The Piecewise Polynomial Partition of Unity Functions for the Generalized Finite Element Methods (II)

by

Hae-Soo Oh*†
Department of Mathematics and Statistics,
University of North Carolina at Charlotte, Charlotte, NC 28223

June G. Kim ‡§
Department of Mathematics,
Kangwon National University, Chunchon, 200-701, Korea

Won-Tak Hong
Department of Mathematics and Statistics,
University of North Carolina at Charlotte, Charlotte, NC 28223

August 17, 2007

Dedicated to Professor Ivo Babuška on the occasion of his 80th birthday

Abstract

A partition of unity (PU) function is an essential component of generalized finite element method (GFEM). The popular Shepard PU functions, which are rational functions, are easy to construct, but they have difficulties in dealing with essential boundary conditions and require lengthy computing time for reasonable accuracy in numerical integration. In this paper, we introduce two simple PU functions. The first one, that is a highly regular piecewise polynomial consisting of two distinct polynomials, is effective for uniformly partitioned patches. The second one, that is a highly regular piecewise polynomial consisting of three distinct polynomials, is for arbitrary partitioned patches. These are different from the B-splines.

*Corresponding author. Tel.: +1-704-687-4930; fax: +1-704-687-6415; E-mail: hso@uncc.edu
†supported in part by funds provided by the University of North Carolina at Charlotte
‡supported in part by the Research Grant of the Kangwon National University
§Visiting Professor of the University of North Carolina at Charlotte

Keywords: Reproducing polynomial particle(RPP) shape functions; partition of unity finite element methods (PUFEM); Shepard functions; the B-splines; Convolution partition of unity functions.

1 Introduction

For the last several decades, the Finite Element Method (FEM) has been a powerful tool in solving challenging science and engineering problems, especially when solution domains have complex geometry ([6], [23]). However, several difficulties arise in implementation of this method. The prominent difficulties include the mesh refinements and construction of higher order interpolation fields.

In order to relax the constraints of the conventional FEM, several generalized finite element methods (GFEM), that use the meshes minimally or do not use the meshes at all, were recently introduced. In the literature, there are many names of GFEM ([1],[2],[3],[4]), such as Element Free Galerkin Method (EFGM) ([1],[11],[12]), h-p Cloud Method([7],[8]), Partition of Unity Finite Element Method (PUFEM)([2],[16],[21],[22]). Another non conventional FEM, closely related to GFEM, are Meshfree Particle Methods, such as Reproducing Kernel Particle Method (RKPM)([2],[9],[13]) and Reproducing Kernel Element Method (RKEM) ([13],[14],[15]).

A partition of unity (PU) is an important ingredient in the construction of approximating functions in GFEM. In this paper, we are mainly concerned with constructing piecewise polynomial PU functions that make GFEM being more effective.

GFEM has many advantages over the classical FEM including the freedom of selecting diverse local approximation spaces. However, the popular Shepard PU functions ([3],[13]), which are rational functions, have several limitations such as difficulties in handling essential boundary conditions and inefficiencies attributable to lengthy computing time.

In order to reduce these limitations, we construct two simple highly regular piecewise polynomial PU functions as follows:

1 PU functions for uniformly partitioned patches: In this paper, we construct piecewise polynomial PU functions with given regularity that consist of two distinct polynomials. We prove that the symmetric piecewise polynomial PU function consisting of two polynomials with assigned regularity exist uniquely.

2 PU functions for arbitrary partitioned patches: Suppose the given domain is partitioned into quadrangles (or hexahedrons) of arbitrary sizes. Then, by taking the convolution of the characteristic functions of patches and the given window function, we construct a family of piecewise polynomials that is a partition of unity subordinate to a finite covering of the given domain. Each function is a simple piecewise polynomial and its regularity is one order higher than the regularity order of the freely chosen window function. Actually, we prove that the simple PU function for uniform patches is a special case of the PU functions by the convolution construction.

We do not claim these PU functions are able to eliminate all of the limitations of the Shepard PU functions. However, by selecting proper local approximation spaces, we are able to make all functions of the GFEM approximation space have the Kronecker delta property along the boundary of the domain.

In order to empathize the simplicity of the proposed new PU functions, we also introduce the B-splines([10]) and the reproducing polynomial particle (RPP) shape functions ([17]). The B-splines and the RPP shape functions are not only PU functions, but also have the polynomial reproducing property. Thus, these PU functions are more effective in RKPM and RKEM.

In section 2, notations and terminologies used in this paper are explained. Two model differential equations and the corresponding variational equations are introduced.

In section 3, we introduce five different types of PU functions and compare them. PU functions are: (i) The Shepard functions, (ii) The B-spline functions, (iii) The RPP functions (iv) The highly regular simple("optimal") piecewise polynomial functions, (v) The convolution piecewise polynomial functions. We prove the uniqueness of an optimal PU function of (iv) with respect to the given regularity, and also prove an optimal PU function of (iv) is a special case of the convolution PU function of (v). The properties of this simple piecewise polynomial PU functions are also proved.

In section 4, we describe GFEM, and prove an error bound of the approximations by GFEM with respect to the proposed PU functions. To make the numerical integration being free from evaluating the partition of unity functions in GFEM, we consider quadrature rules in which the PU functions and their derivatives become the weight functions of the quadrature rules.

In section 5, we show the effectiveness of the proposed PU functions in GFEM as follows:

(i) We apply GFEM with respect to various PU functions to the second order differential equations.

(ii) For a proper choice of local approximation functions that can effectively handle essential boundary conditions, we compare the GFEM solution obtained by using orthogonal polynomial local approximation functions with that obtained by the monomial local approximation functions.

(iii) We apply GFEM with respect to the proposed simple PU functions to the fourth order differential equations to demonstrate that the simple piecewise polynomial PU functions are also effective in dealing with the higher order problems.

In section 6, we extend the proposed one-dimensional PU functions to the two dimensional case. For effectiveness of the two dimensional simple PU function in GFEM, we show numerical solutions of Poisson's equation. Furthermore, we compare the condition numbers of stiffness matrices obtained by using various PU functions in GFEM.

In appendix, in order to state the advantages of using the RPP shape functions over the B-splines, we summarize the properties of the B-splines and compare them with those of the RPP shape functions. Furthermore, we constructed two sequences of polynomials that are orthogonal with respect to the weight functions such as the PU functions and their derivatives. Then, the complexity of the numerical integration in GFEM can be reduced on the patches where local approximation functions are these orthogonal polynomials.

2 Preliminary

2.1 Notations and definitions

Let Ω be a domain in $\mathbb{R}^d$. For any nonnegative integer m, $\mathcal{C}^m(\Omega)$ denotes the space of all functions ϕ such that ϕ together with all their derivatives $D^\alpha\phi$ of orders $|\alpha| \leq m$, are continuous on Ω. Here $\alpha \in \mathbb{Z}^d$, $|\alpha| = \alpha_1 + \alpha_2 + \cdots + \alpha_d$. In the following, a function $\phi \in \mathcal{C}^m(\Omega)$ is said to be a $\mathcal{C}^m$- function.

The support of ϕ is defined by

$$\text{supp } \phi = \overline{\{x \in \Omega : \phi(x) \neq 0\}}.$$

We also use the usual **Sobolev space** denoted by $H^k(\Omega)$. For $u \in H^k(\Omega)$, the norm and the semi-norm, respectively, are

$$\|u\|_{k,\Omega} = \Big\{ \sum_{|\alpha|\leq k} \int_\Omega |\partial^\alpha u|^2 dx \Big\}^{1/2}, \quad \text{and } |u|_{k,\Omega} = \Big\{ \sum_{|\alpha|=k} \int_\Omega |\partial^\alpha u|^2 dx \Big\}^{1/2}.$$

The maximum norm of u is defined by $\|u\|_{\infty,\Omega} = \text{ess sup}\Big\{ |u(x)| : x \in \Omega \Big\}$.

A family $\{U_k : k \in \Lambda\}$ of open subsets of $\mathbb{R}^d$ is said to be **a point finite open covering** of $\Omega \subseteq \mathbb{R}^d$ if there is M such that any $x \in \Omega$ lies in at most M of the open sets U_k and $\Omega \subseteq \bigcup_k U_k$.

For a point finite open covering $\{U_k : k \in \Lambda\}$ of a domain Ω, suppose there is a family $\{\phi_k : k \in \Lambda\}$ of Lipschitz functions on Ω satisfying the following conditions:

1. There is a number C such that $\|\phi_k\|_{\infty,\mathbb{R}^d} \leq C$ for each k.

2. supp $(\phi_k) \subseteq \overline{U}_k$, for each $k \in \Lambda$.

3. $\sum_{k\in\Lambda} \phi_k(x) = 1$ for each $x \in \Omega$.

Then $\{\phi_k : k \in \Lambda\}$ is called **a partition of unity (PU)** subordinate to the covering $\{U_k : k \in \Lambda\}$. The covering sets $\{U_k\}$ are called **patches**.

We call a function $\phi(x)$, whose support is $h[-1,1]^d$, **a (basic) PU function** whenever $\{\phi(x - hk) : k \in \mathbb{Z}^d\}$ is a partition of unity, where h is a positive real number. Unlike definition of partition of unity in ([2],[16]), we do not include the condition on the gradient of ϕ_k (that is, $\|\nabla\phi_k\|_{\infty,\mathbb{R}^d} \leq C/diam(U_k)$) because the maximum norm of the gradients of the convolution PU functions is bounded by the inverse of the support size δ of a scaled window function. We will impose the condition: $diam(U_k) \geq 2\delta$ in the construction of the convolution PU functions.

A **window(or weight) function** is a non-negative continuous function with compact support and is denoted by $w(x)$. We consider the following conical window function in this paper: For $x \in \mathbb{R}$,

$$w(x) = \begin{cases} (1-x^2)^l & : \ |x| < 1 \\ 0 & : \ |x| \geq 1, \end{cases} \tag{1}$$

where l is a positive integer.

In $\mathbb{R}^d$, the weight function $w(x)$ can be constructed from the one-dimensional weight function either as $w(x) = w(\|x\|)$ or as $w(x) = \prod_{i=1}^d w(x_i)$, where $x = (x_1, \cdots, x_d)$ and $\|x\|^2 = x_1^2 + \cdots + x_d^2$. In this paper, we use the later one for a higher dimensional window function.

Definition 2.1. *Let $a, b, r \geq 0$, and $m \geq 0$ be integers. Then $\phi_{[(a,b);r;m]}(x)$ is called a reproducing polynomial particle (RPP) shape function with the polynomial reproducing property of order m (or simply, "of reproducing order m") if and only if it satisfies the following condition:*

$$\sum_{k\in\mathbb{Z}^d} (k)^\alpha \phi_{[(a,b);r;m]}(x-k) = x^\alpha, \ \textit{for } x \in \mathbb{R}^d \textit{ and for } 0 \leq |\alpha| \leq m, \alpha \in \mathbb{Z}^d, \tag{2}$$

where $\phi_{[(a,b);r;m]}(x)$ is a $\mathcal{C}^r$-function whose support is $[a,b]^d$.

The RPP shape functions with the property of reproducing order m exactly interpolate all polynomials of degree $\leq m$.

2.2 Model problems

In order to test the effectiveness of various piecewise polynomial PU functions constructed in the forthcoming sections in dealing with high order differential equations, we introduce two one dimensional model problems and the corresponding variational equations.

[a] Let us consider a second order elliptic equation

$$-\frac{d}{dx}\left([c(x)]\frac{du}{dx}\right) = f \text{ in } (a,b) \tag{3}$$

$$u = 0 \text{ at } x = a, b, \tag{4}$$

where $c(x) \gg 0$. Then the corresponding variational equation is

$$\mathcal{B}_1(u,v) \equiv \int_a^b \{[c(x)]\frac{du}{dx}\frac{dv}{dx}\}dx = \int_a^b fvdx \equiv \mathcal{F}_1(v), \tag{5}$$

where

$$u, v \in H_0^1(a,b).$$

[b] Next we consider a fourth order elliptic equation

$$\frac{d^4u}{dx^4} = f \text{ in } (a,b) \tag{6}$$

$$u = \frac{d^2u}{dx^2} = 0 \text{ at } x = a, b. \tag{7}$$

Then for $v \in H^2(a,b) \cap H_0^1(a,b)$, we have

$$\begin{aligned}\int_a^b fv dx &= \int_a^b \frac{d^4u}{dx^4} v dx \\ &= \left[\frac{d^3u}{dx^3}v\right]_a^b - \left[\frac{d^2u}{dx^2}\frac{dv}{dx}\right]_a^b + \int_a^b \frac{d^2u}{dx^2}\frac{d^2v}{dx^2}dx \\ &= \int_a^b \frac{d^2u}{dx^2}\frac{d^2v}{dx^2}.\end{aligned}$$

Thus, the corresponding variational equation is

$$\mathcal{B}_2(u,v) \equiv \int_a^b \{\frac{d^2u}{dx^2}\frac{d^2v}{dx^2}\}dx = \int_a^b fvdx \equiv \mathcal{F}_2(v), \tag{8}$$

where

$$u, v \in H^2(a,b) \cap H_0^1(a,b) = V.$$

Suppose $v \in V$, then we have

$$\begin{aligned}\|v'\|^2_{L^2(a,b)} &= \int_a^b v' \cdot v' dx = [v'v]_a^b - \int_a^b v'' \cdot v dx \\ &\leq \int_a^b |v''| \cdot |v| dx \leq \|v''\|_{L^2(a,b)} \|v\|_{L^2(a,b)} \\ &\leq \|v''\|_{L^2(a,b)} [(b-a)/\pi] \|v'\|_{L^2(a,b)} (\text{ Poincaré inequality }),\end{aligned}$$

which implies

$$\|v'\|_{L^2(a,b)} \leq [(b-a)/\pi] \|v''\|_{L^2(a,b)}.$$

Thus, $\|v\|_{2,(a,b)} \leq C|v|_{2,(a,b)}$ for a constant C and hence, $\mathcal{B}_2(\cdot,\cdot)$ is V-elliptic.

3 Construction of basic Partition of Unity Shape Functions

This section describes constructions of five different kinds of PU functions:

(I) Smooth Shepard (ss) functions, denoted by $\phi^{(ss)}$ ([11], [12], [13]).

(II) B-Spline functions of degree n, denoted by $b_n(x)$ ([10]).

(III) Reproducing Polynomial Particle Shape (RPP) functions denoted by $\phi_{[(a,b);r;m]}(x)$ ([17], [18], [19]).

(IV) $\mathcal{C}^{n-1}$- piecewise polynomial (pp) PU shape functions, denoted by $\phi^{(pp)}_{g_n}$, where n is a positive integer.

(V) Smooth piecewise polynomial convolution PU shape functions with wide flat top, denoted by $\psi_k^{(\delta,l)}(x)$, where δ is the support size of the scaled window function used in the construction, k indicates the patch number, and l is the power of the conical window function $w(x)$.

Note that the B-spline shape functions and the RPP shape functions themselves are used as basis functions for Meshfree particle methods ([10],[17]).

3.1 The Conventional Methods of Constructing PU shape functions

A: Shepard PU shape functions.

The Shepard PU shape function is defined by

$$\phi^{(ss)}(x) = \frac{w(x)}{w(x-1)+w(x)+w(x+1)}, \text{ for all } x \in \mathbb{R} \tag{9}$$

where $w(x)$ is a weight function. Note that $\phi^{(ss)}(x)$ is as smooth as $w(x)$.

Let us note that this rational PU function has generally two major disadvantages: difficulty in dealing with essential boundary conditions and high cost in numerical integrations.

B: The B-Spline shape functions.

There are many ways to define the B-Spline PU functions. The following definition of B-Splines can be found in Höllig ([10]).

Definition 3.1. *The uniform B-**spline** b_n of degree n is defined by the recursion*

$$b_n(x) = \int_{x-1}^{x} b_{n-1}(t)dt, \tag{10}$$

starting from the characteristic function $b_0(x)$ of the unit interval $[0,1)$.

For example, using the relation (65) of the appendix and the hat function defined by

$$b_1(x) = \begin{cases} x & : \ x \in [0,1] \\ 2-x & : \ x \in [1,2] \\ 0 & : \ x \notin [0,2], \end{cases}$$

we have a cubic B-spline which is a basic PU function:

$$b_3(x+2) = \begin{cases} (1/6)[2+x]^3 & : \quad x \in [-2,-1] \\ (1/6)[-3x^3-6x^2+4] & : \quad x \in [-1,0] \\ (1/6)[3x^3-6x^2+4] & : \quad x \in [0,1] \\ (1/6)[2-x]^3 & : \quad x \in [1,2] \\ 0 & : \quad x \notin [-2,2]. \end{cases}$$

The following is a cubic spline PU function, but it is not B-spline.

$$\phi_3^{(sp)}(t) = \begin{cases} (1/2)[2+t]^3 & : \quad t \in [-2,-1] \\ (1/2)[2+t^3] & : \quad t \in [-1,0] \\ (1/2)[2-t^3] & : \quad t \in [0,1] \\ (1/2)[2-t]^3 & : \quad t \in [1,2] \\ 0 & : \quad x \notin [-2,2]. \end{cases} \tag{11}$$

Remark 3.1. *The Shepard-type PU functions and the B-spline PU functions do not satisfy the Kronecker delta property. Thus, they have difficulties in dealing with the essential boundary conditions.*

The B-splines and the RPP shape functions share several common properties. For a comparison of these two PU shape functions, we provide the properties of B-splines in appendix.

3.2 New methods of Constructing Piecewise Polynomial PU shape functions

A: RPP shape functions.

Oh et al ([17]) present various smooth closed form Reproducing Polynomial Particle (RPP) shape functions of high reproducing order. For example, we have the following RPP shape functions in([17]), that are comparable to the cubic B-splines:

(1) A $\mathcal{C}^2$-RPP shape function of reproducing order 2.

$$\phi_{([-2,2];2;2)}(x) = \begin{cases} -\frac{1}{2}(x+2)^3(x+1)(2x+1) & : \quad x \in [-2,-1], \\ \frac{1}{2}(x+1)(6x^4+9x^3-2x+2) & : \quad x \in [-1,0], \\ -\frac{1}{2}(x-1)(6x^4-9x^3+2x+2) & : \quad x \in [0,1], \\ \frac{1}{2}(x-2)^3(x-1)(2x-1) & : \quad x \in [1,2], \\ 0 & : \quad x \notin [-2,2] \end{cases}$$

(2) A $\mathcal{C}^3$-RPP shape function of reproducing order 2.

$$\phi_{([-2,2];3;2)}(x) = \begin{cases} \frac{1}{2}(x+2)^4(x+1)(6x^2+9x+4) & : \quad x \in [-2,-1], \\ -\frac{1}{2}(x+1)(18x^6+45x^5+30x^4+2x-2) & : \quad x \in [-1,0], \\ \frac{1}{2}(x-1)(18x^6-45x^5+30x^4-2x-2) & : \quad x \in [0,1], \\ -\frac{1}{2}(x-2)^4(x-1)(6x^2-9x+4) & : \quad x \in [1,2], \\ 0 & : \quad x \notin [-2,2] \end{cases}$$

(3) A $\mathcal{C}^4$-RPP shape function of reproducing order 2.

$$\phi_{([-2,2];4;2)}(x) = \begin{cases} -\frac{1}{2}(x+2)^5(x+1)(20x^3+50x^2+44x+13) & : \ x \in [-2,-1], \\ \frac{1}{2}(x+1)(60x^8+210x^7+252x^6+105x^5-2x+2) & : \ x \in [-1,0], \\ -\frac{1}{2}(x-1)(60x^8-210x^7+252x^6-105x^5+2x+2) & : \ x \in [0,1], \\ \frac{1}{2}(x-2)^5(x-1)(20x^3-50x^2+44x-13) & : \ x \in [1,2], \\ 0 & : \ x \notin [-2,2] \end{cases}$$

B: Piecewise polynomial PU functions composed of two distinct polynomials.

An optimal choice of a PU function in GFEM depends on the function being approximated. In appropriate circumstances, we have the followings:

1. An optimal $\mathcal{C}^0$-PU function is the hat function.
2. An optimal $\mathcal{C}^1$-PU function is the burble function defined by

$$w_2(x) \quad = \quad \begin{cases} (\cos(\frac{\pi x}{2}))^2 & \text{if } |x| < 1 \\ 0 & \text{if } |x| \geq 1, \end{cases}$$

   or the piecewise polynomial $\phi_{g_2}^{(pp)}$ defined below.

3. In the following, we construct an optimal piecewise polynomial $\mathcal{C}^n$-PU function, for each positive integer n.

The B-splines of degree n and the RPP shape function of order m, respectively, consist of $n+1$ distinct polynomials and $m+2$ distinct polynomials.

In contrast to these, in this subsection, we construct a family of symmetric $\mathcal{C}^{n-1}$-piecewise polynomial basic PU functions, $\phi_{g_n}^{(pp)}(x), n \geq 1$, that consist of only two distinct polynomials. Thus, applying numerical integrations on two subregions separately, one can have exact integrals of these PU functions. Moreover, these highly regular piecewise polynomial PU functions in GFEM are able to solve higher order differential equations (such as bi-harmonic or poly-harmonic problems).

Definition 3.2. *For integers $n \geq 1$, we define a piecewise polynomial function by*

$$\phi_{g_n}^{(pp)}(x) = \begin{cases} (1+x)^n g_n(x) & : \ x \in [-1,0] \\ (1-x)^n g_n(-x) & : \ x \in [0,1] \\ 0 & : \ |x| \geq 1, \end{cases} \tag{12}$$

where $g_n(x) = a_0^{(n)} + a_1^{(n)}(-x) + a_2^{(n)}(-x)^2 + \cdots + a_{n-1}^{(n)}(-x)^{n-1}$ whose coefficients are inductively constructed by the following recursion formula:

$$a_k^{(n)} = \begin{cases} 1, & \text{if} \quad k = 0 \\ \displaystyle\sum_{j=0}^{k} a_j^{(n-1)}, & \text{if} \quad 0 < k \leq n-2, \\ 2(a_{n-2}^{(n)}), & \text{if} \quad k = n-1. \end{cases} \tag{13}$$

The the coefficients $a_k^{(n)}$ can also be obtained by the following recursion formula:

$$a_k^{(n)} = \begin{cases} 1 & \text{if} \quad k = 0, \\ (\frac{n+k-1}{k})a_{k-1}^{(n)} & \text{if} \quad 1 \le k \le n-1. \end{cases} \tag{14}$$

Let us note the followings:

1. The second recurrence relation implies that $a_1^{(n)} = n$ for each n.
2. Using the recurrence relation (13), $g_n(x)$ is as follows:

$$\begin{aligned} g_1(x) &= 1 \\ g_2(x) &= 1 - 2x \\ g_3(x) &= 1 - 3x + 6x^2, \\ g_4(x) &= 1 - 4x + 10x^2 - 20x^3, \\ g_5(x) &= 1 - 5x + 15x^2 - 35x^3 + 70x^4, \\ g_6(x) &= 1 - 6x + 21x^2 - 56x^3 + 126x^4 - 252x^5, \\ g_7(x) &= 1 - 7x + 28x^2 - 84x^3 + 210x^4 - 462x^5 + 924x^6, \\ \vdots \quad & \vdots \qquad \vdots \end{aligned}$$

   $\phi_{g_1}^{(pp)}, \phi_{g_7}^{(pp)}, \phi_{g_{20}}^{(pp)}$, and $\phi_{g_{30}}^{(pp)}$ that are $\mathcal{C}^0, \mathcal{C}^6, \mathcal{C}^{19}, \mathcal{C}^{29}$- functions, respectively, are depicted and compared in Fig. 1.

3. $\phi_{g_1}^{(pp)}$ is the hat function which is a $\mathcal{C}^0$-piecewise linear PU function. $\phi_{g_2}^{(pp)}$ is a $\mathcal{C}^1$-piecewise cubic polynomial PU function.

By using induction arguments, we prove that the piecewise polynomial functions constructed above are actually basic PU functions.

Lemma 3.1. *For all $n \ge 1$, $g_n(x)$ satisfies the following relation:*

$$1 - x^n g_n(x-1) = (1-x)^n g_n(-x). \tag{15}$$

Proof. For $n = 1, 2, 3$, one can easily show that the relation (15) holds.

By an induction argument, we show that it holds for $n+1$ if we assume that it holds for n. Observing that the coefficients of $g_n(x)$ of (13) are written as $a_0 = a_0^{(n-1)}, a_1^{(n)} = a_0^{(n-1)} + a_1^{(n-1)}, \cdots, a_{n-1}^{(n)} = 2a_{n-2}^{(n)}$.

Now we have

$$g_n(-x) \quad = \quad 1 + \sum_{k=1}^{n-1} a_k^{(n)} x^k = x^{n-1} \left[(\frac{1}{x})^{n-1} + \sum_{k=1}^{n-1} a_k^{(n)} (\frac{1}{x})^{n-k-1} \right]. \tag{16}$$

Let

$$(\frac{1}{x})^{n-1} + \sum_{k=1}^{n-1} a_k^{(n)} (\frac{1}{x})^{n-k-1} = (\frac{1}{x} - 1) \left[(\frac{1}{x})^{n-2} + \sum_{k=1}^{n-2} b_k (\frac{1}{x})^{n-k-2} \right] + b_{n-1}.$$

where, upon using (13),

$$b_k = 1 + \sum_{j=1}^{k} a_j^{(n)} = a_k^{(n+1)}, k = 0, 1, 2, ..., n-1$$

Therefore, upon using $a_n^{(n+1)} = 2a_{n-1}^{(n+1)}$,

$$\begin{aligned} g_n(-x) &= (1-x)\Big[\sum_{k=0}^{n-2} a_k^{(n+1)} x^k\Big] + a_{n-1}^{(n+1)} x^{n-1} \\ &= (1-x)\Big[g_{n+1}(-x) - a_{n-1}^{(n+1)} x^{n-1} - a_n^{(n+1)} x^n\Big] + a_{n-1}^{(n+1)} x^{n-1} \\ &= (1-x) g_{n+1}(-x) - a_n^{(n+1)} x^n + a_{n-1}^{(n+1)} x^n + a_n^{(n+1)} x^{n+1} \\ &= (1-x) g_{n+1}(-x) + x^n \big(a_n^{(n+1)} x - a_{n-1}^{(n+1)}\big) \end{aligned} \tag{17}$$

Thus

$$(1-x)^n g(-x) = (1-x)^{n+1} g_{n+1}(-x) + (1-x)^n x^n \big(a_n^{(n+1)} x - a_{n-1}^{(n+1)}\big) \tag{18}$$

On the other hand, by replacing x with $1-x$ in Eqt. (17), we have

$$\begin{aligned} 1 - x^n g_n(x-1) &= 1 - x^n \Big[x g_{n+1}(x-1) + (1-x)^n \big(a_n^{(n+1)}(1-x) - a_{n-1}^{(n+1)}\big)\Big] \\ &= 1 - x^{n+1} g_{n+1}(x-1) - x^n (1-x)^n \Big[- a_n^{(n+1)} x + a_n^{(n+1)} - a_{n-1}^{(n+1)}\Big] \\ &= 1 - x^{n+1} g_{n+1}(x-1) + x^n (1-x)^n \big[a_n^{(n+1)} x - a_{n-1}^{(n+1)}\big] \end{aligned} \tag{19}$$

Now, by applying induction hypotheses in the right hand sides of (18) and (19), we have the identity

$$(1-x)^{n+1} g_{n+1}(-x) = 1 - x^{n+1} g_{n+1}(x-1),$$

for $n+1$. □

Theorem 3.1. *For all $n \geq 1$, the piecewise polynomial functions $\phi_{g_n}^{(pp)}$ defined by (12) satisfy the following.*

- *(i)* $\phi_{g_n}^{(pp)}(x) \in \mathcal{C}^{n-1}(\mathbb{R})$,
- *(ii)* $\phi_{g_n}^{(pp)}(x-1) + \phi_{g_n}^{(pp)}(x) + \phi_{g_n}^{(pp)}(x+1) = 1$ *for* $x \in (-1, 1)$,
- *(iii)* *The degrees of the polynomial on each subinterval are* $(2n-1)$,
- *(iv)* $\phi_{g_n}^{(pp)}(x)$ *is symmetric about the y-axis,*
- *(v)* $\displaystyle\int_{\mathbb{R}} \phi_{g_n}^{(pp)}(x) dx = 1.$

Proof. (i), (iii), and (iv) are straight forward.

If $x \in [0,1]$, then applying (15), we have the followings:

$$\begin{aligned}\sum_{k\in\mathbb{Z}} \phi_{g_n}^{(pp)}(x-k) &= (1-x)^n g_n(-x) + x^n g_n(x-1) \\ &= \Big[1 - x^n g_n(x-1)\Big] + x^n g_n(x-1) = 1,\end{aligned}$$

which is the proof of (ii).

The proof of (v) is as follows:

$$\begin{aligned}\int_{-1}^{0} (1+x)^n g_n(x)dx &= \int_0^1 (1-x)^n g_n(-x)dx (\text{by the symmetry property}) \\ &= \int_0^1 \Big[1 - x^n g_n(x-1)\Big] dx (\text{by the relation (15)})\end{aligned}$$

Thus,

$$\int_{-1}^{0} (1+x)^n g_n(x)dx + \int_0^1 x^n g_n(x-1)dx = 1.$$

By substitution: $x - 1 = -t$, the second integral becomes

$$\int_0^1 x^n g_n(x-1)dx = \int_0^1 (1-t)^n g_n(-t)dt.$$

Thus, we have the property (v). □

The converse of Theorem 3.1 is also true. In other words, we have the following uniqueness theorem:

Theorem 3.2. *(Uniqueness) A piecewise polynomial $\psi(x)$ satisfies the following conditions:*

1. *$\psi(x)$ is symmetric about the y-axis*
2. *$\psi(x)$ is composed of exactly two polynomials of degree $2n-1$.*
3. *$\psi(x)$ is a $\mathcal{C}^{n-1}$-PU function.*
4. *$\psi(0) = 1$, and supp($\psi(x)$) $= [-1, 1]$.*

if and only if

$$\psi(x) = \phi_{g_n}^{(pp)}.$$

Proof. We only need to prove the necessary part: Let us denote the restriction of $\psi(x)$ onto $[0,1]$ by $\psi^+(x)$. Since $\psi(x)$ is a PU function,

$$\psi(x+1)+\psi(x)+\psi(x-1)=1, \text{ for } x\in[-1,1].$$

Moreover, $\psi(x)\in\mathcal{C}^{n-1}(\mathbb{R})$, we have the following:

$$\begin{aligned}
\frac{d^j\psi(x)}{dx^j}|_{x=0} &= 0, \text{ for } j=1,\cdots,n-1,\\
\psi(x)|_{x=0} &= 1,\\
\psi^+(x) &= (1-x)^n(b_0+b_1x+b_2x^2+\cdots+b_{n-1}x^{n-1}).
\end{aligned}$$

Consider the difference of two functions:

$$\begin{aligned}
G(x) &= (1-x)^n g_n(-x)-\psi^+(x)\\
&= (1-x)^n[(a_0^{(n)}-b_0)+(a_1^{(n)}-b_1)x+\cdots+(a_{n-1}^{(n)}-b_{n-1})x^{n-1}]
\end{aligned}$$

Then, we have

$$G(0)=G'(0)=G''(0)\cdots=G^{(n-1)}(0)=0,$$

which implies that

$$\psi^+(x)=(1-x)^n g_n(-x), \text{ for all } x\in[0,1].$$

□

By using the second recursion formula, we can compute the maximum norm of the gradient of $\phi_{g_n}^{(pp)}$ for all n.

Theorem 3.3. *For the $\mathcal{C}^{n-1}$-PU function $\phi_{g_n}^{(pp)}(x)=(1+x)^n g_n(x)$ defined by (12), we have the following:*

(i) $\|[\phi_{g_n}^{(pp)}]'\|_\infty=(2n-1)a_{n-1}^{(n)}(\frac{1}{4})^{n-1}$.

(ii) $\phi_{g_n}^{(pp)}(x)$ *is monotonically increasing on* $[-1,0]$.

(iii) $\|[\phi_{g_n}^{(pp)}]''\|_\infty=(2n-1)(n-1)a_{n-1}^{(n)}|Q(1+Q)|^{n-2}|2Q+1|$, *where*

$$Q=\left[(4n-6)+\sqrt{2(4n-6)}\right]/2(4n-6).$$

Proof. From the definition 3.2, we have

$$\begin{aligned}
[\phi_{g_n}^{(pp)}]'(x) &= n(1+x)^{n-1}g_n(x)+(1+x)^n g_n'(x)\\
&= (1+x)^{n-1}\big(ng_n(x)+(1+x)g_n'(x)\big).
\end{aligned}$$

Table 1: The selected maximum values of $|[\phi_{g_n}^{(pp)}]'(x)|$ and $|[\phi_{g_n}^{(pp)}]''(x)|$.

	$n=2$	$n=3$	$n=5$	$n=7$	$n=10$	$n=15$	$n=20$	$n=30$
$\|[\phi_{g_n}^{(pp)}]'\|_\infty$	1.5	1.875	2.46	2.93	3.52	4.33	5.01	6.15
$\|[\phi_{g_n}^{(pp)}]''\|_\infty$		5.77	9.37	13.18	18.94	28.97	38.26	57.61

Using (14), we have $g_n(x) = \sum_{k=0}^{n-1}(-1)^k a_k^{(n)} x^k, \quad a_0^{(n)} = 1, \quad a_k^{(n)} = \frac{n+k-1}{k} a_{k-1}^{(n)}, \quad 1 \leq k \leq n-1$. Hence, we obtain the following relation

$$\begin{aligned}
& n g_n(x) + (1+x) g_n'(x) \\
&\qquad = n\big(1 + \sum_{k=1}^{n-1}(-1)^k a_k^{(n)} x^k\big) + \sum_{k=1}^{n-1}(-1)^k k a_k^{(n)} x^{k-1} + \sum_{k=1}^{n-1}(-1)^k k a_k^{(n)} x^k \\
&\qquad = n + \sum_{k=1}^{n-1}(-1)^k n a_k^{(n)} x^k + \sum_{k=0}^{n-2}(-1)^{k+1}(k+1) a_{k+1}^{(n)} x^k + \sum_{k=1}^{n-1}(-1)^k k a_k^{(n)} x^k \\
&\qquad = n - a_1^{(n)} + \sum_{k=1}^{n-2}\Big[(n+k)a_k^{(n)} - (k+1)a_{k+1}^{(n)}\Big](-x)^k + (2n-1)a_{n-1}^{(n)}(-x)^{n-1} \\
&\qquad = (2n-1)a_{n-1}^{(n)}(-x)^{n-1},
\end{aligned} \tag{20}$$

which implies $|[\phi_{g_n}^{(pp)}]'(x)| = (2n-1)a_{n-1}^{(n)}|[-x(1+x)]^{n-1}|$ whose maximum value occurs when $x = -1/2$. Thus we have proved that (i) holds.

Since $(1+x)^{(n-1)} \geq 0$ for $x \in [-1, 0]$, (20) implies that $[\phi_{g_n}^{(pp)}]'(x) \geq 0, \quad x \in [-1, 0]$; and hence the assertion (ii) follows.

Applying (20), we have a simple form of the second order derivative

$$[\phi_{g_n}^{(pp)}]''(x) = -(2n-1)(n-1)a_{n-1}^{(n)}[-x(1+x)]^{n-2}(2x+1), \tag{21}$$

and hence the maximum of the second order derivative occurs when $x = [(4n-6) + \sqrt{2(4n-6)}]/2(4n-6)$. $\square$

In table 1 and Fig. 1, the maximum vales of the first derivative and the second order derivative of piecewise polynomial PU functions are shown. These maximum values are parts of error bounds for GFEM in Theorem 4.1.

C: The convolution PU shape functions

Thus far, we have introduced four different kinds of PU shape functions associated mainly with uniformly distributed particles (the Shepard PU functions can be used for non-uniformly partitioned patches).

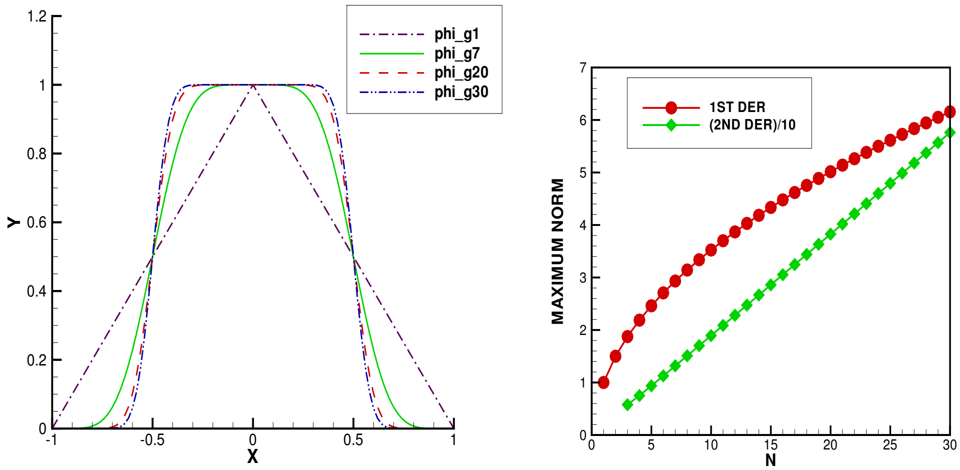

Figure 1: [Left:] The piecewise polynomial PU functions $\phi^{(pp)}_{g1}, \phi^{(pp)}_{g7}, \phi^{(pp)}_{g20}, \phi^{(pp)}_{g30}$ that are $\mathcal{C}^0$-, $\mathcal{C}^6$-, $\mathcal{C}^{19}$-, $\mathcal{C}^{29}$- functions, respectively. [Right:] $\|[\phi^{(pp)}_{gn}]'\|_\infty$, for $n = 1, ..., 30$ and $\|[\phi^{(pp)}_{gn}]''\|_\infty/10$, for $n = 3, ..., 30$

Now we construct a family of functions that is a partition of unity subordinate to an arbitrary covering of a given domain. These PU functions are also used to deal with uniformly (or non uniformly) distributed particles in Meshfree particle methods ([18]).

Consider the scaled window function defined by

$$w^{(l)}_\delta(x) \quad = \quad Aw(\frac{x}{\delta}), \tag{22}$$

where $w(x) = (1 - x^2)^l, |x| \leq 1$, is the conical window function defined by (1) and

$$A^{-1} \quad = \quad \int_{-\delta}^{\delta} w(\frac{x}{\delta})dx.$$

Let $Q_k = (x_k, x_{k+1})$ be an interval with $|x_{k+1} - x_k| \geq 2\delta$ and the characteristic function of Q_k is defined by

$$\chi_{Q_k}(x) = \begin{cases} 1, & \text{if } x \in Q_k, \\ 0, & \text{if } x \notin Q_k. \end{cases} \tag{23}$$

Definition 3.3. *The convolution PU function of* χ_{Q_k} *and* $w_\delta^{(l)}$ *is defined by*

$$\begin{aligned}\psi_k^{(\delta,l)}(x) &= \int_{\mathbb{R}} w_\delta^{(l)}(x-y)\chi_{Q_k}(y)dy = \int_{Q_k} w_\delta^{(l)}(x-y)dy \\ &= \begin{cases} f_{k+1}(x) = \int_{x-x_{k+1}}^{\delta} w_\delta^{(l)}(t)dt, & \text{if } x \in [x_{k+1}-\delta, x_{k+1}+\delta], \\ 1, & \text{if } x \in [x_k+\delta, x_{k+1}-\delta], \\ f_k(x) = \int_{-\delta}^{x-x_k} w_\delta^{(l)}(t)dt, & \text{if } x \in [x_k-\delta, x_k+\delta], \\ 0, & \text{if } x \in \mathbb{R}\backslash[x_k-\delta, x_{k+1}+\delta]. \end{cases}\end{aligned} \tag{24}$$

Since the scaled window function is a polynomial, $\psi_k^{(\delta,l)}(x)$ becomes a piecewise polynomial.

Next, we show that for a fixed δ and a fixed l, $\{\psi_k^{(\delta,l)}\}$ becomes a partition of unity. Suppose a domain $\Omega = (a,b)$ is uniformly (or non-uniformly) partitioned as follows:

$$x_1 = a-\delta < a < x_2 < \cdots x_k < x_{k+1} \cdots < x_n < b < x_{n+1} = b+\delta$$

so that

$$x_{k+1} - x_k \geq 2\delta, \text{ for } 1 \leq k \leq n.$$

Let $Q_k = (x_k, x_{k+1}), k = 1, \cdots, n$. Then we have

$$\sum_{k=1}^{n} \chi_{Q_k}(x) = 1, \text{ for all } x \in \Omega, \text{ except the nodal points,} \tag{25}$$

which implies

$$\sum_{k=1}^{n} \psi_k^{(\delta,l)}(x) = \sum_{k=1}^{n} \left[\int_{\mathbb{R}} \chi_{Q_k}(x-y))w_\delta(y)dy\right] = 1, \text{ for all } x \in \Omega. \tag{26}$$

Thus, these piecewise polynomial $\psi_k^{(\delta,l)}$ have the following properties:

Theorem 3.4. *1.* $supp(\psi_k^{(\delta,l)}(x)) = [x_k-\delta, x_{k+1}+\delta]$.

2. $\{\psi_k^{(\delta,l)}(x) : k = 1, \cdots, n\}$ *is a partition of unity subordinate to an arbitrary covering* $\{Q_k^\delta = (x_k-\delta, x_{k+1}+\delta) : k = 1, \cdots, n\}$ *of* Ω *such that any two are overlapped on an interval with* 2δ *length.*

3.

$$(a) \quad \psi_k^{(\delta,l)}(x) \in \mathcal{C}^l(\mathbb{R}), \tag{27}$$

$$(b) \quad \max|\frac{d}{dx}(\psi_k^{(\delta,l)}(x))| = \max|w_\delta^{(l)}| = A = \mathcal{O}(\delta^{-1}). \tag{28}$$

Proof. (1), (2), and (3-b) are obvious. We only need to prove (3-a).

$$\frac{d\psi_k^{(\delta,l)}(x)}{dx} = \begin{cases} -w_\delta^{(l)}(x-x_{k+1}) & \text{if } x\in[x_{k+1}-\delta, x_{k+1}+\delta], \\ 0, & \text{if } x\in[x_k+\delta, x_{k+1}-\delta], \\ w_\delta^{(l)}(x-x_k) & \text{if } x\in[x_k-\delta, x_k+\delta], \\ 0, & \text{if } x\in\mathbb{R}\backslash[x_k-\delta, x_{k+1}+\delta]. \end{cases} \tag{29}$$

which is continuous and $(l-1)$ differentiable. Thus, the convolution PU function is a $\mathcal{C}^l$-function. □

For example, we have the specific expressions as follows:

- If $l=3$ in the conical window function (1), then the polynomials $f_k(x)$ and $f_{k+1}(x)$ are as follows:

$$\begin{aligned} f_{k+1}(x) &= (\frac{1}{32\delta^7})\left\{[16\delta^3+29\delta^2(x-x_{k+1})+20\delta(x-x_{k+1})^2+5(x-x_{k+1})^3]\right. \\ &\qquad \left.(\delta-x+x_{k+1})^4\right\} & (30) \\ f_k(x) &= (\frac{1}{32\delta^7})\left\{[16\delta^3-29\delta^2(x-x_k)+20\delta(x-x_k)^2-5(x-x_k)^3]\right. \\ &\qquad \left.(\delta+x-x_k)^4\right\}. & (31) \end{aligned}$$

- If $l=5$ in the conical window function, then

$$\max|\frac{d}{dx}(\psi_k^{(\delta,l)}(x))| < (1.28)\times 10^p, \quad \text{if } \delta=10^{-p}, p=0,1,2,\ldots.$$

- For $\delta=0.1, l=3$, the PU functions, $\psi_k^{(\delta,l)}(x)$, are depicted in Fig. 2 when $Q_k, k=1,2,3$, respectively, are $(2,3),(3,3.5)$ and $(3.5,3.9)$.

As a special case of this convolution PU function, by taking

$$\delta = 1/2, \quad x_k=-1/2, \quad x_{k+1}=1/2$$

in (30) and (31), we obtain the following $\mathcal{C}^3$-piecewise polynomial PU function consisting of exactly two distinct polynomials:

$$\psi_{[-1,1]}^{(1/2,3)} = \begin{cases} (1+x)^4(1-4x+10x^2-20x^3) & : \ x\in[-1,0], \\ (1-x)^4(1+4x+10x^2+20x^3) & : \ x\in[0,1], \\ 0 & : \ |x|\geq 1. \end{cases}$$

Hence, we have reproduced $\phi_{g4}^{(pp)}$ as a special case of the convolution PU functions.

Generally, we have the following:

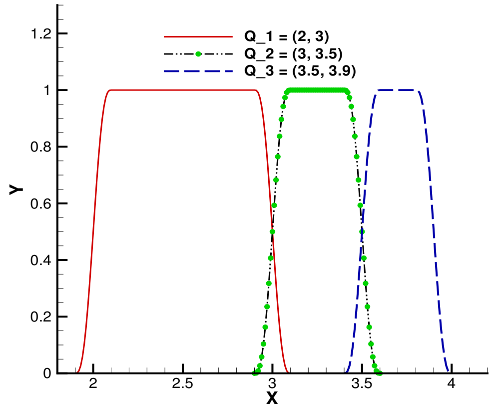

Figure 2: The graphs of $\psi_k^{(\delta,l)} = \chi_{Q_k} * w_\delta^{(l)}$ for $Q_1 = (2,3)$; $\quad Q_2 = (3, 3.5)$; $\quad Q_3 = (3.5, 3.9)$, when $\delta = 0.1$ and $l = 3$

Corollary 3.1. *Suppose* $\psi_0^{(\delta,l)}(x)$ *is the convolution of* $\chi_{Q_0}, Q_0 = (-1/2, 1/2)$, *and the scaled window function* $w_\delta^{(l)}(x)$ *with* $\delta = 1/2$. *Then*

$$\psi_0^{(\delta,l)}(x) = \phi_{g_n}^{(pp)}(x), n = l + 1.$$

Proof. It suffices to show that $\psi_0^{(\delta,l)}(x)$ satisfies the conditions of Theorem 3.2 for $n = l + 1$.

(1)If $n = l + 1$, then $w_\delta^{(l)}(x)$ is a polynomial of degree $2n - 2$; and hence $\psi_0^{(\delta,l)}(x)$ is a piecewise polynomial of two symmetric polynomials of degree $2n - 1$.

(2) By Theorem 3.4, $\psi_0^{(\delta,l)} \in \mathcal{C}^l(\mathbb{R})$; and hence we have $\psi_0^{(\delta,l)} \in \mathcal{C}^{n-1}(\mathbb{R})$.

□

4 Generalized Finite Element Methods

The B-splines, $b_n(x)$, and the RPP shape functions, $\phi_{([a,b];r;n)}(x)$, are able to reproduce the monomials $x^k, 0 \le k \le n$. Thus, these functions have good approximability for Meshfree particle methods.

On the other hand, the PU functions $\phi_{g_n}^{(pp)}(x)$ and $\psi_k^{(\delta,l)}(x)$ are not able to generate the complete polynomials of degree n. Thus, along with these PU functions, we use sets of local approximation functions that have good approximation properties on the patches Q_j.

For example, we may selectively choose one of the following basis $\{f_k(x) : k = 0, 1, ..., N_j\}$ for the local approximation spaces ${}^jV$.

(1) $f_k(\xi) = \xi^k : k = 0, 1, ..., N_j$, which are monomials.

(2) $f_k(\xi) = L_k^{(0)}(\xi) : k = 0, 1, ..., N_j$, which are orthogonal polynomials with respect to the weight $\left(\phi_{g_n}^{(pp)}\right)^2$ on $[-1, 1]$.

(3) $f_k(\xi) = L_k^{(1)}(\xi) : k = 0, 1, ..., N_j$, which are orthogonal polynomials with respect to the weight $\left([\phi_{g_n}^{(pp)}]'\right)^2$ on $[-1, 1]$.

(4) Let $-1 = \xi_0, \xi_1, \cdots, \xi_{N_j} = 1$ be $(N_j + 1)$ distinct points in $[-1, 1]$ and consider the Lagrange interpolating polynomials:

$$L_{N_j,k}(\xi) = \prod_{i=0, i\neq k}^{N_j} \frac{(\xi - \xi_i)}{(\xi_k - \xi_i)}, k = 0, 1, \cdots, N_j$$

Then, for any polynomial $p(x)$ of degree $\leq N_j$, we have

$$\sum_{k=0}^{N_j} p(x_k) L_{N_j,k}(x) = p(x), \text{ for all } x \in \mathbb{R}.$$

That is, $\{L_{N_j,k}(x) : k = 0, 1, \cdots, N_j\}$ reproduces the polynomials of degree $\leq N_j$.

In this paper, we view $L_{N_j,k}(x)$ as a global polynomial defined on $\mathbb{R}$.

(5) $f_k(\xi) = \sin k(\pi/2)(\xi + 1) : k = 0, 1, ..., N_j$, which are the sine functions with various frequencies.

The special orthogonal polynomials, $L_k^{(0)}(\xi)$ and $L_k^{(1)}(\xi)$, for the local approximation functions, are constructed in the appendix.

4.1 GFEM for uniformly distributed patches

Let

$$\begin{aligned} \Omega &= (a, b), \\ h &= (b - a)/N, \\ Q_j^h &= (a + h(j-1), a + h(j+1)), j = 1, 2, \cdots, N. \end{aligned}$$

Next, we define a linear patch mapping $\Psi_j : [-1, 1] \longrightarrow Q_j^h$ as follows:

$$\Psi_j(\xi) = h\xi + x_j, \tag{32}$$

where $x_j = a + hj, j = 0, \cdots, n$. Let

$$\psi_j^{(pp)}(x) = (\phi_{g_n}^{(pp)} \circ \Psi_j^{-1})(x).$$

Then $\{\psi_j^{(pp)}(x) : j = 0, 1, \cdots, N\}$ is a partition of unity subordinate to the covering $\{Q_j^h : j = 0, 1, 2, \cdots, N\}$ of Ω. Note that some patches Q_j^h may go outside Ω.

Now, for each j, let

$$^jV = \text{span}\{\left[{}^j\varphi_k(x) := f_k \circ \Psi_j^{-1}(x)\right]; k = 0, 1, ..., N_j\}$$

be a local approximation space on Q_j^h, where $\{f_k : k = 0, 1, ..., N_j\}$ is a basis of the local approximation space on the patch Q_j^h.

Then a GFEM approximation space is defined by

$$V_h = \text{span}\{[\psi_j^{(pp)} \cdot {}^j\varphi_k] : k = 0, 1, 2, ..., N_j, j = 0, 1, ..., N\}. \tag{33}$$

Now the Galerkin approximation method associated with the finite dimensional vector space V_h is as follow: Find $u_h = \sum_{j=0}^{N} \sum_{k=0}^{N_j} c_{jk}[\psi_j^{(pp)}(x) \cdot {}^j\varphi_k(x)]$ such that

$$\mathcal{B}_1(u_h, \psi_i^{(pp)}(x) \cdot {}^i\varphi_k(x)) = \mathcal{F}_1(\psi_i^{(pp)}(x) \cdot {}^i\varphi_k(x)) \text{ for (3)}, \tag{34}$$

$$\mathcal{B}_2(u_h, \psi_i^{(pp)}(x) \cdot {}^i\varphi_k(x)) = \mathcal{F}_2(\psi_i^{(pp)}(x) \cdot {}^i\varphi_k(x)) \text{ for (6)}, \tag{35}$$

for all i and k.

This method is said to be **generalized finite element method (GFEM).**

Remark 4.1. *(1) Unlike the shepard-type PU functions, $\psi_j^{(pp)}$ and its derivatives are piecewise polynomials consisting of only two distinct polynomials (the minimal number for piecewise polynomials). Thus, the numerical integrals, in the bilinear forms of (34) and (35), are exact by computing the integrals on two subintervals whenever the local approximation functions are polynomials.*

(2) The advantage of GFEM is the freedom of selectively choosing local approximation functions. If the function to be approximated contains a singularity on a patch Q_j^h, then one can chose singular local approximation functions that resembles the singularity. In this case, the numerical integral can not be exact. Optimal choices of local approximation spaces are extensively discussed in ([5]).

(3) Using these orthogonal polynomial approximation functions has some advantages over using the monomial approximation functions in computing stiffness matrices. However, it has disadvantages in dealing with essential boundary conditions as it is shown in the numerical examples of section 5.

[Numerical Integrations for Second Order Equations]

In the following, we will suppress h in Q_j^h, and will write it by by Q_j.

Typical components of the stiffness matrix of the system (34) are of the form

$$
\begin{aligned}
(1)\qquad & \int_{Q_i} [\psi_i^{(pp)} \cdot {}^i\varphi_k]' \cdot [\psi_i^{(pp)} \cdot {}^i\varphi_l]' dx \\
& = \int_{Q_i} \left([\psi_i^{(pp)}]' \cdot {}^i\varphi_k + \psi_i^{(pp)} \cdot [{}^i\varphi_k]'\right) \left([\psi_i^{(pp)}]' \cdot {}^i\varphi_l + \psi_i^{(pp)} \cdot [{}^i\varphi_l]'\right) dx, \qquad (36)\\
(2)\qquad & \int_{Q_i\cap Q_j} [\psi_i^{(pp)} \cdot {}^i\varphi_k]' \cdot [\psi_j^{(pp)} \cdot {}^j\varphi_l]' dx \\
& = \int_{Q_i\cap Q_j} \left([\psi_i^{(pp)}]' \cdot {}^i\varphi_k + \psi_i^{(pp)} \cdot [{}^i\varphi_k]'\right) \left([\psi_j^{(pp)}]' \cdot {}^j\varphi_l + \psi_j^{(pp)} \cdot [{}^j\varphi_l]'\right) dx, \qquad (37)
\end{aligned}
$$

where j is either $i-1$ or $i+1$.

The components of the load vector are of the form

$$\int_{Q_i} f(x)[\cdot\psi_i^{(pp)}(x) \cdot {}^i\varphi_k(x)]dx. \qquad (38)$$

Now these integrals can be transformed into integrals on the reference patch $\bar{Q} = [-1, 1]$ by the patch mapping Ψ_i. For example, the details of computing these components are as follows:

$$
\begin{aligned}
& \int_{Q_i\cap Q_{i-1}} \frac{d}{dx}(\psi_i^{(pp)} \cdot {}^i\varphi_k) \cdot \frac{d}{dx}(\psi_{i-1}^{(pp)} \cdot {}^{i-1}\varphi_l) dx \\
= \; & \int_{-1}^{0} h \left[\frac{d}{dx}([\phi_{g_n}^{(pp)} \cdot f_k] \circ \Psi_i^{-1})\right] \circ \Psi_i \cdot \left[\frac{d}{dx}([\phi_{g_n}^{(pp)} \cdot f_l] \circ \Psi_{i-1}^{-1})\right] \circ \Psi_i \cdot d\xi \\
= \; & \int_{-1}^{1} \frac{1}{2h} \left[\frac{d}{d\xi}(\phi_{g_n}^{(pp)} \cdot f_k)\right] (\frac{\xi-1}{2}) \cdot \left[\frac{d}{d\xi}(\phi_{g_n}^{(pp)} \cdot f_l)\right] (\frac{\xi+1}{2}) \cdot d\xi
\end{aligned}
$$

On the other hand, if we expand (36) and consider termwise integrations, then each term of these integrals can be expressed as

$$
\begin{aligned}
& \int_{-1}^{1} G(\xi) \cdot \frac{d^l[\phi_{g_n}^{(pp)}]}{d\xi^l}(\xi) \cdot \frac{d^m[\phi_{g_n}^{(pp)}]}{d\xi^m}(\xi) \cdot p_k(\xi) d\xi, \qquad 0 \le l, m \le 1 \\
& = \int_{-1}^{0} G(\xi) \cdot \frac{d^l[\phi_{g_n}^{(pp)}]}{d\xi^l}(\xi) \cdot \frac{d^m[\phi_{g_n}^{(pp)}]}{d\xi^m}(\xi) \cdot p_k(\xi) d\xi \\
& + \int_{0}^{1} G(\xi) \cdot \frac{d^l[\phi_{g_n}^{(pp)}]}{d\xi^l}(\xi) \cdot \frac{d^m[\phi_{g_n}^{(pp)}]}{d\xi^m}(\xi) \cdot p_k(\xi) d\xi \qquad (39)
\end{aligned}
$$

where $G(\xi)$ is the product of Jacobian functions and the coefficient functions of differential equation, and $p_k(\xi)$ is a product of two local approximation functions or their derivatives. Hence, $p_k(\xi)$ is a polynomial when the local approximation functions are polynomials.

Thus, we have the following features about the numerical integrations for the stiffness matrices:

- If $G(\xi)$ is a polynomial and local approximation functions are polynomial, the integrands of the last two integrals are polynomials, and hence one can exactly calculate stiffness matrices at a lower computing cost.

- Or, we can observe the computation of stiffness matrices as numerical integrations of the form

$$\int_{-1}^{1} W_1^{ord(2)}(\xi) g(\xi) d\xi, \tag{40}$$

where $g(\xi) = G(\xi) \cdot p_k(\xi)$ and $W_1^{ord(2)}(\xi)$ is one of the following functions:

$$\phi(\xi), \quad \phi(\xi)\phi(\xi), \quad \phi(\xi)\phi'(\xi), \quad \phi'(\xi)\phi'(\xi), \tag{41}$$

where $\phi(\xi) = [\phi_{g_n}^{(pp)}](\xi)$.

Along this interpretation, we have to deal with numerical integrations of the form

$$\int_{0}^{1} W_2^{ord(2)}(\xi) g(\xi) d\xi, \tag{42}$$

where $W_2^{ord(2)}(\xi)$ is one of the tensor product $\{\phi((\xi+1)/2), \phi'((\xi+1)/2)\} \otimes \{1, \phi((\xi-1)/2), \phi'((\xi-1)/2)\}$. More specifically, it is one of the following functions:

$$\begin{array}{lll} \phi((\xi+1)/2) & \phi((\xi-1)/2)\phi((\xi+1)/2) & \phi'((\xi-1)/2)\phi(\xi+1)/2) \\ & \phi((\xi-1)/2)\phi'(\xi+1)/2) & \phi'((\xi-1)/2)\phi'(\xi+1)/2) \end{array} \tag{43}$$

For effective calculations of these integrals, we could consider eight quadrature rules with respect to these eight weight functions listed in (41) and (43). However, we sometimes use singular local approximation functions on a patch containing singularity. In this case, we have no advantages in using these special quadrature rules. However, one can selectively use these special quadrature rules for the patches where local approximation functions are polynomials. The orthogonal polynomials to weight functions $W(x) = [\phi_{g_n}^{(pp)}]^2$ and $W(x) = [\frac{d}{dx}\phi_{g_n}^{(pp)}]^2$ are given in appendix.

[Numerical Integrations for Fourth Order Equations]

Typical components of the stiffness matrix of the system (35) are of the form

$$\begin{aligned} (1) \quad & \int_{Q_i} \Big([\psi_i^{(pp)}]'' \cdot {}^i\varphi_k + 2[\psi_i^{(pp)}]' \cdot {}^i\varphi_k' + [\psi_i^{(pp)}] \cdot {}^i\varphi_k'' \Big) \\ & \qquad \cdot \Big([\psi_i^{(pp)}]'' \cdot {}^i\varphi_l + 2[\psi_i^{(pp)}]' \cdot {}^i\varphi_l' + [\psi_i^{(pp)}] \cdot {}^i\varphi_l'' \Big)\, dx, \end{aligned} \tag{44}$$

$$\begin{aligned} (2) \quad & \int_{Q_i \cap Q_j} \Big([\psi_i^{(pp)}]'' \cdot {}^i\varphi_k + 2[\psi_i^{(pp)}]' \cdot {}^i\varphi_k' + [\psi_i^{(pp)}] \cdot {}^i\varphi_k'' \Big) \\ & \qquad \cdot \Big([\psi_j^{(pp)}]'' \cdot {}^j\varphi_l + 2[\psi_j^{(pp)}]' \cdot {}^j\varphi_l' + [\psi_j^{(pp)}] \cdot {}^j\varphi_l'' \Big)\, dx, \end{aligned} \tag{45}$$

where $j = i+1, i-1$, and components of the load vector are of the form

$$\int_{Q_i} f(x) \cdot [\psi_i^{(pp)}](x) \cdot {}^i\varphi_k(x) dx. \tag{46}$$

Similarly, letting $[\psi_j^{(pp)}] = \phi$, we have the following relation for the fourth order equations:

$$\begin{aligned}
&\int_{Q_i \cap Q_{i-1}} \frac{d^2}{dx^2}([\psi_i^{(pp)}] \cdot {}^i\varphi_k) \cdot \frac{d^2}{dx^2}([\psi_{i-1}^{(pp)}] \cdot {}^{i-1}\varphi_l) dx \\
= &\int_{-1}^{0} h \left[\frac{d^2}{dx^2}([\phi \cdot f_k] \circ \Psi_i^{-1}) \right] \circ \Psi_i \cdot \left[\frac{d^2}{dx^2}([\phi \cdot f_l] \circ \Psi_{i-1}^{-1}) \right] \circ \Psi_i \cdot d\xi \\
= &\int_{-1}^{1} \frac{1}{2h^3} \left[\frac{d^2}{d\xi^2}(\phi \cdot f_k) \right] (\frac{\xi - 1}{2}) \cdot \left[\frac{d^2}{d\xi^2}(\phi \cdot f_l) \right] (\frac{\xi + 1}{2}) \cdot d\xi
\end{aligned}$$

The integrand of the last integral is a polynomial on $[-1, 0]$ and on $[0, 1]$ whenever the local approximation functions are polynomials. Hence, the components of the stiffness matrices can be exactly computed in such cases.

4.2 Error bounds

In this section, by using existing error analysis ([2]), we state more specific error bounds of GFEM with respect to the piecewise polynomial PU function $\phi_{g_n}^{(pp)}$.

Let $(\Psi_n^{(pp)})_i(x) = \phi_{g_n}^{(pp)}((x - x_i)/h)$. Then from Table 1, we have specific error bounds: for example, if $n = 3$, then

$$\|(\Psi_n^{(pp)})_i(x)\|_{L^\infty(\mathbb{R})} = 1, \tag{47}$$

$$\|\frac{d}{dx}(\Psi_n^{(pp)})_i(x)\|_{L^\infty(\mathbb{R})} = \|[\phi_{g_n}^{(pp)}]'\|_{L^\infty(\mathbb{R})}/h < 1.88/h, \tag{48}$$

$$\|\frac{d^2}{dx^2}(\Psi_n^{(pp)})_i(x)\|_{L^\infty(\mathbb{R})} = \left[\|[\phi_{g_n}^{(pp)}]''\|_{L^\infty(\mathbb{R})} \right] /h^2 < 5.77/h^2. \tag{49}$$

Thus, Theorem 6.1 of [2] can be modified as follows:

Theorem 4.1. *Suppose $u \in H^2(\Omega)$ and $\phi_{g_n}^{(pp)}$ is a basic PU function in GFEM. If, for all $i = 0, 1, 2, ..., N$, there exists ${}^iu_{(gfe)}$ on each patch Q_i such that*

$$\|u - {}^iu_{(gfe)}\|_{L_2(\Omega \cap Q_i)} \leq \varepsilon_1(i), \tag{50}$$

$$\|\frac{d}{dx}(u - {}^iu_{(gfe)})\|_{L_2(\Omega \cap Q_i)} \leq \varepsilon_2(i), \tag{51}$$

$$\|\frac{d^2}{dx^2}(u - {}^iu_{(gfe)})\|_{L_2(\Omega \cap Q_i)} \leq \varepsilon_3(i). \tag{52}$$

Then, the GFE solution

$$u_{(gfe)} = \sum_{i=0}^{N} \Psi_i^{(pp)} \cdot {}^i u_{(gfe)} \in V_h$$

satisfies

$$\|u - u_{(gfe)}\|_{L_2(\Omega)} \leq \sqrt{2} \left(\sum_{i=0}^{N} \varepsilon_1(i)^2 \right)^{1/2} \tag{53}$$

$$\|\frac{d}{dx}(u - u_{(gfe)})\|_{L_2(\Omega)} \leq 2 \left(\sum_{i=0}^{N} \left[\varepsilon_2(i)^2 + (\|[\phi_{g_n}^{(pp)}]'\|_{L^\infty(\mathbb{R})}/h)^2 \varepsilon_1(i)^2 \right] \right)^{1/2} \tag{54}$$

$$\|\frac{d^2}{dx^2}(u - u_{(gfe)})\|_{L_2(\Omega)} \leq 2 \left(\sum_{i=0}^{N} \left[\varepsilon_3(i)^2 + 2(\|[\phi_{g_n}^{(pp)}]'\|_{L^\infty(\mathbb{R})}/h)^2 \varepsilon_2(i)^2 + \left([\|[\phi_{g_n}^{(pp)}]''\|_{L^\infty(\mathbb{R})}/h^2] \varepsilon_1(i) \right)^2 \right] \right)^{1/2} \tag{55}$$

Proof. The proofs of (53) and (54) are similar to the proofs of Theorem 6.1 of [2]. Thus, we only need to prove (55) that is not in [2].

Since $\{\Psi_i^{(pp)} : i = 0, 1, ..., N\}$ is a partition of unity subordinate to the open covering $\{Q_i^\circ : i = 0, 1, 2, ..., N\}$,

$$\begin{aligned} \|\frac{d^2}{dx^2}(u - u_{(gfe)})\|^2_{L_2(\Omega)} &= \|\frac{d^2}{dx^2} \sum_{i=0}^{N} \Psi_i^{(pp)}(u - {}^i u_{(gfe)})\|^2_{L_2(\Omega)} \\ \leq 2\| \sum_{i=0}^{N} \left[\frac{d^2}{dx^2} \Psi_i^{(pp)} \right] (u - {}^i u_{(gfe)})\|^2_{L_2(\Omega)} &+ 4\| \sum_{i=0}^{N} \frac{d}{dx} \Psi_i^{(pp)} \frac{d}{dx}(u - {}^i u_{(gfe)})\|^2_{L_2(\Omega)} \\ &+ 2\| \sum_{i=0}^{N} \Psi_i^{(pp)} \left[\frac{d^2}{dx^2}(u - {}^i u_{(gfe)}) \right] \|^2_{L_2(\Omega)}. \end{aligned} \tag{56}$$

For each $x \in \Omega \subset \mathbb{R}$, the sums of (56) have at most two nonzero terms. Therefore, we have

$$\begin{aligned} | \sum_{i=0}^{N} \left[\frac{d^2}{dx^2} \Psi_i^{(pp)} \right] \left[(u - {}^i u_{(gfe)}) \right] |^2 &\leq 2 \sum_{i=0}^{N} | \left[\frac{d^2}{dx^2} \Psi_i^{(pp)} \right] \left[(u - {}^i u_{(gfe)}) \right] |^2, \\ | \sum_{i=0}^{N} \left[\frac{d}{dx} \Psi_i^{(pp)} \right] \left[\frac{d}{dx}(u - {}^i u_{(gfe)}) \right] |^2 &\leq 2 \sum_{i=0}^{N} | \left[\frac{d}{dx} \Psi_i^{(pp)} \right] \left[\frac{d}{dx}(u - {}^i u_{(gfe)}) \right] |^2, \\ | \sum_{i=0}^{N} \left[\Psi_i^{(pp)} \right] \left[\frac{d^2}{dx^2}(u - {}^i u_{(gfe)}) \right] |^2 &\leq 2 \sum_{i=0}^{N} | \left[\Psi_i^{(pp)} \right] \left[\frac{d^2}{dx^2}(u - {}^i u_{(gfe)}) \right] |^2 . \end{aligned}$$

Hence, (47), (48), (49), and (56) imply the desired result.

□

5 Numerical solutions of the second and the fourth order elliptic equations

In this section, we solve the second order and the fourth order differential equations by using the GFEM approximation space constructed with aid of the PU functions $\phi_{g_n}^{(pp)}$.

Let us note that for the numerical solutions for the second order equations, we can employ the PU functions $\phi_{g_n}^{(pp)}$ for all $n \geq 1$. However, for the fourth order equations, we use the PU function $\phi_{g_n}^{(pp)}$ for $n \geq 3$ in GFEM.

Let $\mathcal{U}(w) = \frac{1}{2}\mathcal{B}(w, w)$ be the strain energy of w, where $\mathcal{B}(\cdot,\cdot)$ denote the bilinear forms defined in section 2. Then the relative error in energy norm is defined as

$$\|e\|_{E,r} = \left[\frac{|\mathcal{U}(u_{ex}) - \mathcal{U}(u_{app})|}{\mathcal{U}(u_{ex})}\right]^{1/2}.$$

All test problems of this section are solved by GFEM with respect to following data

$$\begin{aligned} \Omega &= (-1, 1), \\ h &= 0.5, \\ x_i &= -1 + h \cdot i, \quad i = 0, 1, 2, ..., 4, \\ N_i &= M, \text{ for all } i \text{ (uniform dimension of local approximation spaces).} \end{aligned}$$

5.1 Numerical Solutions for the Second Order Differential Equations

Example 1. $u(x) = e^x(1 - x^2)$ solves the following model problem

$$-\frac{d^2}{dx^2}u(x) = f(x) \text{ in } (-1, 1),$$

where $f(x) = e^x(1 + 4x + x^2)$. Then the strain energy is

$$\frac{1}{2}\int_\Omega (\frac{du}{dx})^2 dx = 2.6524776642670.$$

To compare an approximablity of local approximation functions: $\phi_{g_n}^{(pp)}(\xi) \otimes \{f_k(\xi) : k = 0, 1, 2, \cdots, N\}$, we test a various piecewise polynomial PU functions, $\phi_{g_1}^{(pp)}$(the hat function), $\phi_{g_3}^{(pp)}$, $\phi_{g_7}^{(pp)}$, and $\phi_{g_{30}}^{(pp)}$ in GFEM.

As one can see from Table 2 and Fig. 3, GFEM for all of these piecewise polynomial PU functions yields highly accurate results. Theorem 4.1, together with Theorem 3.3(or Table 1), implies that the errors are getting bigger as n (in $\phi_{g_n}^{(pp)}$) is increasing. Results in Table 2 and Fig. 3 support this theory.

Table 2: Relative error (%) in energy norm ($\|e\|_{E,r} \times 100$) of GFEM solutions when the piecewise polynomial PU functions $\phi^{(pp)}_{g_n}, n = 1, 3, 7, 30$, respectively, are used

p-deg	g_1	g_3	g_7	g_{30}
1	6.703986	25.2157	29.8296	32.1909
2	0.505911	2.83537	0.8166	2.18725
3	0.024968	0.20079	0.0908	0.22226
4	0.000834	0.00910	0.0080	0.00620
5	0.000335	0.00076	0.0017	0.00434
6	0.000058	0.00087	0.0016	0.00433

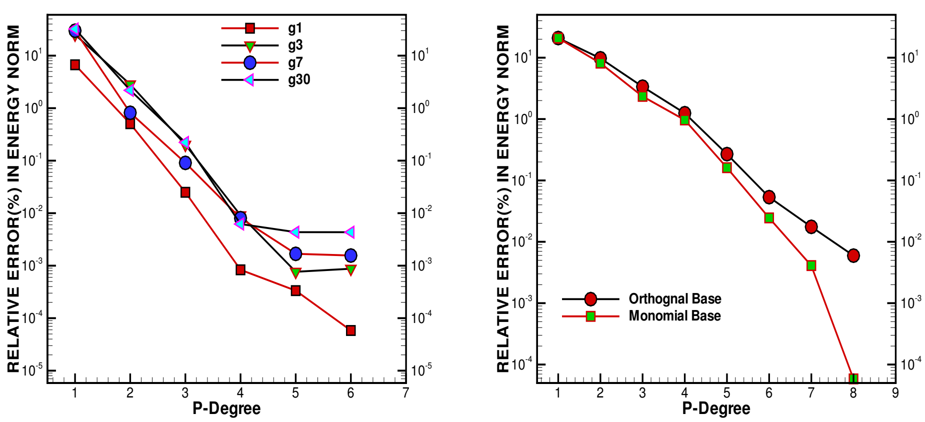

Figure 3: [Left] Relative error(%) in energy norm of GFEM when the piecewise polynomial PU functions $\phi^{(pp)}_{g_n}, n = 1, 3, 7, 30$, respectively, are used. [Right] Relative error (in percent) in Energy Norm of GFEM solutions for the monomial local approximation functions and the orthogonal local approximation functions that are orthogonal with respect to the weight function $[\phi^{(pp)}_{g_3}]'(x)^2$

Let $\{L_k(x), k = 0, 1, 2, ...\}$ be the sequence of orthogonal polynomials such as Legendre polynomials and those polynomial in Table 5, and $\{P_k(x) = x^k, k = 0, 1, 2, ...\}$ be a sequence of monomials Then

$$\begin{cases} L_k(0) = 0 & \text{if } k \text{ is an odd integer} \\ L_k(0) \neq 0 & \text{if } k \text{ is an even integer,} \end{cases} \quad \text{and} \quad \begin{cases} P_k(0) = 0 & \text{if } k \neq 0 \\ P_k(0) = 1 & \text{if } k = 0. \end{cases}$$

Suppose the boundary condition at one end point is Dirichlet and the reference basis functions are $P_k(x) = x^k, k = 0, 1, 2, ..., M$. Then we only need to constrain P_0. On the other hand, if $\{L_k(x), k = 0, 1, 2, ..., M\}$ is used as local basis functions, we have to constrain all even degree orthogonal polynomials $L_{2k}(x), k = 0, 1, 2, \cdots$. In this case, GFEM for the orthogonal basis functions will lose exactly one half local degree of freedom at the end point node. Thus GFEM for the orthogonal basis functions $\{L_k(x), k = 0, 1, 2, ..., M\}$ may yield inferior results than GFEM for the monomial basis functions $\{x^k, k = 0, 1, ..., M\}$. This is demonstrated in the next example.

Example 2. $u(x) = e^x(1 - x^2)^6$ solves the following model problem

$$-\frac{d^2}{dx^2}u(x) = f(x) \text{ in } (-1, 1) \tag{57}$$

where $f(x) = -e^x((1 - x^2)^6 - 24x(1 - x^2)^5 - 12(1 - x^2)^5 + 120x^2(1 - x^2)^4)$. Then the strain energy is 1.8568504251271325.

For numerical solutions, we apply GFEM, with $\mathcal{C}^3$-PU shape function $\phi_{g3}^{(pp)}$, for two sets of local basis functions: (i) the orthogonal polynomial basis functions, that are constructed in appendix (Table 5), and (ii) the monomial basis functions. The relative error (%) in energy norm are depicted in Fig. 3. Since the differences in degrees of freedom between two cases is increasing as the p-degree of polynomial basis is increasing, the results for the monomial basis is improving compared to that of orthogonal basis as the p-degree is increasing.

If $\phi_{gn}^{(pp)}(\xi) \otimes \{L_{n,k}(\xi) : k = 0, 1, 2, \cdots, N\}$, were used for the local basis functions, then each function of the GFEM approximation space has the Kronecker delta property; and hence they can handle the essential boundary conditions in the most effective way. However, the Lagrangian polynomials have a longer computing time than the monomials.

5.2 Numerical Solutions for the Fourth Order Differential Equations

In conventional FEM, the Lagrange basis functions is unable to handle the fourth order differential equation, and hence Hermite-type-interpolating basis functions have to be constructed for the valid FE solutions.

However, GFEM related to the local basis $\phi_{gn}^{(pp)} \otimes \{f_k(\xi) : k = 0, 1, 2, \cdots, N\}$, can effectively handle the following fourth order equations whenever $n \geq 3$.

Example 3. $u(x) = (1 - x^2)^3$ solves the following model problem

$$\frac{d^4}{dx^4}u(x) = f(x) \text{ in } (-1, 1) \tag{58}$$

$$u(\pm 1) = \frac{d^2u}{dx^2}(\pm 1) = 0, \tag{59}$$

Table 3: In case the true solution of the fourth order equation is $u(x) = (1 - x^2)^3$, the computed Strain Energy and Relative error in percent with respect to the PU function $\phi_{g_n}^{(pp)}, n = 3, 7$, respectively, are presented

	$\phi_{g_3}^{(pp)}$		$\phi_{g_7}^{(pp)}$	
p-deg	Strain Energy	$\|e\|_{E,r} \times 100$	Strain Energy	$\|e\|_{E,r} \times 100$
1	7.0006138392856	72.21	3.9575930714358	85.41
2	13.065142760779	32.69	13.422758749197	28.71
3	14.401452691593	12.46	14.584210106868	5.51
4	14.621316438602	2.23	14.627138618156	0.99
5	14.628525836503	0.18	14.628506376614	0.21
6	14.628571428571	0.0	14.628571428571	0.0
7	14.628571428571	0.0	14.628571428571	0.0

where

$$f(x) = 72 - 360x^2.$$

Then the strain energy is

$$\frac{1}{2}\int_\Omega (\frac{d^2u}{dx^2})^2 dx = 14.6285714285714.$$

Since the true solution is a polynomial of degree 6, we expect that GFEM with monomial local basis yields an exact solution if the p-degree of the polynomial local approximation functions is ≥ 6. Table 3 shows the expected results.

Example 4. $u(x) = e^x(1 - x^2)^6$ solves the following model problem

$$\frac{d^4}{dx^4}u(x) = f(x) \text{ in } (-1, 1) \tag{60}$$

$$u(\pm 1) = \frac{d^2u}{dx^2}(\pm 1) = 0, \tag{61}$$

where

$$\begin{aligned} f(x) &= e^x\left[(1-x^2)^6 - 48(1-x^2)^5x + 720(1-x^2)^4x^2 - 72(1-x^2)^5 - 3840(1-x^2)^3x^3\right. \\ &+ \left. 1440(1-x^2)^4x + 5760(1-x^2)^2x^4 - 5760(1-x^2)^3x^2 + 360(1-x^2)^4\right]. \end{aligned}$$

Then the strain energy is

$$\frac{1}{2}\int_\Omega (\frac{d^2u}{dx^2})^2 dx = 36.5967697946804.$$

Table 4: In case the true solution is $u(x) = e^x(1-x^2)^6$, the computed Strain Energy and Relative error in percent with respect to the PU function $\phi_{g_n}^{(pp)}, n = 3, 7$, respectively, are presented

	$\phi_{g_3}^{(pp)}$		$\phi_{g_7}^{(pp)}$	
p-deg	Strain Energy	$\|e\|_{E,r} \times 100$	Strain Energy	$\|e\|_{E,r} \times 100$
1	23.063844246881	60.81	10.2300079812235	84.88
2	30.531319564732	40.71	31.610797528159	36.91
3	35.129578103172	20.02	35.058958147538	20.50
4	36.358613687588	8.07	36.563612470462	3.01
5	36.584153082802	1.86	36.592112462508	1.13
6	36.596278547147	0.37	36.596433383392	0.303
7	36.596757502127	0.058	36.596768474234	0.019
8	36.596769791046	0.001	36.596769604282	0.007

Relative errors in percent for this fourth order differential equation are depicted in Fig. 4. The computed strain energies obtained by GFEM with respect to the PU functions, $\phi_{g_3}^{(pp)}$ and $\phi_{g_7}^{(pp)}$, are compared in Table 4.

From the forgoing two examples, we have observed the following:

1. GFEM for $\phi_{g_n}^{(pp)}, n \geq 3$, can solve the fourth order equations.

2. From Table 1, we have

$$\begin{aligned} \|[\phi_{g_7}^{(pp)}]''\|_\infty (= 13.18) &\gg \|[\phi_{g_3}^{(pp)}]''\|_\infty (= 5.77), \\ \|[\phi_{g_7}^{(pp)}]'\|_\infty (= 2.93) &\gg \|[\phi_{g_3}^{(pp)}]'\|_\infty (= 1.88). \end{aligned}$$

   From the error estimates of Theorem 4.1, we expect that in the case where $\phi_{g_3}^{(pp)}$ is used for GFEM is better than the cases where $\phi_{g_7}^{(pp)}$is used. However, Fig. 6 and Table 8 show that in the case of lower polynomial degree, the global approximation space constructed by $\phi_{g_7}^{(pp)} \otimes \{1, x, x^2, ..., x^7\}$ approximates $e^x(1-x^2)^6$ slightly better than the approximation space constructed by using $\phi_{g_3}^{(pp)} \otimes \{1, x, x^2, ..., x^7\}$. We observe a similar fact in Table 7. In other words, $\phi_{g_7}^{(pp)}$ works better in lower p-degree, however $\phi_{g_3}^{(pp)}$ works better in higher p-degree.

3. We can also apply this method to the fourth order problems with other types of boundary conditions.

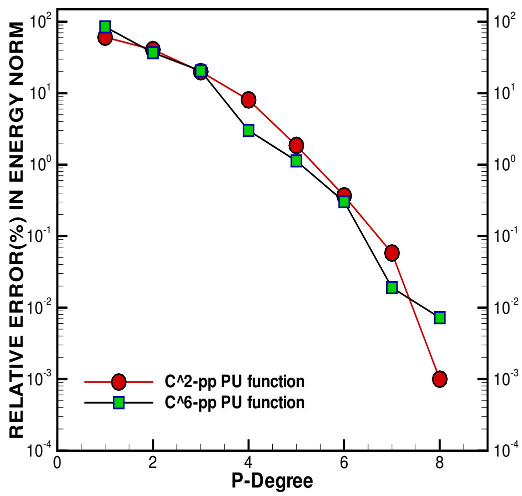

Figure 4: The relative errors(%) in the energy norm when the piecewise polynomial PU function $\phi_{g3}^{(pp)}$ and $\phi_{g7}^{(pp)}$, respectively, are used for the fourth order differential equation

6 The higher dimensional partition of unity functions

An obvious choice for higher dimensional PU functions is the tensor product of one dimensional PU functions constructed in section 3. Although simple patches, like squares and cubes, are used in many problems, we also need to use non uniform patches to deal with problems on non convex domains. The tensor product of PU functions does not fit for arbitrary patches. Thus, we consider the two dimensional PU functions on two different cases: rectangular patches and arbitrary quadrangular patches.

6.1 Two dimensional PU functions for uniform rectangular patches

Let us consider the piecewise polynomial consisting of four polynomials:

$$\Phi_{g_n}^{(4pp)}(\xi,\eta) = \phi_{g_n}^{(pp)}(\xi) \times \phi_{g_n}^{(pp)}(\eta).$$

Then $\Phi_{g_n}^{(4pp)}(\xi,\eta)$ have the following properties:

1. For all $(\xi,\eta) \in Q_0 = [-1,1]\times[-1,1]$,

$$\sum_{a=-1,0,1;b=-1,0,1} \Phi_{g_n}^{(4pp)}(\xi-a,\eta-b) = 1. \tag{62}$$

   That is, this is a piecewise polynomial PU function with support Q_0.

2. $\Phi_{g_n}^{(4pp)}(\xi,\eta) \in \mathcal{C}^{n-1}(\mathbb{R}^2)$ and is positive on $(-1,1)\times(-1,1)$. $\Phi_{g_n}^{(4pp)}(0,0) = 1$.

3. $\Phi_{g_n}^{(4pp)}(\xi,\eta)$ is symmetric about each coordinate axis, and it is composed of exactly four polynomials of degree $2n-1$ in each variable.

Like the one dimensional case, the two dimensional PU function $\Phi_{g_n}^{(4pp)}$ is uniquely determined by these properties. Thus, $\Phi_{g_n}^{(4pp)}(\xi,\eta)$ is a "simple" piecewise polynomial PU function. The product PU shape function $\Phi_{g_n}^{(4pp)}$ with $n=3$ is depicted in Fig. 5.

To test the effectiveness of $\Phi_{g_n}^{(4pp)}$ in GFEM, we consider the following two dimensional model problem:

Example 5. $u(x,y) = (1-x^2)(1-y^2)e^{x+y}$ solves the Poisson's equation

$$-\Delta u = f, \quad -1 \le x, y \le 1,$$

where $-f(x)$ is the Laplacian of the true solution.

For numerical solutions, we assume the followings:

- $h = 2/N, \quad (x_i, y_j) = (-1+hi, -1+hj), 0 \le i, j \le N.$

- The local approximation functions are the tensor product of Lagrange interpolating polynomials of degree 4:

$$L_{4,k}(\xi) \times L_{4,l}(\eta), 0 \le k, l \le 4.$$

- The PU function used in GFEM is $\Phi_{g_n}^{(4pp)}(\xi,\eta)$ for $n=3$.

Relative Error in energy norm in percent versus the diameter of the uniform patch in $\log\times\log$ scale is depicted in Fig. 5. The five relative errors in Fig. 5 are when the physical domain, $\Omega = [-1,1]\times[-1,1]$, is uniformly partitioned so that it can have $9(N=2), 16(N=3), 36(N=5), 49(N=6), 64(N=7)$ particles, respectively.

Note that the slope of the line in Fig. 5 is about 4 (Actually, the slopes of the lines connecting two adjacent squares in Fig. 5 are $3.95, 4.03, 4.08, 4.09$, respectively). This results are what we expect from the error estimate ([2]).

6.2 Two dimensional PU functions for arbitrary quadrangular patches

Suppose $\Omega \subset \mathbb{R}^2$ is a polygonal domain, $\Omega^\delta = \{(x,y) \in \mathbb{R}^2 : dist(\Omega,(x,y)) \le \delta\}$, and Ω^δ is partitioned into arbitrary quadrangular patches $Q_j, 1 \le j \le N$, so that

$$\text{each side of } Q_j \text{ is } \ge 2\delta \text{ and } \Omega^\delta = \textstyle\bigcup_{j=1}^N Q_j.$$

Then we have

$$\sum_{j=1}^{N} [\chi_{Q_j}(x,y)] = 1 \text{ almost everywhere in } \Omega.$$

Let the two dimensional normalized window function be defined by

$$w^{(l)}_{\delta\times\delta}(x,y) = w^{(l)}_{\delta}(x)w^{(l)}_{\delta}(y)$$

so that its support is the square

$$B_\delta = [-\delta,\delta]\times[-\delta,\delta] \text{ and } \int_{B_\delta} w^{(l)}_{\delta\times\delta}(x,y)dxdy = 1.$$

Then we have

$$\sum_{j=1}^{N}[\chi_{Q_j}] * [w^{(l)}_{\delta\times\delta}](x,y) \quad = \quad 1 \text{ for all } (x,y)\in\Omega.$$

Here " $*$ " is the convolution defined by

$$\int_{\mathbb{R}^2} \chi_{Q_j}(\xi,\eta)w^{(l)}_{\delta\times\delta}(x-\xi,y-\eta)d\xi d\eta. \tag{63}$$

If we write this convolution function by $\psi_j^{\delta\times\delta}$, then $\{\psi_j^{\delta\times\delta} : 1\le j\le N\}$ is a partition of unity subordinate to the covering $\{Q_j^\delta : 1\le j\le N\}$, where $Q_j^\delta = \{(x,y) : dist(Q_j,(x,y))\le\delta\}$.

Moreover, each convolution is a piecewise polynomial. However, the piecewise polynomial structure is complicated unless the quadrangle Q_j is a rectangle. We consider the convolution PU functions in the following two cases:

[The case when Q_j is a rectangle]

If Q_j is a rectangle, then $\psi_j^{\delta\times\delta}(x,y)$ is the tensor product of two convolution PU functions constructed in section 3.2, and hence it becomes a piecewise polynomial consisting of at most nine polynomials.

For example, let us consider the conical window function $w(x)$ with exponent $l=3$, in the window function. Let

$$w_s(x) = \frac{35}{32}w(x).$$

Then we have

$$\begin{aligned}
w^{(l)}_\delta(x) &= \frac{1}{\delta}w_s(\frac{x}{\delta}), \text{ where } l=3 \text{ is the exponent of (1)(not derivative order)},\\
F(x) &= \int_{-\infty}^{x} w_s(t)dt = \begin{cases} 0 & : \ x\le -1,\\ -\frac{1}{32}(x+1)^4\left(5x^3-20x^2+29x-16\right) & : \ -1\le x\le 1,\\ 1 & : \ x\ge 1\end{cases}\\
F_\delta(x) &= \int w^{(l)}_\delta(x)dx = F(\frac{x}{\delta}).
\end{aligned}$$

Let Q be a quadrangle. Then the convolution of χ_Q and the scaled window function $w^{(l)}_{\delta\times\delta}, l = 3$, is as follows:

$$\left(\chi_Q * w^{(l)}_{\delta\times\delta}\right)(x,y) = \int_{Q\cap[B_\delta+(x,y)]} w^{(l)}_{\delta\times\delta}(x-u,y-v)dudv,$$

which is a piecewise polynomial. Here $B_\delta + (x,y)$ is the translation of B_δ (the δ-box) by (x,y).

In particular, if $Q = [a,b]\times[c,d]$, one can show that the convolution PU function $\psi^{\delta\times\delta}_Q(x,y)$, with $l = 3$, can be explicitly represented by the following polynomials:

$$\psi^{\delta\times\delta}_Q(x,y) = \begin{cases} F_\delta(y-c) & : \ (x,y)\in[a+\delta,b-\delta]\times[c-\delta,c+\delta], \\ F_\delta(b-x)F_\delta(y-c) & : \ (x,y)\in[b-\delta,b+\delta]\times[c-\delta,c+\delta], \\ F_\delta(b-x) & : \ (x,y)\in[b-\delta,b+\delta]\times[c+\delta,d-\delta], \\ F_\delta(b-x)F_\delta(d-y) & : \ (x,y)\in[b-\delta,b+\delta]\times[d-\delta,d+\delta], \\ F_\delta(d-y) & : \ (x,y)\in[a+\delta,b-\delta]\times[d-\delta,d+\delta], \\ F_\delta(x-a)F_\delta(d-y) & : \ (x,y)\in[a-\delta,a+\delta]\times d-\delta,d+\delta], \\ F_\delta(x-a) & : \ (x,y)\in[a-\delta,a+\delta]\times[c+\delta,d-\delta], \\ F_\delta(x-a)F_\delta(y-c) & : \ (x,y)\in[a-\delta,a+\delta]\times[c-\delta,c+\delta], \\ 1 & : \ (x,y)\in[a+\delta,b-\delta]\times[c+\delta,d-\delta], \\ 0 & : \ (x,y)\notin Q \end{cases} \qquad (64)$$

[The case when Q_j is a non rectangular quadrangle]

Suppose Q_j is not a rectangle. Then the piecewise polynomial structure of $\psi^{\delta\times\delta}_j(x,y)$ is more complicated. As the minimal angle of a quadrangle Q_j gets smaller and lengths of sides become different, the piecewise polynomial structure becomes increasingly complicated. In this case, the convolution PU function consists of more than 12 distinct polynomials.

The computer code for $\psi^{\delta\times\delta}_j(x,y)$ and its derivatives can be found in ([18]). Numerical examples with respect to the convolution PU functions will be included in the forthcoming paper.

Consider a rectangle $Q_1 = [0,1]\times[0,2]$ and a non-rectangular quadrangle Q_2 whose four vertices are $(1,0),(2,1),(1.5,3),(1,2)$. The convolution PU functions on Q_1 and Q_2 with respect to $\delta = 0.1$ are depicted in Fig. 5, from which one can observe that the graphs are

$$\begin{cases} 1, & \text{if } (x,y)\in Q_i \text{ and } dist((x,y),\partial Q_i)\geq\delta, \\ 0, & \text{if } (x,y)\notin Q_i \text{ and } dist((x,y),\partial Q_i)\geq\delta, \end{cases}$$

for $i = 1,2$. Since $\|\nabla(\psi^{\delta\times\delta}_j)\|_{L_\infty(\mathbb{R}^2)}$ is $\mathcal{O}(1/\delta)$, by Theorem 6.1 of [2] and Theorem 4.1, the error in GFEM solution with respect to the convolution PU function is bounded by the sum of local approximation errors, unless δ is unrealistically small.

6.3 Condition numbers of stiffness matrices in GFEM

The condition numbers of stiffness matrices in GFEM depend on the choice of PU function and local approximation functions. However, they are generally very large (for example, see [21], [22]).

In this section, we briefly discuss the condition numbers resulted by using the proposed PU functions in GFEM. For this end, we consider the following local approximation functions.

1. (Lagrange local approximation functions)
   - Let $\xi_0 = -1, \xi_1 = -0.5, \xi_2 = 1, \xi_3 = 0.5, \xi_4 = 1$.
   - $$L_{4,k}(\xi) = \prod_{i=0, i\neq k}^{4} \frac{(\xi - \xi_i)}{(\xi_k - \xi_i)}, k = 0, 1, \cdots, 4.$$
   - Note that $[\phi_{g_n}^{(pp)} \times L_{4,k}], k = 0, 1, \cdots, 4$ are linearly independent, but they do not satisfy the Kronecker delta property. By taking tensor product, we obtain 25 two dimensional local approximation functions.
2. (Modified Lagrange local approximation functions)
   - Let $\bar{\xi}_0 = -0.75, \bar{\xi}_1 = -0.5, \bar{\xi}_2 = 1, \bar{\xi}_3 = 0.5, \bar{\xi}_4 = 0.75$.
   - Then the modified Lagrange interpolating function is defined by
   $$ML_{4,k}(\xi) = (1/C_k) \prod_{i=0, i\neq k}^{4} \frac{(\xi - \bar{\xi}_i)}{(\bar{\xi}_k - \bar{\xi}_i)}, k = 0, 1, \cdots, 4,$$
   where $C_k = \phi_{g_n}^{(pp)}(\bar{\xi}_k), k = 0, 1, \cdots, 4$.
   - Note that $[\phi_{g_n}^{(pp)} \times ML_{4,k}], k = 0, 1, \cdots, 4$ satisfy the Kronecker delta property and its support is $[-1, 1]$. By taking tensor product, we obtain 25 two dimensional local approximation functions related to the modified Lagrange interpolation polynomials.

Now we construct the GFEM approximation space spanned by 225 global basis functions constructed by parallel translation of these 25 local approximation functions to nine particles located at

$$\begin{array}{lll} (-1,0), & (0,0), & (1,0) \\ (-1,1), & (0,1), & (1,1) \\ (-1,-1), & (0,-1), & (1,-1) \end{array}$$

Consider Poisson's equation $\Delta u = f$ in $[-1, 1] \times [-1, 1]$. The condition numbers of stiffness matrices of size 225×225 for Galerkin approximation solutions of this equation are shown in

Table 5: The condition numbers when the approximation functions are (the simple PU function)×(Lagrange interpolating function), (the simple PU function)×(modified Lagrange interpolating function), and (the convolution PU function)×(Lagrange interpolating function), respectively

	$\Phi_{g_3}^{(4pp)}\times$ Lagrange	$\Phi_{g_3}^{(4pp)}\times$ modified Lagrange	Convol PU $\times$ Lagrange
Cond Number	3.30E+14	4.18E+11	1.22E+06

the first and the second column of Table 5. The third condition number is obtained by using the convolution PU function in GFEM, where the domain is partitioned into 12 rectangular patches. In this case, the dimension of the GFEM approximation space is 300.

Table 5 shows that the condition number when the convolution PU function is used is much smaller than that when the simple PU function is used.

An optimal choice of PU function and local approximation functions for a smaller condition number will be discussed in the forthcoming paper.

Acknowledgements

We are very grateful to referees and Professor John Osborn, editor of the special volumes honoring Professor Ivo Babuška's 80th birthday. They provided various constructive suggestions and valuable references to enhance this paper.

Appendix

A Properties of B-splines

The B-splines, defined by 3.1, have the following properties ([10]):

1. $b_n \in \mathcal{C}^{n-1}(\mathbb{R})$ is positive on $(0, n+1)$ and vanishes outside this interval.
2. b_n is a piecewise polynomial: it is a polynomial of degree n on each interval $[k, k+1], k = 0, \cdots, n$.
3. The B-spline of degree n is symmetric about $y = (n+1)/2$.
4. (Recurrence Relation)

$$b_n(x) = \frac{x}{n} b_{n-1}(x) + \frac{n+1-x}{n} b_{n-1}(x-1). \tag{65}$$

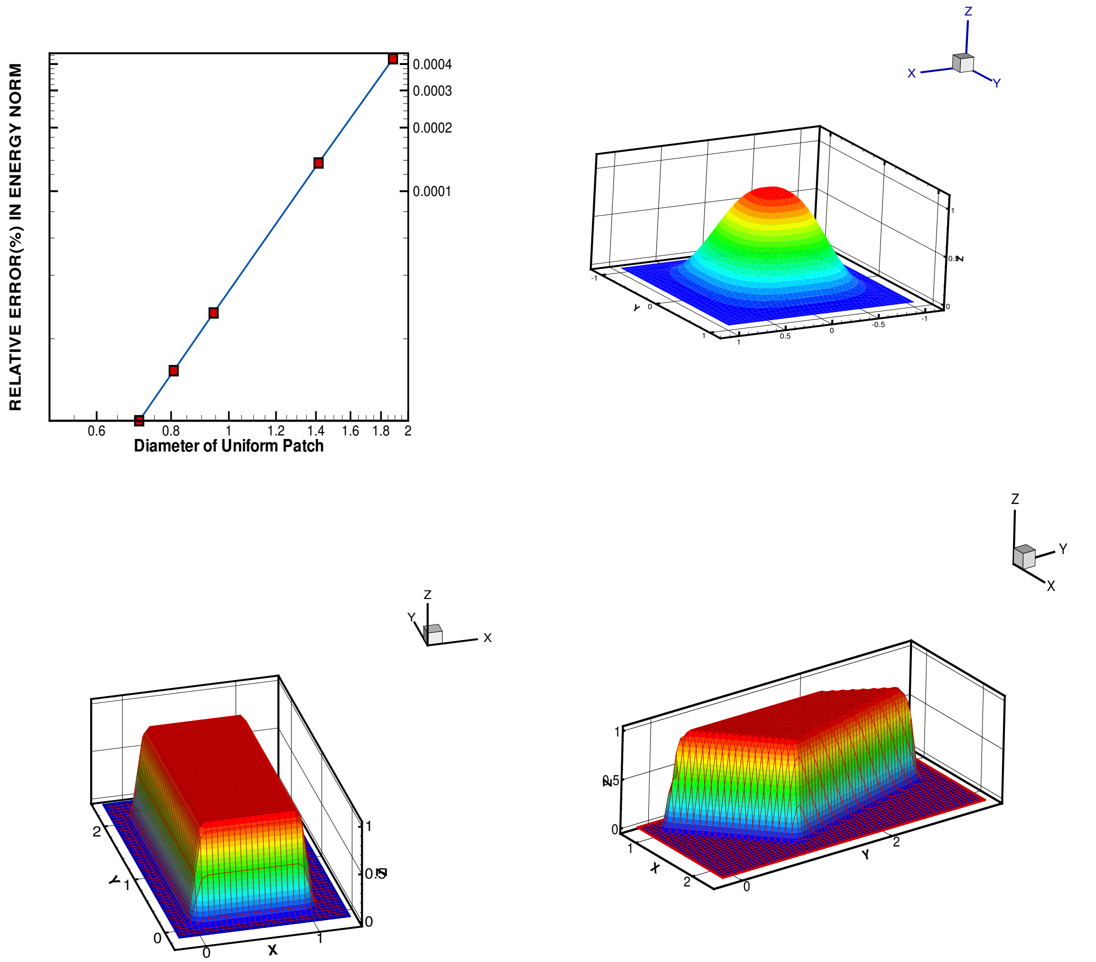

Figure 5: (Top Left) Percentage Relative Errors of GFEM solution of Example 5 in energy norm versus the diameters of uniform patches; (Top Right) Two dimensional PU function $\phi_{g3}^{(pp)}(x,y) = [\Phi_{g3}^{(4pp)}](x) \times [\phi_{g3}^{(pp)}](y)$; (Bottom Left) The convolution PU functions on the rectangle $Q_1 : (0,0) \to (1,0) \to (1,2) \to (0,2)$; (Bottom Right) The convolution PU functions on the quadrangle $Q_2 : (1,0) \to (2,1) \to (1.5,3) \to (1,2)$

5. (Convolution Property)

$$b_{n+m+1}(x) = \int_{\mathbb{R}} b_m(x-y) b_n(y) dy. \tag{66}$$

6. (Marsden's Identity) For $x, t \in \mathbb{R}$,

$$(x-t)^n \quad = \quad \sum_{k \in \mathbb{Z}} (k+1-t) \cdots (k+n-t) b_n(x-k). \tag{67}$$

   From Marsden identity, we have

$$\sum_k b_n(x-k) = 1, \tag{68}$$

   which proves that b_n is a PU shape function. Moreover, the same identity can be applied to show the next property:

7. Any monomial $x^l, 0 \leq l \leq n$ can be represented by $b_n(x-k), k \in \mathbb{Z}$, and hence, they are linearly independent.

In contrast to the B-splines, the RPP shape functions have the following features:

- The RPP shape functions are not always B-splines. However, they satisfy almost all of the properties of B-splines, except the positivity and the convolution property.

- The RPP shape functions satisfy the Kronecker delta property; and hence the RPP shape functions can handle essential boundary conditions more effectively than the B-spline basis functions. Moreover, by using the Kronecker delta property, one can show that the RPP shape functions are linearly independent.

- The monomials $x^k, k \leq m$, are exactly interpolated by the RPP shape functions with reproducing property of order m. In other words, Definition (2) implies that, for all $x, t \in \mathbb{R}$,
$$(x-t)^m = \sum_{k \in \mathbb{Z}} (k-t)^m \phi_{[(a,b);r;m]}(x-k).$$
  that corresponds to Marsden's indentity (67).

- Unlike B-spline, by increasing the degree of polynomial as shown in examples of section 3, the regularities of RPP shape functions can go as high as we wish without increasing the number of distinct polynomials.

- The RPP shape functions do not have the convolution property (66) that enable B-splines to have the simple exact integration.

B Orthogonal Polynomials with respect to the PU weight functions

Using Theorem 3 of section 3.7 of ([20]), we can construct a sequence of orthogonal polynomials for the given weight function as follows.

Let $P_n(x), n = 1, 2, 3, ...,$ be the sequence of orthogonal polynomials on $[-1, 1]$ for the weight $W(x)$. The orthogonal polynomials are denoted as

$$P_n(x) = x^n + a(n, n-1)x^{n-1} + a(n, n-2)x^{n-2} +, ..., +a(n, 1)x + a(n, 0).$$

Then the coefficients are determined by the following recursion formula.

$$\begin{aligned}
a(1,1) &= 1.0 \\
a(1,0) &= -\int_{-1}^{1} W(x) \cdot x dx / \int_{-1}^{1} W(x) dx \\
Q(n, n-1) &= \int_{-1}^{1} W(x) \cdot x^n \left[\sum_{k=1}^{n} a(n-1, n-k) \cdot x^{n-k} \right] dx \\
Q(n-1, n-1) &= \int_{-1}^{1} W(x) \cdot x^{n-1} \left[\sum_{k=1}^{n} a(n-1, n-k) \cdot x^{n-k} \right] dx \\
Q(n-2, n-2) &= \int_{-1}^{1} W(x) \cdot x^{n-2} \left[\sum_{k=2}^{n} a(n-2, n-k) \cdot x^{n-k} \right] dx \\
\beta(n) &= Q(n, n-1)/Q(n-1, n-1) + a(n-1, n-1) \\
\gamma(n) &= Q(n-1, n-1)/Q(n-2, n-2)
\end{aligned}$$

$$\begin{aligned}
a(n, n) &= 1.0 \\
a(n, n-1) &= a(n-1, n-2) - \beta(n) \\
a(n, n-2) &= a(n-1, n-3) - \beta(n) \cdot a(n-1, n-2) - \gamma(n) \\
a(n, k) &= a(n-1, k-1) - \beta(n) \cdot a(n-1, k) - \gamma(n) \cdot a(n-2, k), \quad n-3 \geq k \geq 1 \\
a(n, 0) &= -\beta(n) \cdot a(n-1, 0) - \gamma(n) \dot{a}(n-2, 0)
\end{aligned}$$

Using this recursion formula, we constructed two sets of the orthogonal polynomials for $W = [\phi_{g3}^{(pp)}]^2$ and $W = (\frac{d}{dx}[\phi_{g3}^{(pp)}])^2$, respectively, up to degree 7 in Table 9.

Table 6: The orthogonal polynomials $P_n(x) = \sum_{k=0}^{n} a(n, n-k) \cdot x^{n-k}$ for weight function $W(x)$ and $[-1, 1]$. Obviously, $P_0(x) = 1, P_1(x) = x$.

n	$n-k$	$W(x) = ([\phi_{g_3}^{(pp)}](x))^2$	$W(x) = ([\phi_{g_3}^{(pp)}]'(x))^2$
2	2	1.0	1.0
2	1	0.0	0.0
2	0	-0.0757897719223721	-0.2153110047846890
3	3	1.0	1.0
3	2	0.0	0.0
3	1	-0.1872897196261682	-0.3179487179487180
3	0	0.0	0.0
4	4	1.0	1.0
4	3	0.0	0.0
4	2	-0.3297528177647252	-0.5654633835365361
4	1	0.0	0.0
4	0	0.0107972457152757	0.0532926313466594
5	5	1.0	1.0
5	4	0.0	0.0
5	3	-0.4911333369260064	-0.7145191409897305
5	2	0.0	0.0
5	1	0.0410221579021176	0.1006847183317776
5	0	0.0	0.0
6	6	1.0	1.0
6	5	0.0	0.0
6	4	-0.6685440421663305	-0.9734638846781050
6	3	0.0	0.0
6	2	0.0995238378567415	0.2471084892468069
6	1	0.0	0.0
6	0	-0.0019155469770001	-0.0137998467645397
7	7	1.0	1.0
7	6	0.0	0.0
7	5	-0.8569568219056297	-1.1479067026963812
7	4	0.0	0.0
7	3	0.1920596350896082	0.3717512217290534
7	2	0.0	0.0
7	1	-0.0096446457782426	-0.0313635727617114
7	0	0.0	0.0

References

[1] Atluri, S. and Shen, S.: The Meshless Method, Tech Science press, 2002.

[2] Babuska, I., Banerjee, U., Osborn, J.E.: *Survey of meshless and generalized finite element methods:A unified appraoch,* Acta Numerica, (2003) 1-125 .

[3] Babuska, I., Banerjee, U., Osborn, J.E.: *Generalized finite element methods:Main Ideas, Results, and Perspectives,* Int. J. of Computational Methods, Vol. 1 (2004) 67-103.

[4] Babuska, I., Banerjee, U., Osborn, J.E.: *On the approximability and the selection of particle shape functions,* Numer. math. 96 (2004) 601-640.

[5] Babuska, I., Banerjee, U., Osborn, J.E.: *On principles for the selection of shape functions for Generalized Finite Element Method,* Comput. Methods Appl. Mech. Engrg. 191 (2002) 5595-5629.

[6] Ciarlet, P.G. : *Basic Error Estimates for Elliptic Problems,* Hanbook of Numerical Analysis, Vol II, North Hollnad, 1991.

[7] Duarte, C.A. and Oden, J.T.: *Hp clouds-a meshless method to solve boundary vale problems,* Technical Report 95-05, TICAM, The University of Texas as Austin, May 1995.

[8] Duarte, C.A. and Oden, J.T.: *An hp adaptive method using clouds,* Comput. Methods Appl. Mech. Engrg., Vol. 139 (1996) 237-262.

[9] Han, W. and Meng, X. :*Error alnalysis of reproducing kernel particle method*, Comput. Methods Appl. Mech. Engrg. 190 (2001) 6157-6181.

[10] Höllig, K : Finite Element methods with B-Spline, SIAM, 2003.

[11] Lancaster, P. and K. Salkauskas :*Surfaces Generated by Moving Least Squares methods*, Math. of Com., 37, pp141-158 (1981).

[12] Levin, D. :*The Approximation Power of Moving Least Squares*, Math. of Com., 67 (1998) 1517-1531 .

[13] Li, S. and Liu, W.K. : Meshfree Particle Methods, Springer-Verlag 2004.

[14] Li, S., Lu, H., Han, W., Liu, W.K., and Simkins, D.C.Jr. :*Reproducing Kernel Element Method: Part II. Globally Conforming I^m/C^n hierarchies,* Comput. Methods Appl. Mech. Engrg., Vol. 193 (2004) 953-987.

[15] Liu, W.K., Han, W., Lu, H., Li, S., and Cao, J. :*Reproducing Kernel Element Method: Part I. Theoretical formulation,* Comput. Methods Appl. Mech. Engrg., Vol. 193 (2004) 933-951.

[16] Melenk, J.M. and Babuška I. :*The partition of unity finite element method:Theory and application* , Comput. Methods Appl. Mech. Engrg. 139 (1996) 239-314.

[17] Oh, H. S., Kim, J.G., and Jeong, J. W. :*The Closed Form Reproducing polynomial particle shape functions for meshfree particle methods,* Comput. Methods Appl. Mech. Engrg. 196 (2007) 3435-3461

[18] Oh, H. S., Kim, J.G., and Jeong, J. W. :*The smooth piecewise polynomial particle shape functions corresponding to path-wise non uniformly spaced particles for meshfree particle methods,* Comput Mech 40 (2007) 569-594

[19] Oh, H.-S., Jeong, J.W., and Kim, J. G. :*The Reproducing Singularity Particle Shape Functions for Problems Containing Singularities* , Comput Mech DOI 10.1007/s00466-007-0174-x (2007)

[20] Stroud, A.H. :*Numerical Quadrature and Solution of Ordinary Differential Equations,* Springer-Verlag, 1974

[21] Stroubolis, T., K. Copps, and I. Babuska: *Generalized Finite Element method*, Comput. Methods Appl. Mech. Engrg., 190 (2001) 4081-4193.

[22] Stroubolis, T., L. Zhang, and I. Babuska: *Generalized Finite Element method using mesh-based handbooks:application to problems in domains with many voids*, Comput. Methods Appl. Mech. Engrg., 192 (2003) 3109-3161.

[23] Szabo, B. and Babuska, I. , Finite Element Analysis, John Wiley, 1991.