact $\phi^{\text{left}} + \phi^{\text{right}} = 1$ on Δabc follows immediately.

4. Generalized finite element method with the new flat-top partition of unity function

The Generalized finite element approximation for a second order elliptic boundary value problem can be obtained by the standard Galerkin procedure once the approximation space is known. Hence, we only focus how to construct the finite dimensional approximation space V^{app} with the use of new mesh based flat-top partition of unity functions.

4.1. The finite dimensional vector space

Let us consider the following second order elliptic boundary value problem:

$$\begin{cases} -\Delta u = f & \text{in } \Omega, \\ u = u_d & \text{on } \Gamma_D, \\ \nabla u \cdot \mathbf{n} = u_n & \text{on } \Gamma_N, \end{cases}$$





Fig. 5. An example on the use of different element orders: (a) approximation order marked on each patch and (b) associated nodes distribution within elements.

(7)

where Ω is a domain, **n** is the outward normal vector along boundary Γ and $\overline{\Gamma}_D \cup \overline{\Gamma}_N = \Gamma$, $\Gamma_D \cap \Gamma_N = \emptyset$. Then the corresponding variational equation is the following:

(8)

Find $u \in H^1_p(\Omega)$ such that

$$\mathcal{B}(u, v) = \mathcal{F}(v)$$
 for all $v \in H^1_{\mathsf{D}}(\Omega)$,

where $\mathcal{B}(u, v) = \int_{\Omega} \nabla u \cdot \nabla v$ and $\mathcal{F}(v) = \int_{\Omega} f v + \int_{\Gamma_N} u_n v$. The procedure explained in this section is similar to the generalized finite element method [9,21], except the construction of new partition of unity function. The finite dimensional vector space for the generalized finite element approximation to this model problem can be constructed as follows:

- Step 1: (Generate a mesh) To construct a mesh based partition of unity functions with a flat-top, the domain Ω is partitioned into elements $\{E_i\}$. We supply these elements to generate partition of unity on the computational domain Ω . Although mixture of quadrangular and triangular mesh are allowed, we will use only quadrangular mesh for demonstration purpose.
- Step 2: (Construct flat-top partition of unity functions) Let ϕ_i be the flat-top partition of unity function corresponding to the elements E_i , i = 1, 2, ..., N. We use the procedure that is explained in Section 3.1 to obtain the necessary subregions to define the patch Q_i (the support of partition of unity function) and create the flat-top partition of unity functions on top of Q_i , i = 1, 2, ..., N.
- Step 3: (Determine the order of approximation on each element) We may use same order of approximation for every element. However, the proposed method allows us to use different approximation order on each element. Different approximation orders are marked on each element in Fig. 5(a). Let us recall that the mapping T in Eq. (2) maps reference



Fig. 6. Construction of the global approximation function: (a) compactly supported piecewise polynomial flat-top partition of unity function $\phi_i(x,y)$; (b) quadratic, k = 2, Lagrange interpolating polynomial ${}^{i}L_{k}(x, y)$; and (c) global approximation function ${}^{i}N_{k}(x, y) = \phi_{i}(x, y) \cdot {}^{i}L_{k}(x, y)$.

8694

element \hat{E} to the physical element E_i , i.e. $\mathbf{T}(\hat{E}) = E_i$. Suppose that the order of approximation for element E_i is p. Then uniformly (or quasi-uniformly) distributed nodes ${}^i\hat{p}_k$, $k = 1, ..., n_i = (p + 1)^2$ on the reference element \hat{E} through the mapping \mathbf{T} will place $(p + 1)^2$ points ${}^ip_k = \mathbf{T}({}^i\hat{p}_k)$, $k = 1, ..., n_i$ on $E_i \subset Q_i$. Example of the distribution of nodes with different order p are shown in Fig. 5(b).

Step 4: (*Create interpolation functions*) Let ${}^{i}\hat{L}_{k}$, $k = 1, ..., n_{i}$ are the Lagrange interpolating polynomials of the nodes \hat{p}_{k} in the reference element \hat{E} . Then these interpolating polynomials on the reference element can be used to build interpolating functions ${}^{i}L_{k}$ on the physical element E_{i} as follows:

$${}^{i}L_{k} = {}^{i}\widehat{L}_{k} \circ \mathbf{T}^{-1} \quad k = 1, \dots, n_{i}.$$

$$\tag{9}$$

The Lagrange interpolating polynomials of order p exactly reproduce polynomials up to degree p and have the Kronecker delta property.

Step 5: (*Construct global basis functions*) The global approximation functions ${}^{i}N_{k}(x, y)$ with compact support are constructed as follows:

$${}^{i}N_{k}(x,y) = \phi_{i}(x,y) \cdot {}^{i}L_{k}(x,y), \quad i = 1, \dots, N; \ k = 1, \dots, n_{i}$$
(10)

where the index *i* represents element number and n_i is determined by the approximation order of the *i*-th element. The partition of unity function $\phi_i(x, y)$ and Lagrange interpolating polynomial ${}^iL_k(x, y)$ with k = 2 is pictured in Fig. 6(a) and (b), respectively. The resulting functions ${}^iN_k(x, y)$, see Fig. 6(c), are piecewise polynomials that have compact support. These global approximation functions are continuous and correspond to the nodes:

$${}^{i}p_{k}, \quad i=1,2,\ldots,N; \ k=1,2,\ldots,n_{i}.$$
 (11)

Since ϕ_i is only in the class of C^0 , the regularity of ${}^iN_k(x, y)$ is also C^0 .

Step 6: (Build finite dimensional approximation space) Let the vector space spanned by the approximation functions in Eq. (10) be V^{app}. With the approximation space V^{app}, one can follow the standard Galerkin procedure to get the generalized finite element approximation.



Fig. 7. Numerical integration over the intersection of two patches Q_i and Q_j . The support of ^{*i*}**N** and ^{*j*}**N** are Q_i and Q_j , respectively. Four quadrature points are chosen assuming ^{*i*}**N** and ^{*j*}**N** are quadratic.



Fig. 8. The effect of the flat-top parameter σ on the matrix condition number and the related error. Uniform mesh size h = 0.5 and p = 8 are used.





Fig. 9. Modeling example for the problem with a smooth solution: (a) a mesh with three elements with shaded flat-top regions and different approximation order; (b) distribution of nodes on each element; and (c) error distribution with $\|\cdot\|_{\infty}$ norm.

4.2. Discrete equations

The global approximation u^{app} is obtained as the sum of the local approximation $u^i = \sum_{k=1}^{n_i} c_{i_k}^{\ i} L_k$ multiplied by the partition of unity function ϕ_i :

$$u^{\text{app}}(x,y) = \sum_{i=1}^{N} \phi_i u^i = \sum_{i=1}^{N} \phi_i \left(\sum_{k=1}^{n_i} c_{i_k}{}^i L_k \right) = \sum_{i=1}^{N} \sum_{k=1}^{n_i} c_{i_k} \phi_i{}^i L_k = \sum_{i=1}^{N} \sum_{k=1}^{n_i} c_{i_k}{}^i N_k,$$
(12)

where ${}^{i}N_{k}$ is the *k*-th shape function subordinated to the patch Q_{i} . See for example Fig. 6(c).

Let us use the following vector notation:

$$\mathbf{N} = \begin{bmatrix} {}^{1}\mathbf{N} | {}^{2}\mathbf{N} | {}^{3}\mathbf{N} | \cdots | {}^{N}\mathbf{N} \end{bmatrix}, \quad \mathbf{c} = [\mathbf{c}_{1} | \mathbf{c}_{2} | \mathbf{c}_{3} | \cdots | \mathbf{c}_{N}]^{\mathrm{T}},$$
(13)

where ${}^{i}\mathbf{N} = [{}^{i}N_{1}{}^{i}N_{2}{}^{i}N_{3}{}\cdots{}^{i}N_{k}]$ and $\mathbf{c}_{i} = [c_{i_{1}}|c_{i_{2}}|c_{i_{3}}|\cdots{}|c_{i_{k}}]$. Substituting Eq. (12) into the weak form (8) with the introduced vector notation, we obtain the following discrete linear system,

$$\mathbf{K}\mathbf{C} = \mathbf{R},\tag{14}$$

where

$$\mathbf{K} = \int_{\Omega} (\nabla \mathbf{N})^{\mathrm{T}} \nabla \mathbf{N} d\Omega$$
(15)

and

$$\mathbf{R} = \mathbf{R}_{D} + \mathbf{R}_{N} = \int_{\Omega} \mathbf{N}^{\mathrm{T}} f d\Omega + \int_{\Gamma_{N}} \mathbf{N}^{\mathrm{T}} (\nabla u \cdot \mathbf{n}) d\Omega.$$
(16)

4.3. Numerical integration

We can perform *exact* numerical integration for the stiffness matrix given in Eq. (15). This is possible because ⁱN in Eq. (13) are polynomials on each of the subregions (triangular or quadrangular shape; see for example Fig. 1(e)) that form the support of partition of unity function ϕ_i . The subregions are pre-determined at the early stage, see for example Fig. 1(d), when the flat-top regions are interconnected. Hence, numerical integration can be performed efficiently by creating optimal number of quadrature points in each subregion.



Fig. 10. Mesh sequence on a L-shape domain: (a) h = 1.0; (b) h = 0.5; (c) h = 0.25; and (d) h = 0.125.

Each triangular or quadrangular subregion can be mapped from the reference coordinate (r, s) by mapping **T**, Eq. (2). Hence, the evaluation of partition of unity function and its derivatives at the physical quadrature point are performed in the reference coordinate system as follows:

$$\phi_i(\mathbf{g}_x, \mathbf{g}_y) = \phi_i \circ \mathbf{T}(\mathbf{g}_r, \mathbf{g}_s) = \widehat{\phi}(\mathbf{g}_r, \mathbf{g}_s), \tag{17}$$

(18)

$$\nabla_{xy}\phi_i(\mathbf{g}_x,\mathbf{g}_y) = J(\mathbf{T})^1 \nabla_{rs} \overline{\phi}_i(\mathbf{g}_r,\mathbf{g}_s),$$

where (g_x, g_y) is the physical quadrature point, (g_r, g_s) is the reference quadrature point, $J(\mathbf{T})^T$ is the transpose of Jacobian matrix, and $\hat{\phi}$ is the partition of unity function in the reference coordinate system. Note that $\phi_i = 1$ and $\nabla_{xy}\phi_i = 0$ if the quadrature point is chosen inside the flat-top region.

The evaluation of local approximation functions ${}^{i}L_{k}(\mathbf{g}_{x},\mathbf{g}_{y})$ and its derivatives $\nabla^{i}L_{k}(\mathbf{g}_{x},\mathbf{g}_{y})$ can be done similarly in the reference coordinate system. Therefore, ${}^{i}N_{k}$ in Eq. (13) and its derivatives $\nabla^{i}N_{k} = \nabla\phi_{i}{}^{i}L_{k} + \phi_{i}\nabla^{i}L_{k}$ can be effectively calculated in the reference coordinate system as in the conventional finite element method.

Let us consider an example. Fig. 7 shows the support of ^{*i*}**N** and ^{*j*}**N** and their intersection. Since the support of ^{*i*}**N** and ^{*j*}**N** are restricted by the partition of unity functions, support of ^{*i*}**N** and support of ^{*j*}**N** coincides with the patch Q_i and Q_j , respectively. Note the intersection of two patches $Q_i \cap Q_j$ consists of three rectangular subregions. On each of the subregions, ^{*i*}**N** and ^{*j*}**N** are polynomials so the optimal number of quadrature points can be obtained to perform exact numerical integration. In the case when linear Lagrange interpolating polynomials are used, ^{*i*}**N** and ^{*j*}**N** become piecewise quadratic polynomial because of



Fig. 11. h-Convergence for the problem with a smooth solution.

multiplying partition of unity function. Thus, four quadrature points for each subregions of $Q_i \cap Q_j$ should suffice to obtain *exact* integration for (i,j)th block of the stiffness matrix **K** in Eq. (15).

5. Numerical examples

We provide basic error estimates for the generalized finite element method with the new flat-top partition of unity function. A general error bounds with quasi-reproducing assumption can be found in [1] without flat-top. For the case with flattop partition of unity, similar statement can be found in [9].



Fig. 12. *p*-Convergence for the problem with a smooth solution (C' = 10).



Fig. 13. *h*-Convergence with p = 8 for the problem with a strong singularity.

Theorem. Suppose $u^{\text{app}} \in H^l$ is the Galerkin approximation for the boundary value problem in Eq. (8) and has the polynomial reproducing property of order p. Suppose also $u \in H^q(\Omega)$ and $k + 1 = \min(q, p + 1)$. Then for $0 \le s \le \min\{l, k + 1\}$,

$$||u - u^{app}||_{s,\Omega} \leqslant Ch^{k+1-s} ||u||_{k+1,\Omega},$$
(19)

where C is independent of u and h.

Proof. Theorem 3.5 in *Ref.* [1] along with Theorem 2.4.1 in *Ref.* [23] (Céa's Lemma) provides the proof.

Corollary. If $u \in C^{\infty}$ and $u^{\text{app}} \in H^1$ then k = p, so the previous Theorem implies the following estimate:

$$|u - u^{\operatorname{app}}||_{1,\Omega} \leqslant Ch^{p} ||u||_{p+1,\Omega}.$$
(20)

Also when solution is not smooth, for instance if $u \in H^{5/3-\epsilon}$ with $p \ge 1$ and $u^{app} \in H^1$ then the following convergence bound is the best that can be achieved for all $\epsilon > 0$.

$$||u - u^{app}||_{1,\Omega} \leqslant Ch^{2/3-\epsilon} ||u||_{p+1,\Omega}.$$
(21)

We refer chapter 4.5 of Ref. [24] for Sobolev spaces with fractional indices.

Throughout this section we will denote the strain energy of u as $U(u) = \frac{1}{2}\mathcal{B}(u, u)$ where $\mathcal{B}(\cdot, \cdot)$ represents the bilinear form of the weak formulation in Eq. (8). We also calculate the relative error in energy norm as follows:

$$||\boldsymbol{e}||_{\boldsymbol{E}} = \sqrt{\frac{|\mathcal{U}(\boldsymbol{u}) - \mathcal{U}(\boldsymbol{u}^{\mathrm{app}})|}{\mathcal{U}(\boldsymbol{u})}}.$$
(22)

In this section, the proposed method is tested with different local approximation order p and various mesh size h. The Kronecker delta property is used to impose essential boundary conditions for all numerical examples in this section. Thus, imposing boundary condition is straight forward as in conventional finite element method.

5.1. Effect of flat-top size on the condition number

The condition number is critical to get the convergence of the iterative algorithm such as the conjugate gradient method [25]. As pointed out earlier, piecewise linear partition of unity functions (without flat-top) and using linear polynomials as local approximation space may result in severely ill-conditioned matrix. This phenomenon can be reproduced by taking $\sigma \rightarrow 0$ in our proposed partition of unity function. In the case $\sigma = 0$, the partition of unity function results singular matrix with the use of linear local approximation functions [8].

Using the direct matrix solver, we test the effect of flat-top size on the condition number with the model problem given in Eq. (7) by setting $\Omega = [-1, 1] \times [0, 1]$, $u_d = u_n = 0$, $\Gamma_N = [-1, 1] \times \{y = 1\}$, $\Gamma_D = \partial \Omega \setminus \Gamma_N$ and $f(x, y) = 0.5\pi^2 \cos(0.5\pi x) \sin(0.5\pi y)$.

Fig. 8 clearly demonstrates the effect of flat-top size on the matrix condition number. Based on our numerical tests, it is advised to widen the flat-top size with the increment of approximation order p. We only present the result for p = 8 with



Fig. 14. Error distribution in $\|\cdot\|_{\infty}$ norm for the problem with a strong singularity (h = 0.25 and p = 2).

uniform mesh size h = 0.5. As shown enlarging the flat-top size (σ close to 1) effectively reduces the matrix condition number.

5.2. Convergence test

We test our proposed method on an L-shape domain. A problem with a smooth solution is considered with homogenous essential boundary condition. In the model problem given in Eq. (7), we set $\Gamma = \Gamma_D$, $u_d = 0$, and $f = 2\pi^2 \sin(\pi x) \sin(\pi y)$. The analytic solution to the boundary value problem is $u(x, y) = \sin(\pi x) \sin(\pi y)$.

As a modeling example, we first demonstrate the proposed method with a mesh that consists three elements with h = 1.0 and flat-top parameter $\sigma = 0.9$, as shown in Fig. 9(a). Quartic interpolation is chosen for the element in the second quadrant and quadratic interpolation is used for the rest. With this particular setting, we have distribution of nodes as shown in Fig. 9(b). The degrees of freedom is 19 after imposing boundary conditions. Fig. 9(c) shows the error distribution with $|| \cdot ||_{\infty}$ norm. Note that there is no error along the boundary and the errors are localized in the elements where low order p is used.



Fig. 15. Different configurations for the singular problem; (a) uniform mesh size h = 0.125 with progressively increasing *p* toward the point of singularity and (b) graded mesh with progressively increasing *p* toward the point of singularity.

In the beginning of this section, we provided *a priori* error estimates for the generalized finite element approximation with the new partition of unity function. Since the analytic solution for the smooth boundary value problem is in the class of C^{∞} , according to Eq. (20), we expect the following rate of convergence:

$$||\boldsymbol{u} - \boldsymbol{u}^{\mathsf{app}}||_1 \approx O(h^p), \tag{23}$$

where *p* is the approximation order. To verify the expected rate of convergence, we test the proposed method on different meshes, see Fig. 10.

The expected slope is *p* if the errors in energy norm versus the mesh size *h* are plotted in log–log scale when *p* is fixed and *h* is changed. Only the result of even order *p* is plotted in Fig. 11, but we validated the expected slope for all $1 \le p \le 7$.

On the other hand, according to the Corollary, if we fix the mesh size *h* and change the approximation order *p* instead, we expect $\log(Ch) = -\log(C'/h)$ for the slope where *C'* is the reciprocal of *C*. Numerical results in Fig. 12, show good agreement with the predicted convergence rates. This corresponds to the *p*-convergence in the conventional finite element method.



Fig. 16. Convergence comparison between different configurations. Number next to the marked data points indicates the order p; (a) graded mesh and progressively increasing p toward singularity; (b) uniform mesh size h = 0.125 with progressively increasing p toward singularity; and (c) uniform h-refinement with a fixed p = 8.



Fig. 17. Relative errors in energy norm depending on the configuration of interpolation orders. Unstructured quadrangular mesh is used.

5.3. Problem with a strong singularity on an L-shape domain

We test our proposed method for a non-smooth problem, $u \in H^{5/3-\epsilon}$ for all $\epsilon > 0$, on an L-shape domain with flat-top parameter $\sigma = 0.95$. We consider the Laplace equation with analytical solution $u(r, \theta) = r^{\frac{2}{3}} \sin\left(\frac{2}{3}\theta\right) \in H^{5/3-\epsilon}$. We set $\Gamma_D = \partial \Omega$, $u_d = u, f = 0$ in the model problem given in Eq. (7).

In Fig. 13, we see the rate of convergence, slope $\frac{2}{3}$, with the mesh sequence in Fig. 10 which agrees well with the theoretical prediction, in Eq. (20). Unlike the smooth problem where $u \in C^{\infty}$, we loose the exponential convergence rate even for a relatively small mesh size, h = 0.125. The reason for the slow convergence is due to the presence of the singularity located at the reentrant corner. Fig. 14 shows the error distribution in $|| \cdot ||_{\infty}$ norm with h = 0.25 and p = 2. The error is concentrated near the reentrant corner. However, we can effectively treat the singularity by grading the mesh and increasing p toward the singularity, a well known technique in p version of finite element method [26]. Fig. 15(a) and (b) shows two different strategies we have tested. As shown in Fig. 16, both convergence rates outperform the best convergence rate that is obtained by uniform h refinement with p = 8. Especially Fig. 16(a) shows the typical hp convergence.

We finally demonstrate the flexibility of our proposed method with different order p on a distorted quadrangular mesh. For the test problem we have considered, there was virtually no difference in convergence between structured and unstructured mesh. Different p enrichments and its convergence result on a simple unstructured mesh is shown in Fig. 17 demonstrating the flexible ability of the proposed partition of unity function.

6. Concluding remarks

We have developed a new flat-top partition of unity function that can be built on a given *mesh*. The partition of unity functions are given as piecewise polynomial in *closed form* which is significantly different from the Shepard partition of unity. The generalized finite element approximation with the use of new flat-top partition of unity function possesses three robust features; exact numerical integrations, direct imposition of boundary conditions, and different orders of local *p* enrichment. We presented the construction procedure of the new flat-top partition of unity, showed how to create the generalized finite element approximation space, and described the numerical integration technique.

The flat-top size effect on the matrix condition number was examined. The numerical test showed that a flat-top parameter σ close to 1 (i.e. larger flat-top region) allows stable enrichment. We also tested *h* and *p* convergence and confirmed the theoretical convergence rate for exact solutions to the test problems. Compared to the uniform enrichment case, applying higher order enrichment only in the region where it is needed can save significant number of degrees of freedom.

We applied the new partition of unity function for simple two-dimensional problems only for demonstration purposes. However, recalling that proposed method to construct the flat-top partition of unity poses practical modeling capabilities, the potential of the proposed method in three-dimensional continuum, involving beam, plate and shell problems [26–29] can be explored.

Acknowledgments

This work was supported by the Human Resources Development (No. 20114030200040) of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) Grant funded by the Korea government Ministry of Knowledge Economy.

References

- [1] I. Babuška, U. Banerjee, J.E. Osborn, Survey of meshless and generalized finite element methods: a unified approach, Acta Numer. 12 (1) (2003) 1–125.
- [2] J. Babuška, U. Banerjee, J. Osborn, Generalized finite element methods main ideas, results and perspective, Int. J. Comput. Meth. 1 (1) (2004) 67–103.
- [3] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, P. Krysl, Meshless methods: an overview and recent developments, Comput. Meth. Appl. Mech. Eng. 139 (1-4) (1996) 3-47.
- [4] S. Li, W.K. Liu, Meshfree Particle Methods, Springer, 2004.
- [5] S. Atluri, S. Shen, The Meshless Method, Tech Science Press, 2002.
- [6] G. Liu, Meshfree Methods: Moving Beyond the Finite Element Method, CRC Press, 2002.
- [7] S. De, K.J. Bathe, The method of finite spheres, Comput. Mech. 25 (4) (2000) 329–345.
- [8] R. Tian, G. Yagawa, H. Terasaka, Linear dependence problems of partition of unity-based generalized fems, Comput. Meth. Appl. Mech. Eng. 195 (37–40) (2006) 4768–4782.
- [9] H.S. Oh, J.G. Kim, W.T. Hong, The piecewise polynomial partition of unity functions for the generalized finite element methods, Comput. Meth. Appl. Mech. Eng. 197 (45–48) (2008) 3702–3711.
- [10] D. Cho, A note on the singular linear system of the generalized finite element methods, Appl. Math. Comput. 217 (15) (2011) 6691–6699.
- [11] C.A. Duarte, I. Babuška, J.T. Oden, Generalized finite element methods for three-dimensional structural mechanics problems, Comput. Struct. 77 (2) (2000) 215–232.
- [12] T. Strouboulis, L. Zhang, I. Babuška, Generalized finite element method using mesh-based handbooks: application to problems in domains with many voids, Comput. Meth. Appl. Mech. Eng. 192 (28–30) (2003) 3109–3161.
- [13] T. Strouboulis, I. Babuška, R. Hidajat, The generalized finite element method for helmholtz equation: theory, computation, and open problems, Comput. Meth. Appl. Mech. Eng. 195 (37–40) (2006) 4711–4731.
- [14] T. Strouboulis, K. Copps, I. Babuška, The generalized finite element method, Comput. Meth. Appl. Mech. Eng. 190 (32-33) (2001) 4081-4193.
- [15] T. Strouboulis, I. Babuška, K. Copps, The design and analysis of the generalized finite element method, Comput. Meth. Appl. Mech. Eng. 181 (1-3)
- (2000) 43–69.
- [16] N. Sukumar, N. Mos, B. Moran, T. Belytschko, Extended finite element method for three-dimensional crack modelling, Int. J. Numer. Meth. Eng. 48 (11) (2000) 1549–1570.

- [17] C. Daux, N. Mos, J. Dolbow, N. Sukumar, T. Belytschko, Arbitrary branched and intersecting cracks with the extended finite element method, Int. J. Numer. Meth. Eng. 48 (12) (2000) 1741–1760.
- [18] C.A. Duarte, J.T. Oden, An h-p adaptive method using clouds, Comput. Meth. Appl. Mech. Eng. 139 (1–4) (1996) 237–262.
- [19] I. Babuška, J. Melenk, The partition of unity method, Int. J. Numer. Meth. Eng. 40 (1996) 727-758.
- [20] H.S. Oh, J.G. Kim, J.W. Jeong, The smooth piecewise polynomial particle shape functions corresponding to patch-wise non-uniformly spaced particles for meshfree particle methods, Comput. Mech. 40 (3) (2007) 569–594.
- [21] H.S. Oh, J.W. Jeong, W.T. Hong, The generalized product partition of unity for the meshless methods, J. Comput. Phys. 229 (5) (2010) 1600-1620.
- [22] D. Shepard, A two-dimensional interpolation function for irregularly-spaced data, in: ACM '68: Proc., ACM, New York, NY, USA, 1968, pp. 517–524.
- [23] P.G. Ciarlet, The Finite Element Method for Elliptic Problems, SIAM, Society for Industrial and Applied Mathematics, 2002.
- [24] C. Johnson, Numerical Solution of Partial Differential Equations by the Finite Element Method, Dover Publications, 2009.
- [25] G. Golub, C.V. Loan, Matrix Computations, The Johns Hopkins University Press, 1989.
- [26] B. Szabó, I. Babuška, Finite Element Analysis, Wiley-Interscience, 1991.
- [27] K.J. Bathe, Finite Element Procedures, Prentice Hall, 1995.
- [28] P.S. Lee, K.J. Bathe, Development of MITC isotropic triangular shell finite elements, Comput. Struct. 82 (11–12) (2004) 945–962.
- [29] P.S. Lee, K.J. Bathe, The quadratic MITC plate and MITC shell elements in plate bending, Adv. Eng. Soft. 41 (5) (2010) 712-728.