Journal of Computational Physics 229 (2010) 1600–1620

Contents lists available at ScienceDirect

Journal of Computational Physics

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

The generalized product partition of unity for the meshless methods

Hae-Soo Oh [a,*,1], Jae Woo Jeong [b], Won Tak Hong [a]

[a] Department of Mathematics and Statistics, University of North Carolina at Charlotte, Charlotte, NC 28223, United States
[b] Department of Mathematics and Statistics, Miami University, Hamilton, OH 45011, United States

ARTICLE INFO

Article history:
Received 18 April 2009
Received in revised form 30 July 2009
Accepted 28 October 2009
Available online 1 November 2009

Keywords:
Meshless methods
Convolution partition of unity functions
Partition of unity function with flat-top
Patchwise reproducing polynomial particle method (RPPM)
Product partition of unity

ABSTRACT

The partition of unity is an essential ingredient for meshless methods named by GFEM, PUFEM (partition of unity FEM), XFEM (extended FEM), RKPM (reproducing kernel particle method), RPPM (reproducing polynomial particle method), the method of *hp* clouds in the literature. There are two popular choices for partition of unity: a piecewise linear FEM mesh and the Shepard-type partition of unity. However, the partition of unity (PU) by a FEM mesh leads to the singular (or nearly singular) matrices and non-smooth approximation functions. The Shepard-type partition of unity requires lengthy computing time and its implementation is difficult. In order to alleviate these difficulties, Oh et al. introduced the smooth piecewise polynomial PU functions with flat-top, that lead to small matrix condition numbers, and almost everywhere partition of unity, that can handle essential boundary conditions. Nevertheless, we could not have the smooth closed form PU functions with flat-top for general polygonal patches (2D) and general polyhedral patches (3D). In this paper, we introduce one of the most simple and efficient partition of unity, called *the (generalized) product partition of unity*. The product PU functions constructed by this method are the closed form smooth piecewise polynomials with flat-top and could handle background meshes (general polygonal patches as well as general polyhedral patches) arising in practical applications of meshless methods.

1. Introduction

Meshless methods [1,2,4,12–18,27,28] have several advantages over the conventional finite element method [3,29]. However, they have some difficulties such as large matrix condition numbers (or singular stiffness matrix), complicated (or non-smooth) partition of unity (PU) functions, ineffectiveness in handling essential boundary conditions, lengthy computing time due to complicated PU functions, and so forth.

In order to alleviate these difficulties, Oh et al. [21–26] introduced patchwise RPPM (reproducing polynomial particle method) with use of the convolution PU functions with flat-top. It was shown in [23] that the PU function with flat-top lead to the small matrix condition number. However, it is not easy to extend the two-dimensional construction of the convolution PU functions to the three-dimensional cases. Oh et al. [23] constructed smooth piecewise polynomial PU functions with flat-top in one-dimensional case. Obviously, the tensor product of these one-dimensional PU functions yields higher dimensional PU functions with flat-top. However, tensor products of intervals cannot make neither triangular, general quadrangular patches (2D), nor tetrahedral, pentahedral, general hexahedral patches, arising in background meshes for meshless methods.

* Corresponding author. Tel.: +1 704 687 4930; fax: +1 704 687 6415.
E-mail address: hso@uncc.edu (H.-S. Oh).
[1] Supported in part by NSF Grant DMS-07-13097.

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.10.047

In this paper, we introduce a (two and three dimensional) unified method constructing a partition of unity associated with background meshes. We call this simple method *the generalized product method*. The procedure of this method is as follows:

- First, we construct a partition of unity on $\mathbb{R}$ that consists of two simple smooth piecewise polynomials that look like the step functions.
- Second, through the coordinate projection from $\mathbb{R}^d,\ d = 2, 3,$ onto $\mathbb{R}$, we construct the partition of unity on $\mathbb{R}^d$ that consists of two smooth PU functions with flat-top. Via the proper affine transformations, two PU functions are copied onto each side (in 2-Dim case), or each face (in 3-Dim case), of a patch, say a polygonal (or polyhedral) patch Q. Then, exactly one of two PU functions planted on each face is identically 1 on most part of the patch Q (called the flat-top part of Q).
- Finally, The product of those PU functions corresponding to the sides (faces) of Q that are "1" on the flat-top part of Q become a smooth PU function with flat-top corresponding to Q.

After introducing some preliminary results in Section 2, the generalized product method to construct smooth partition of unity is introduced in Section 3. In the same section, we also prove that the generalized product method yields a partition of unity on a domain $\Omega \subset \mathbb{R}^2$ as well as a domain $\Omega \subset \mathbb{R}^3$. In order to show the effectiveness of this method, two-dimensional and three-dimensional numerical tests are carried out in Section 4.

We claim that the generalized product method constructing a partition of unity makes meshless methods much more useful.

2. Preliminary

2.1. Definitions

In this section, we introduce definitions and terminologies that are used throughout this paper.

Let $\overline{\Omega}$ is the closure of $\Omega \subset \mathbb{R}^d$, we define the vector space $\mathscr{C}(\overline{\Omega})$ to consist of all those functions $\phi \in \mathscr{C}^m(\Omega)$ for which $D^\alpha\phi$ is bounded and uniformly continuous on Ω for $|\alpha| = \alpha_1 + \cdots + \alpha_d \leqslant m$. In the following, a function $\phi \in \mathscr{C}^m(\Omega)$ is said to be a $\mathscr{C}^m$-function. If Ψ is a function defined on Ω, we define the *support* of Ψ as

$$\mathrm{supp}\Psi = \overline{\{x \in \Omega | \Psi(x) \neq 0\}}.$$

For an integer $k \geqslant 0$, 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

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

Furthermore, the maximum norm of a function u defined on Ω is defined by

$$\|u\|_{\infty,\Omega} = \max\{|u(x)| \, x \in \Omega\}.$$

A family $\{U_k | k \in \mathscr{D}\}$ 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 an integer 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 $\{U_k | k \in \mathscr{D}\}$ of a domain Ω, suppose there is a family $\{\phi_k | k \in \mathscr{D}\}$ of Lipschitz functions on Ω satisfying the following conditions:

1. For $k \in \mathscr{D},\ 0 \leqslant \phi_k(x) \leqslant 1,\ x \in \mathbb{R}^d$.
2. The support of ϕ_k is contained in $\overline{U}_k$, for each $k \in \mathscr{D}$.
3. $\sum_{k \in \mathscr{D}} \phi_k(x) = 1$ for each $x \in \Omega$.

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

By *almost everywhere partition of unity*, we mean $\{\phi_k \, k \in \mathscr{D}\}$ such that the condition 3 of a partition of unity is not satisfied only at finitely many points (2D) or lines (3D) on a part of the boundary.

Let $\omega = \mathrm{supp}(\phi)$. Then $\omega^{flt} = \{x \in \omega | \phi(x) = 1\}$ and $\omega^{nflt} = \overline{\{x \in \omega | 0 \leqslant |\phi(x)| < 1\}}$ are called the *flat-top* part and the *non-flat-top* part of ω, respectively. The function ϕ is said to be *a function with flat-top* if $\omega^{flt} \neq \emptyset$. Moreover, $\{\phi_k | k \in \mathscr{D}\}$ is called a *partition of unity with flat-top* whenever it is partition of unity and ϕ_k is a function with flat-top for each $k \in \mathscr{D}$.

Notice that if $f_1, \ldots, f_n$ are linearly independent on $\omega^{flt} \neq \emptyset$, the product functions, $\phi \cdot f_1, \ldots, \phi \cdot f_n$, are also linearly independent on ω. However, if $\omega^{flt} = \emptyset$, the product functions, $\phi \cdot f_1, \ldots, \phi \cdot f_n$, could be linearly dependent.

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

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

where l is an integer. Then $w(x)$ is a $\mathscr{C}^{l-1}$-function. In $\mathbb{R}^d$, the weight function $w(x_1,\ldots,x_d)$ can be constructed from a one-dimensional weight function as $w(x_1,\ldots,x_d)=\prod_{i=1}^{d}w(x_i)$.

In this paper, we use the normalized window function defined by

$$w_\delta^l(x)=Aw\left(\frac{x}{\delta}\right), \tag{2}$$

where $A=[(2l+1)!]/[2^{2l+1}(l!)^2\delta]$ [6] is the constant that makes $\int_{\mathbb{R}}w_\delta^l(x)dx=1$.

Let Λ be a finite index set and Ω denotes a bounded domain in $\mathbb{R}^d$. Let $\{x_j|j\in\Lambda\}$ be a set of a finite number of uniformly or non-uniformly spaced points in $\mathbb{R}^d$, that are called *particles*.

Definition 2.1. Let k be a non-negative integer. Then the functions $\phi_j(x)$ corresponding to the particles $x_j,\ j\in\Lambda$ are called the RPP (reproducing polynomial particle) shape functions with the reproducing property of order k (or simply, "of reproducing order k") if and only if they satisfy the following condition:

$$\sum_{j\in\Lambda}(x_j)^\alpha\phi_j(x)=x^\alpha,\quad \text{for } x\in\Omega\subset\mathbb{R}^d \quad \text{and for } 0\leqslant|\alpha|\leqslant k. \tag{3}$$

Note that the RPP shape functions $\phi_j,\ j\in\Lambda$, of reproducing order k can exactly interpolate polynomials of degree $\leqslant k$.

2.2. Methods of constructing partition of unity

A partition of unity is an essential ingredient of meshless methods. Several methods constructing partition of unity have been suggested in the literature. The popular partitions of unity are the nodal shape functions corresponding to a Finite Element mesh [18,27,28], the particle-partition of unity [5], and the convolution partition of unity [12,22,23,25]. The particle-partition of unity is also known as the Shepard functions. A variant of this is the signed partition of unity for the *hp* clouds [4]. These partition of unity have the following salient features:

A: The Nodal shape functions of a finite element space:

- (Use background mesh for the construction). Let $\mathscr{T}$ be a simple finite element mesh of $\Omega\subset\mathbb{R}^d$. Let φ_α be the nodal shape functions (the hat functions) corresponding to the nodes x_α in the mesh $\mathscr{T}, \alpha=1,\ldots,N$, respectively. Then $\{\varphi_\alpha|\alpha=1,\ldots,N\}$ is a partition of unity.
- (Lower regularity) The hat functions φ_α are $\mathscr{C}^0$ functions.
- (Large Matrix condition number and no flat-top). These PU functions have no flat-top. Therefore, even though, $f_k^\alpha, k=1,\ldots,N_\alpha$, are linearly independent local approximation functions on $\omega_\alpha=\text{supp}(\varphi_\alpha)$, the product functions, $\varphi_\alpha\cdot f_k^\alpha, k=1,\ldots,N_\alpha$, may not be linearly independent. Hence, the associated stiffness matrix is singular or nearly singular.
- (Simple numerical integration). The overlapping parts of supports of any two PU functions are members of the background mesh. Integration is similar to that of the conventional FEM.
- (Closed form functions). The PU functions are of closed form.
- (Imposing essential BC is difficult). Imposing essential boundary conditions is difficult. The penalty methods, the Lagrange multiplier methods, and the Nietche's method are suggested.

B: The particle-partition of unity:

- (Use particles for the construction and no background mesh are needed). Instead of a background mesh, particles $x_1,\ldots,x_N$ are planted in the domain $\Omega\subset\mathbb{R}^d$. The scaled window functions $w_{h_\alpha}^l(x-x_\alpha)$, defined by (2) with various radius h_α, are constructed at the particles $x_\alpha,\ \alpha=1,\ldots,N$. Here, the locations of particles and the sizes of radius h_α are determined so that $\Omega\subset\cup_{\alpha=1}^N\omega_\alpha$, where ω_α denotes the support of $w_{h_\alpha}^l(x-x_\alpha)$. For each $\alpha=1,\ldots,N$, let $\varphi_\alpha(x)=w_{h_\alpha}^l(x-x_\alpha)/S_\alpha(x)$, where $S_\alpha(x)$ is the sum of the window functions $w_{h_\beta}^l(x-x_\beta)$ such that $\omega_\alpha\cap\omega_\beta$ is non-empty. Then $\{\varphi_\alpha|\alpha=1,\ldots,N\}$ is a partition of unity subordinate to the covering $\{\omega_\alpha\}$.
- (Highly regular). The Shepard PU functions $\varphi_\alpha(x)$ are $\mathscr{C}^{l-1}$.
- (Large matrix condition number and no flat-top in general). Even though $f_k^\alpha,\ k=1,\ldots,N_\alpha$, are linearly independent local approximation functions on ω_α, the product functions, $\varphi_\alpha\cdot f_k^\alpha,\ k=1,\ldots,N_\alpha$, may not be linearly independent. Thus, the matrix condition number could be very large. However, it is possible to select the particles and radii of patches ω_α so that φ_α become functions with non-empty flat-top ($\omega_\alpha^{flt}\neq 0$). Then, the product functions, $\varphi_\alpha\cdot f_k^\alpha,\ k=1,\ldots,N_\alpha$, are linearly independent whenever the local approximation functions f_k^α are linearly independent on ω_α^{flt} (see the flat-top condition of [5]) .
- (Very complicated numerical integration). The overlapping parts of supports of any two PU functions could have lens shapes and the computation of stiffness matrix is expensive.
- (Closed form functions). The PU functions are complicated rational functions (the signed partition of unity for *hp* clouds that is a generalized version of the particle PU, have no closed form). In the higher dimensional cases, tracking of overlapping parts of PU functions for the computation of stiffness matrices are more difficult than the conventional finite element meshes.

- (Imposing essential BC is difficult). Imposing essential boundary conditions is difficult. The penalty methods, the Lagrange multiplier methods, and the Nietche's method are suggested.

C: The convolution partition of unity:

- (Use background mesh for the construction). Oh et al. [21–23,25] introduced the convolution method constructing a partition of unity in which all PU functions are piecewise polynomials with wide flat-top. It is important to note that the background mesh for the convolution PU is not a FEM mesh, but a simple subdivision of Ω into disjoint polygons (polyhedrons). Let $\{\Omega_\alpha | \alpha = 1, \dots, N\}$ be a subdivision of the domain $\Omega(\subset \mathbb{R}^d)$ into patches (hanging nodes are allowed) so that

$$\sum_{\alpha=1}^{N} \chi_{\Omega_\alpha}(x) = 1, \quad \text{for all } x \in \Omega \text{ except those points on } \partial\Omega_\alpha$$

  and δ be a fixed small number. Suppose, for each $\alpha = 1, \dots, N$, $\varphi_\alpha = \chi_{\Omega_\alpha} * w_\delta^l$ is the convolution of characteristic function of patch Ω_α and the scaled window function, then $\{\varphi_\alpha\}$ is a partition of unity subordinate to a covering $\{\omega_\alpha\}$, where $\omega_\alpha = \{x | \mathrm{dist}(x, \Omega_\alpha) \leqslant \delta\} = \mathrm{supp}(\varphi_\alpha)$.
- (Highly regular). The convolution PU functions $\varphi_\alpha(x)$ are $\mathscr{C}^{l-1}$-functions.
- (Small matrix condition number because of the flat-top condition). If δ is small, the convolution PU functions φ_α have wide flat-top, and hence the product functions, $\varphi_\alpha \cdot f_k^\alpha,\ k = 1, \dots, N_\alpha$, are linearly independent whenever the local approximation functions $f_k^\alpha,\ k = 1, \dots, N_\alpha$, are linearly independent on ω_α^{flt}, the flat-top part. Thus, the matrix condition number is small in general.
- (Simple numerical integration). The overlapping parts of supports of any two PU functions is a quadrangle and can be automatically detected. Thus, since φ_α are piecewise polynomial, the Gaussian quadrature yield exact integrals for the product functions $\varphi_\alpha \cdot f_k^\alpha$, if the local approximation functions f_k^α are polynomials.
- (Not closed form functions). The closed form of the convolution PU functions are not available in general. However, the computer code to generate the convolution PU functions for two-dimensional case can be found in www.math.uncc.edu/~hso/. Moreover, the flat-top parts and non-flat-top parts can be easily determined for numerical integrals of convolution PU functions.
- (Imposing essential BC is simple). Imposing essential boundary conditions is as simple as that of the conventional FEM.
- (Difficulties). The closed form smooth PU functions for general polygonal patches (2D) and general polyhedral patches(3D) have not been available yet.

Most of popular PU for the meshless methods have some difficulties as listed above. In this paper, we introduce a new efficient PU that alleviate these difficulties arising in popular partition of unity.

3. One-dimensional partition of unity functions

In this section, we briefly review one-dimensional partition of unity (PU) with flat-top. For details of this construction, we refer to Oh et al. [23], in which we showed that PU functions with flat-top lead to a small matrix condition number.

Throughout this paper, we reserve the small real number δ, usually, $0.01 \leqslant \delta \leqslant 0.1$, for the width of non-flat-top part of the PU functions.

3.1. One-dimensional partition of unity functions without flat-top

For any positive integer $n, \mathscr{C}^{n-1}$-piecewise polynomial basic PU functions are constructed as follows: for integers $n \geqslant 1$, we define a piecewise polynomial function by

$$\phi_{g_n}^{(pp)}(x) = \begin{cases} \phi_{g_n}^L(x) := (1+x)^n g_n(x), & \text{if } x \in [-1, 0], \\ \phi_{g_n}^R(x) := (1-x)^n g_n(-x), & \text{if } x \in [0, 1], \\ 0, & \text{if } |x| \geqslant 1, \end{cases} \tag{4}$$

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

$$a_k^{(n)} = \begin{cases} 1, & \text{if } k = 0, \\ \sum_{j=0}^{k} a_j^{(n-1)}, & \text{if } 0 < k \leqslant n-2, \\ 2(a_{n-2}^{(n)}), & \text{if } k = n-1. \end{cases} \tag{5}$$

Using the recurrence relation (5), $g_n(x)$ is as follows:

$$
\begin{aligned}
&g_1(x) = 1,\\
&g_2(x) = 1 - 2x,\\
&g_3(x) = 1 - 3x + 6x^2,\\
&g_4(x) = 1 - 4x + 10x^2 - 20x^3,\\
&g_5(x) = 1 - 5x + 15x^2 - 35x^3 + 70x^4,\\
&\ \vdots \quad \vdots \qquad\quad \vdots
\end{aligned}
$$

Then, $\phi_{g_n}^{(pp)}$ has the following properties whose proofs can be found in [23].

- $\phi_{g_n}^{(pp)}(x) + \phi_{g_n}^{(pp)}(x-1) = 1$ for all $x \in [0,1]$ and $0 \leqslant \phi_{g_n}^{(pp)}(x) \leqslant 1$, for all $x \in \mathbb{R}$. Hence, $\left\{\phi_{g_n}^{(pp)}(x-j) | j \in \mathbb{Z}\right\}$ is a partition of unity on $\mathbb{R}$.
- $\phi_{g_n}^{(pp)}(x)$ is a $\mathscr{C}^{n-1}$-function.
- The gradient of the scaled basic PU function is bounded as follows:

$$\frac{d}{dx}\left[\phi_{g_n}^{(pp)}\left(\frac{x}{2\delta}\right)\right] \leqslant \frac{C}{\delta}, \tag{6}$$

where the constant C is $\leqslant 0.9$ for $n \leqslant 3$.

3.2. One-dimensional convolution PU with flat-top

Using the basic PU function $\phi_{g_n}^{(pp)}$ defined by (4), we construct a $\mathscr{C}^{n-1}$-PU function with flat-top whose support is $[a-\delta, b+\delta]$ with $(a+\delta) < (b-\delta)$ in a closed form as follows:

$$\psi_{[a,b]}^{(\delta,n-1)}(x) = \begin{cases} \phi_{g_n}^{L}\left(\frac{x-(a+\delta)}{2\delta}\right), & \text{if } x \in [a-\delta, a+\delta],\\ 1, & \text{if } x \in [a+\delta, b-\delta],\\ \phi_{g_n}^{R}\left(\frac{x-(b-\delta)}{2\delta}\right), & \text{if } x \in [b-\delta, b+\delta],\\ 0, & \text{if } x \notin [a-\delta, b+\delta]. \end{cases} \tag{7}$$

Here, in order to make a PU function have a flat-top, we assume $\delta \leqslant (b-a)/3$.

Since the two functions $\phi_{g_n}^{R}$, $\phi_{g_n}^{L}$, defined by (4), satisfy the following relation:

$$\phi_{g_n}^{R}(\xi) + \phi_{g_n}^{L}(\xi - 1) = 1, \quad \text{for } \xi \in [0,1], \tag{8}$$

if $\varphi\,[-\delta,\delta] \to [0,1]$ is defined by

$$\varphi(x) = (x+\delta)/(2\delta),$$

then we have

$$\phi_{g_n}^{R}(\varphi(x)) + \phi_{g_n}^{L}(\varphi(x) - 1) = 1, \quad \text{for } x \in [-\delta,\delta].$$

Using the latter equation gives two basic one-dimensional $\mathscr{C}^{n-1}$ functions

$$\psi_0^R(x) = \begin{cases} 1, & \text{if } x \leqslant -\delta,\\ \phi_{g_n}^{R}\left(\frac{x+\delta}{2\delta}\right), & \text{if } x \in [-\delta,\delta],\\ 0, & \text{if } x \geqslant \delta, \end{cases} \tag{9}$$

$$\psi_0^L(x) = \begin{cases} \phi_{g_n}^{L}\left(\frac{x-\delta}{2\delta}\right), & \text{if } x \in [-\delta,\delta],\\ 1, & \text{if } x \geqslant \delta,\\ 0, & \text{if } x \leqslant -\delta \end{cases} \tag{10}$$

such that

$$0 \leqslant \psi_0^L(x), \psi_0^R(x) \leqslant 1, \quad \psi_0^R(x) + \psi_0^L(x) = 1, \quad \text{for all } x \in \mathbb{R}. \tag{11}$$

Let us consider the following background mesh on the domain $\Omega = (A,B) \subset \mathbb{R}$ for the construction of the convolution partition of unity with wide flat-top:

$$A < A + \delta \leqslant k_1 < k_2 < \cdots < k_{N-1} < k_N + \delta \leqslant B,$$

where $(k_{j+1} - k_j)$ is larger than 3δ, for $j = 1, \ldots, N-1$.

Using (7), (9) and (10), we construct a closed form functions defined on Ω as follows:

$$\begin{aligned}
&\Psi_0(x) = \psi_0^R(x - k_1) \quad \text{(by using (9))},\\
&\Psi_1(x) = \psi_{[k_1,k_2]}^{(\delta,n-1)}(x) \quad \text{(by using (7))},\\
&\cdots = \cdots\\
&\Psi_{(N-1)}(x) = \psi_{[k_{N-1},k_N]}^{(\delta,n-1)}(x) \quad \text{(by using (7))},\\
&\Psi_N(x) = \psi_0^L(x - k_N) \quad \text{(by using (10))}.
\end{aligned}$$

Let us note that these functions are convolution functions as follows: Let $\chi_{[a,b]}$ be the characteristic function of $[a,b]$ and w_δ^n be the scaled window function defined by (2), then Theorem 3.5 of [23] shows that

$$\begin{aligned}
&\Psi_0(x) = \text{the convolution of } \chi_{(-\infty,k_1)} \text{ and the scaled window function } w_\delta^n,\\
&\Psi_1(x) = \text{the convolution of } \chi_{(k_1,k_2)} \text{ and the scaled window function } w_\delta^n,\\
&\quad\cdots = \cdots\\
&\Psi_{N-1}(x) = \text{the convolution of } \chi_{(k_{N-1},k_N)} \text{ and the scaled window function } w_\delta^n,\\
&\Psi_N(x) = \text{the convolution of } \chi_{(k_N,\infty)} \text{ and the scaled window function } w_\delta^n,
\end{aligned}$$

Therefore, the family of $\mathscr{C}^{n-1}$-functions $\{\Psi_0, \Psi_1, \ldots, \Psi_N\}$ is a partition of unity subordinate to the covering $\{(-\infty, k_1+\delta), (k_1-\delta, k_2+\delta), \ldots, (k_N-\delta, \infty)\}$ of Ω. Note that the width of the overlapping part of any two patches is 2δ.

The PU functions defined by (9) and (10) will play important roles for the construction of higher dimensional PU. Even though the supports of these two function are unbounded, we call them PU functions in what follows. Note that these two PU functions look like a left step function and a right step function, respectively, provided that δ is small.

Remark 3.1. The PU functions defined by (9) and (10) are basic building blocks for the product PU functions defined in the following sections. These PU functions are highly regular piecewise polynomials. However, in order to take advantage of piecewise polynomials in the numerical integrations, the integral domains could be divided into several pieces. Therefore, it is worthy to consider the following cutoff function [20] for the construction of the basic PU functions:

$$q(t) = \begin{cases} exp(-a/t^2), & \text{if } t > 0,\\ 0, & \text{if } t \leqslant 0, \end{cases} \tag{12}$$

where a is a positive constant. Then $q(t)$ is in $\mathscr{C}^\infty(\mathbb{R})$. Let

$$h(t) = \frac{q(t)}{q(t) + q(2\delta - t)}. \tag{13}$$

Let

$$\psi_0^L(x) = h(x+\delta), \quad \psi_0^R(x) = 1 - h(x+\delta) \tag{14}$$

Then these two smooth functions satisfy the conditions (11).

The tensor product of one-dimensional closed form PU functions, defined by (7), gives higher dimensional PU function for rectangular patches and cubic patches. However, the tensor product of one-dimensional PU functions is unable to yield any PU functions for triangular, quadrangular, tetrahedral, pentahedral, general hexahedral patches.

In next two sections, we introduce a simple unified method, named by *the generalized product method*, to construct smooth closed form partition of unity corresponding to given partition of two-dimensional (or three-dimensional) domain. The proposed method to construct PU also uses an unstructured background mesh. However, notice that it is not a FEM mesh, but a simple subdivision of the domain Ω in which hanging nodes and sides are allowed.

4. The higher dimensional partition of unity with flat-top

4.1. The generalized product partition of unity for two-dimensional domains

Using (9) and (10), we obtain two *basic two-dimensional $\mathscr{C}^{n-1}$-PU functions* defined by

$$\Psi_x^R(x,y) = \psi_0^R(x) \quad \text{and} \quad \Psi_x^L(x,y) = \psi_0^L(x), \quad \text{for all } (x,y) \in \mathbb{R}^2 \tag{15}$$

that satisfy

$$\Psi_x^R(x,y) + \Psi_x^L(x,y) = 1, \quad \text{for all } (x,y) \in \mathbb{R}^2.$$

In other words, two functions are the compositions of the coordinate projection, $(x,y) \to x$, and ψ_0^R, ψ_0^L, respectively. Moreover, observing $\Psi_x^L = 1 - \Psi_x^R$ may have an advantage on implementing these two basic functions. The graph of Ψ_x^R is sketched in Fig. 1. The schematic diagram for Ψ_x^R and Ψ_x^L are shown in Fig. 2. That is,

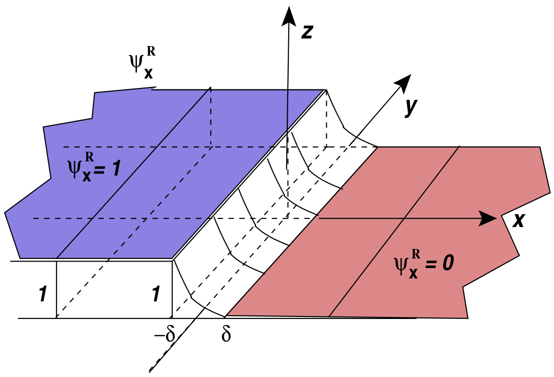

Fig. 1. Sketchy graph of Ψ_x^R.

$$\begin{cases} \Psi_x^L(x,y) = 1, & \text{if } x \geqslant \delta, \\ \Psi_x^L(x,y) = 0, & \text{if } x \leqslant -\delta, \\ 0 \leqslant \Psi_x^L(x,y) \leqslant 1, & \text{if } |x| \leqslant \delta \end{cases} \quad \text{and} \quad \begin{cases} \Psi_x^R(x,y) = 0, & \text{if } x \geqslant \delta, \\ \Psi_x^R(x,y) = 1, & \text{if } x \leqslant -\delta, \\ 0 \leqslant \Psi_x^R(x,y) \leqslant 1, & \text{if } |x| \leqslant \delta. \end{cases} \tag{16}$$

Suppose $\overrightarrow{P_1P_2}$ is a straight line connecting two points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ with $x_1 \leqslant x_2$ such that $y_1 < y_2$ if $x_1 = x_2$. Then the angle between the positive x-axis and $\overrightarrow{P_1P_2}$ is determined by the following formula:

$$\begin{cases} \theta = \tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right), & \text{if } x_2 \neq x_1, \\ \theta = \pi/2, & \text{if } x_2 = x_1. \end{cases} \tag{17}$$

Next, we consider an affine transformation on $\mathbb{R}^2$ that transforms the straight line $\overrightarrow{P_1P_2}$ onto the y-axis:

$$T_{P_1P_2}(x,y) = \begin{bmatrix} \cos(\pi/2 - \theta) & -\sin(\pi/2 - \theta) \\ \sin(\pi/2 - \theta) & \cos(\pi/2 - \theta) \end{bmatrix} \begin{bmatrix} x - x_1 \\ y - y_1 \end{bmatrix}. \tag{18}$$

Now, through the transformation $T_{P_1P_2}$, the PU constructed by (15), is transformed to a new PU as follows:

$$\Psi_{P_1P_2}(x,y) = \Psi_x^R(T_{P_1P_2}(x,y)), \quad \Psi^\star_{P_1P_2}(x,y) = \Psi_x^L(T_{P_1P_2}(x,y)),$$

that satisfy

$$\Psi^\star_{P_1P_2}(x,y) + \Psi_{P_1P_2}(x,y) = 1, \quad \text{for all } (x,y) \in \mathbb{R}^2. \tag{19}$$

Note that the straight line $\overrightarrow{P_1P_2}$ divides $\mathbb{R}^2$ into two patches: Ω_{above} (the above, or the left, of the line $\overrightarrow{P_1P_2}$) and Ω_{below} (the below, or the right, of the line $\overrightarrow{P_1P_2}$). Hence, $\{\Psi_{P_1P_2}, \Psi^\star_{P_1P_2}\}$ is the partition of unity of $\mathbb{R}^2$ corresponding to the background mesh $\{\Omega_{above}, \Omega_{below}\}$.

In what follows, applying this two-piece partition of unity on $\mathbb{R}^2$ to each side of patches of a background mesh, we build a closed form partition of unity that will be called the product partition of unity.

Without loss of generality, we assume the domain Ω is a polygonal domain with vertices $A_1, \ldots, A_6$, as shown in Fig. 3. We consider the following five cases of subdivisions (background meshes) of Ω, shown in Figs. 3–7. To each of these subdivisions, we construct the product partition of unity.

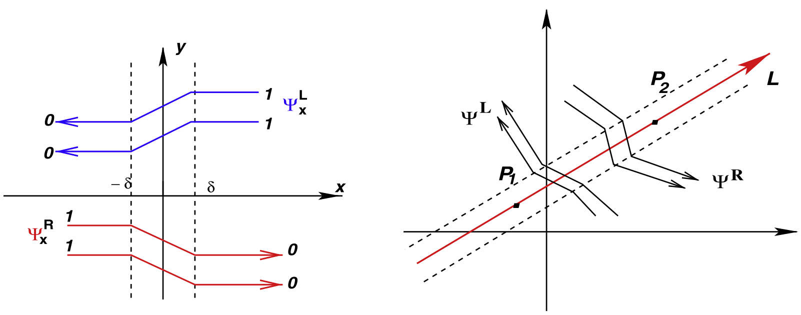

Fig. 2. Schematic diagram of basic PU functions Ψ_x^R and Ψ_x^L in dimension two (left). Transformed basic PU by the affine transformation $T_{P_1P_2}$ (right).

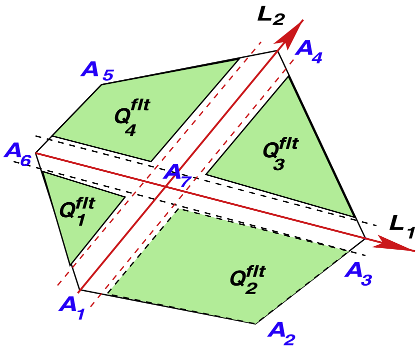

Fig. 3. The diagram of Case I.

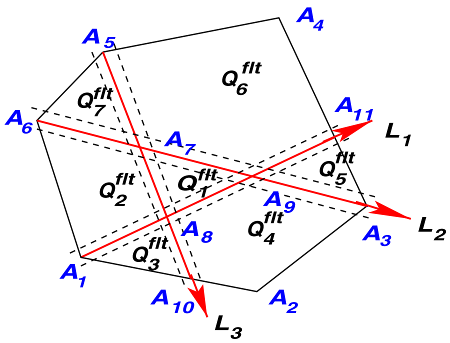

Fig. 4. The diagram of Case II.

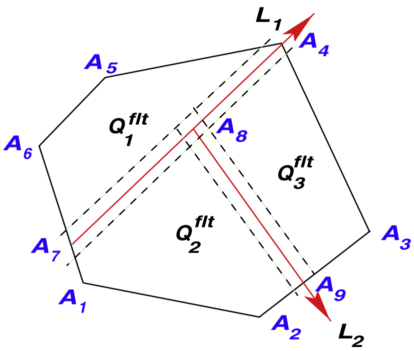

Fig. 5. The diagram of Case III.

Case I: *Background mesh determined by two intersecting lines*. Suppose the background mesh is determined by two intersecting lines as shown in Fig. 3. That is, the domain Ω is partitioned by two intersecting straight lines: $L_1 = \overrightarrow{A_6A_3}$ and $L_2 = \overrightarrow{A_1A_4}$.

Using (17) and (18), we define two affine transformations as follows:

$$T_1 = T_{A_6A_3}, \quad T_2 = T_{A_1A_4}.$$

Then, two pairs of PU on $\mathbb{R}^2$ are constructed by the following relations:

$$\Psi_1(x,y) = \Psi_x^R(T_1(x,y)) \quad \text{and} \quad \Psi_1^*(x,y) = 1 - \Psi_1(x,y),$$
$$\Psi_2(x,y) = \Psi_x^R(T_2(x,y)) \quad \text{and} \quad \Psi_2^*(x,y) = 1 - \Psi_2(x,y).$$

In other words, we have

- Ψ_1 is the PU function corresponding to the upper half plane that is above L_1; $\Psi_1^\star$ is the PU function corresponding to the lower half plane that is below L_1.

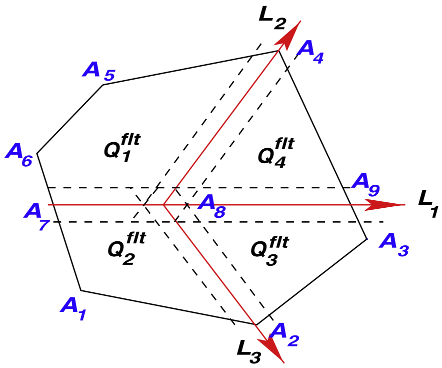

Fig. 6. The diagram of Case IV.

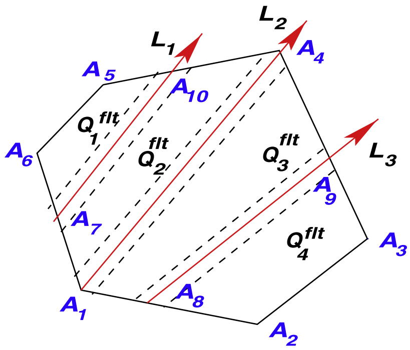

Fig. 7. The diagram of Case V.

- Ψ_2 is the PU function corresponding to the upper half plane that is above L_2; $\Psi_2^\star$ is the PU function corresponding to the lower half plane that is below L_2.
- $\Psi_1(x,y) + \Psi_1^\star(x,y) = 1;\ \Psi_2(x,y) + \Psi_2^\star(x,y) = 1,$ for $(x,y) \in \mathbb{R}^2$.

Thus, we have the following equalities:

$$1 = \left(\Psi_1 + \Psi_1^\star\right)\left(\Psi_2 + \Psi_2^\star\right) = \Psi_1\Psi_2 + \Psi_1^\star\Psi_2 + \Psi_1\Psi_2^\star + \Psi_1^\star\Psi_2^\star = \Psi_4^P + \Psi_1^P + \Psi_3^P + \Psi_2^P. \tag{20}$$

In other words, we have

- $\Psi_1^P = \Psi_1^\star\Psi_2,\ \Psi_2^P = \Psi_1^\star\Psi_2^\star,\ \Psi_3^P = \Psi_1\Psi_2^\star$, and $\Psi_4^P = \Psi_1\Psi_2$, are the PU functions corresponding to the triangular patch $Q_1 = A_6A_1A_7$, the quadrangular patch $Q_2 = A_1A_2A_3A_7$, the triangular patch $Q_3 = A_3A_4A_7$, and the quadrangular patch $Q_4 = A_4A_5A_6A_7$, respectively.

In Fig. 3, the shaded areas $Q_1^{flt}, Q_2^{flt}, Q_3^{flt}, Q_4^{flt}$, are the flat-top parts of the four PU functions, $\Psi_1^P,\ \Psi_2^P,\ \Psi_3^P,\ \Psi_4^P$, respectively. The strips bounded by dotted lines, which have 2δ-width, are the overlapping parts of adjacent PU functions. Each of these strips contain the non-flat-top parts of these PU functions. Note that for $j = 1,2,3,4$,

$$Q_j^{flt} = \{(x,y) \in Q_j | \mathrm{dist}((x,y), L_k) > \delta, k = 1,2\}.$$

It is important to note the following:

- For each $i = 1,2,3,4$, the four functions, $\Psi_1, \Psi_1^\star, \Psi_2, \Psi_2^\star$, are either 1 or 0 on the flat-top region Q_i^{flt}. Moreover, we observe: for $i = 1,2,3,4$,

$$\Psi_i^P \text{ is the product of only the functions } \Psi_1, \Psi_1^\star, \Psi_2, \Psi_2^\star, \text{ that equal 1 on } Q_i^{flt}.$$

Thus, the set of these product functions $\left\{\Psi_i^P | i = 1,2,3,4\right\}$ is called *the generalized product partition of unity*. We also call each of these functions *a product PU function*.

Case II: *Background mesh determined by three lines intersecting one another*. Suppose the domain Ω is partitioned by three straight lines, $L_1 = \overrightarrow{A_1A_8},\quad L_2 = \overrightarrow{A_6A_7},\quad L_3 = \overrightarrow{A_5A_7}$, to have patches $Q_1, \ldots, Q_7$, as shown in Fig. 4.Using (17) and (18), we define three affine transformations as follows:

$$T_1 = T_{A_1A_8}, \quad T_2 = T_{A_6A_3}, \quad T_2 = T_{A_5A_8}.$$

Then, three pairs of PU on $\mathbb{R}^2$ are constructed by the following relations:

$$\begin{aligned}
&\Psi_1(x,y) = \Psi_x^R(T_1(x,y)) \quad \text{and} \quad \Psi_1^\star(x,y) = 1 - \Psi_1(x,y),\\
&\Psi_2(x,y) = \Psi_x^R(T_2(x,y)) \quad \text{and} \quad \Psi_2^\star(x,y) = 1 - \Psi_2(x,y),\\
&\Psi_3(x,y) = \Psi_x^R(T_3(x,y)) \quad \text{and} \quad \Psi_3^\star(x,y) = 1 - \Psi_3(x,y).
\end{aligned} \tag{21}$$

Since $(\Psi_1 + \Psi_1^\star) = (\Psi_2 + \Psi_2^\star) = (\Psi_3 + \Psi_3^\star) = 1$, we have the following seven product PU functions:

$$\begin{aligned}
1 &= (\Psi_1 + \Psi_1^\star)(\Psi_2 + \Psi_2^\star)(\Psi_3 + \Psi_3^\star) = (\Psi_1\Psi_2 + \Psi_1^\star\Psi_2 + \Psi_1\Psi_2^\star + \Psi_1^\star\Psi_2^\star)(\Psi_3 + \Psi_3^\star)\\
&= \Psi_1^\star\Psi_2(\Psi_3 + \Psi_3^\star) + (\Psi_1\Psi_2 + \Psi_1\Psi_2^\star + \Psi_1^\star\Psi_2^\star)(\Psi_3 + \Psi_3^\star)\\
&= \Psi_5^P + (\Psi_1\Psi_2\Psi_3 + \Psi_1\Psi_2^\star\Psi_3 + \Psi_1^\star\Psi_2^\star\Psi_3) + (\Psi_1\Psi_2\Psi_3^\star + \Psi_1\Psi_2^\star\Psi_3^\star + \Psi_1^\star\Psi_2^\star\Psi_3^\star)\\
&= \Psi_5^P + \Psi_6^P + \Psi_1^P + \Psi_4^P + \Psi_7^P + \Psi_2^P + \Psi_3^P.
\end{aligned} \tag{22}$$

In other words, we have a product partition of unity whose members are the products of some of basic PU functions defined by (21). That is,

- $\Psi_1^P = \Psi_1\Psi_2^\star\Psi_3$, $\Psi_2^P = \Psi_1\Psi_2^\star\Psi_3^\star$, $\Psi_3^P = \Psi_1^\star(\Psi_2^\star)\Psi_3^\star$, $\Psi_4^P = \Psi_1^\star\Psi_2^\star\Psi_3$, $\Psi_5^P = \Psi_1^\star\Psi_2(\Psi_3)$, $\Psi_6^P = \Psi_1\Psi_2\Psi_3$, and $\Psi_7^P = (\Psi_1)\Psi_2\Psi_3^*$, are the PU function corresponding to the triangular patch $Q_1 = A_7A_8A_9$, the quadrangular patch $Q_2 = A_6A_1A_8A_7$, the triangular patch $Q_3 = A_1A_{10}A_8$, the pentagonal patch $Q_4 = A_{10}A_2A_3A_9A_8$, the triangular patch $Q_5 = A_3A_{11}A_9$, the pentagonal patch $Q_6 = A_{11}A_4A_5A_7A_9$, the triangular patch $Q_7 = A_5A_6A_7$, respectively. Here the PU functions inside braces, $(\Psi_2^\star), (\Psi_3), (\Psi_1)$, can be dropped because they are 1 on the supports of Ψ_3^P, Ψ_5^P, Ψ_7^P, respectively.

In Fig. 4, the strips with 2δ-width, bounded by dotted lines, are non-flat-top parts. If we write the union of non-flat-top parts by Ω^{nflt}, then $Q_i^{flt} = Q_i \setminus \Omega^{nflt} = \{(x,y) \in Q_i | \text{dist}((x,y), L_k) > \delta,\ k = 1,2,3\}$ is the flat-top part of Q_i for $i = 1,\dots,7$. Let us note the following:

- For $i = 1,\dots,7$, the product PU function Ψ_i^P is the product of only the functions $\Psi_1,\ \Psi_1^\star,\ \Psi_2,\ \Psi_2^\star,\ \Psi_3,\ \Psi_3^\star$, that equal 1 on Q_i^{flt}.

Case III: *Background mesh determined by one line and one ray*. A straight line $L_1 = \overleftrightarrow{A_7A_4}$ and a ray $L_2 = \overrightarrow{A_8A_9}$ divide Ω into three regions as shown in Fig. 5. Using (17) and (18), we define two affine transformations:

$$T_1 = T_{A_7A_4}, \qquad T_2 = T_{A_8A_9}.$$

Then, two pairs of PU on $\mathbb{R}^2$ are constructed by the following relations:

$$\begin{aligned}
&\Psi_1(x,y) = \Psi_x^R(T_1(x,y)) \quad \text{and} \quad \Psi_1^\star(x,y) = 1 - \Psi_1(x,y),\\
&\Psi_2(x,y) = \Psi_x^R(T_2(x,y)) \quad \text{and} \quad \Psi_2^\star(x,y) = 1 - \Psi_2(x,y).
\end{aligned}$$

Since $(\Psi_1 + \Psi_1^\star) = (\Psi_2 + \Psi_2^\star) = 1$, we have

$$1 = (\Psi_1 + \Psi_1^\star)(\Psi_2 + \Psi_2^\star) = \Psi_1(\Psi_2 + \Psi_2^\star) + \Psi_1^\star(\Psi_2 + \Psi_2^\star) = \Psi_1^P + \Psi_1^\star\Psi_2 + \Psi_1^\star\Psi_2^\star = \Psi_1^P + \Psi_3^P + \Psi_2^P. \tag{23}$$

Hence, we have proved that for $i = 1,2,3$, the product PU function Ψ_i^P is the product of only the functions $\Psi_1,\ \Psi_1^\star,\ \Psi_2,\ \Psi_2^\star$, that equal 1 on $Q_i^{flt} = \{(x,y) \in Q_i | \text{dist}((x,y), L_k) > \delta,\ k = 1,2\}$ in Fig. 5.

Case IV: *Background mesh determined by one line and two rays intersecting at a point so that two rays locate on the left side and on the right side of the line, respectively*.The domain Ω is partitioned into four patches, $Q_1,\dots,Q_4$, by one line $L_1 = \overleftrightarrow{A_7A_8}$ and two rays, $L_2 = \overrightarrow{A_8A_4}$, $L_3 = \overrightarrow{A_8A_2}$, as shown in Fig. 6. Using (17) and (18), we define three affine transformations as follows:

$$T_1 = T_{A_7A_8}, \quad T_2 = T_{A_8A_4}, \quad T_3 = T_{A_8A_2}.$$

Then, three pairs of PU on $\mathbb{R}^2$ are constructed by the following relations:

$$\begin{aligned}
&\Psi_1(x,y) = \Psi_x^R(T_1(x,y)) \quad \text{and} \quad \Psi_1^\star(x,y) = 1 - \Psi_1(x,y),\\
&\Psi_2(x,y) = \Psi_x^R(T_2(x,y)) \quad \text{and} \quad \Psi_2^\star(x,y) = 1 - \Psi_2(x,y),\\
&\Psi_3(x,y) = \Psi_x^R(T_3(x,y)) \quad \text{and} \quad \Psi_3^\star(x,y) = 1 - \Psi_3(x,y).
\end{aligned}$$

Since $\Psi_1 + \Psi_1^\star = \Psi_2 + \Psi_2^\star = \Psi_3 + \Psi_3^\star = 1$, we have the following relations:

$$\begin{aligned}
1 &= (\Psi_1 + \Psi_1^\star)(\Psi_2 + \Psi_2^\star)(\Psi_3 + \Psi_3^\star) = (\Psi_1\Psi_2 + \Psi_1^\star\Psi_2 + \Psi_1\Psi_2^\star + \Psi_1^\star\Psi_2^\star)(\Psi_3 + \Psi_3^\star)\\
&= \Psi_1\Psi_2(\Psi_3 + \Psi_3^\star) + \Psi_1\Psi_2^\star(\Psi_3 + \Psi_3^\star) + \Psi_1^\star\Psi_2(\Psi_3 + \Psi_3^\star) + \Psi_1^\star\Psi_2^\star(\Psi_3 + \Psi_3^\star)\\
&= \Psi_1\Psi_2 + \Psi_1\Psi_2^\star + \Psi_1^\star\Psi_3[\Psi_2 + \Psi_2^\star] + \Psi_1^\star\Psi_3^\star[\Psi_2 + \Psi_2^\star] = \Psi_1^P + \Psi_4^P + \Psi_3^P + \Psi_2^P.
\end{aligned} \tag{24}$$

In other words, $\Psi_1^P = \Psi_1\Psi_2$, $\Psi_2^P = \Psi_1^\star\Psi_3^\star$, $\Psi_3^P = \Psi_1^\star\Psi_3$, and $\Psi_4^P = \Psi_1\Psi_2^\star$. Thus, we have proved that for $i = 1,\dots,4$, the product PU function Ψ_i^P is the product of only the functions $\Psi_1,\ \Psi_1^\star,\ \Psi_2,\ \Psi_2^\star,\ \Psi_3,\ \Psi_3^\star$, that equal 1 on Q_i^{flt} in Fig. 6.

Case V: *Background mesh determined by straight line(s) with no intersections.*
Suppose three lines $L_1 = \overrightarrow{A_7A_{10}},\ L_2 = \overrightarrow{A_1A_4},\ L_3 = \overrightarrow{A_8A_9}$ with no intersections divide the domain Ω into four patches as shown in Fig. 7. Using (17) and (18), we define three affine transformations as follows:

$$T_1 = T_{A_7A_{10}}, \quad T_2 = T_{A_1A_4}, \quad T_3 = T_{A_8A_9}.$$

Then, three pairs of PU on $\mathbb{R}^2$ are constructed by the following relations:

$$\begin{aligned}
\Psi_1(x,y) &= \Psi_x^R(T_1(x,y)) \quad \text{and} \quad \Psi_1^\star(x,y) = 1 - \Psi_1(x,y);\\
\Psi_2(x,y) &= \Psi_x^R(T_2(x,y)) \quad \text{and} \quad \Psi_2^\star(x,y) = 1 - \Psi_2(x,y);\\
\Psi_3(x,y) &= \Psi_x^R(T_3(x,y)) \quad \text{and} \quad \Psi_3^\star(x,y) = 1 - \Psi_3(x,y).
\end{aligned}$$

Using $(\Psi_1 + \Psi_1^\star) = (\Psi_2 + \Psi_2^\star) = (\Psi_3 + \Psi_3^\star) = 1$, we have

$$\begin{aligned}
1 &= (\Psi_1 + \Psi_1^\star)(\Psi_2 + \Psi_2^\star)(\Psi_3 + \Psi_3^\star) = (\Psi_1\Psi_2 + \Psi_1^\star\Psi_2 + \Psi_1\Psi_2^\star + \Psi_1^\star\Psi_2^\star)(\Psi_3 + \Psi_3^\star)\\
&= \left(\Psi_1 + \Psi_2^P + 0 + \Psi_2^\star\right)(\Psi_3 + \Psi_3^\star) = \Psi_1 + \left(\Psi_2^P\Psi_3 + \Psi_2^\star\Psi_3\right) + \left(\Psi_2^P\Psi_3^\star + \Psi_2^\star\Psi_3^\star\right)\\
&= \Psi_1 + \left(\Psi_2^P + \Psi_3^P\right) + (0 + \Psi_3^\star) = \Psi_1^P + \Psi_2^P + \Psi_3^P + \Psi_4^P.
\end{aligned} \tag{25}$$

In other words, $\Psi_1^P = \Psi_1,\ \Psi_2^P = \Psi_1^\star\Psi_2,\ \Psi_3^P = \Psi_2^\star\Psi_3$, and $\Psi_4^P = \Psi_3^\star$, are the PU functions corresponding to the quadrangular patches $Q_1 = A_{10}A_5A_6A_7,\ Q_2 = A_7A_1A_4A_{10},\ Q_3 = A_1A_8A_9A_4$, and $Q_4 = A_2A_3A_9A_8$, respectively. Therefore, we have proved that for $i = 1,2,3,4$, the product PU function Ψ_i^P is the product of only the functions $\Psi_1,\ \Psi_1^\star,\ \Psi_2,\ \Psi_2^\star,\ \Psi_3,\ \Psi_3^\star$, that equal 1 on Q_i^{flt} in Fig. 7.

In the following theorem, combining the five cases shown above, we construct a closed form partition of unity with wide flat-top that corresponds to subdivisions of the domain Ω arising in the practical applications of meshless methods.

Theorem 4.1. *Suppose, for a partition of unity on a domain Ω, a background mesh on Ω is constructed by dividing it into the m-number of convex subregions (patches) $Q_1,\ldots,Q_m$, by the n-numbers of straight lines, rays, or broken lines $L_1,\ldots,L_n$ so that, for each $j = 1,\ldots,m$,*

$$Q_j^{flt} = \{(x,y) \in Q_j | \mathrm{dist}((x,y),L_k) > \delta, \quad k = 1,\ldots,n\}, \ \textit{the flat-top part of}\ Q_j,$$

has a positive measure. We assume the following rules and definitions:

1. *At each vertex of the partition, no more than two lines or rays can intersect, except Case IV in Fig. 6. That is, only those five cases shown in Figs. 3–7 and their combinations are allowed.*
2. *The orientations of lines and rays are as shown in Figs. 3–7. For $k = 1,\ldots,n$, T_{L_k} is an affine transformation on $\mathbb{R}^2$, defined by (17) and (18), that maps the line L_k onto the y-axis so that orientations can be matched. Using (17) and (18), we define the n-fairs of basic PU functions by*

$$\Psi_k^R = \Psi_x^R \circ T_{L_k} \quad \text{and} \quad \Psi_k^L = \Psi_x^L \circ T_{L_k}, \quad \text{for each } k = 1,\ldots,n. \tag{26}$$

3. *For each j, the patch Q_j is surrounded by the lines or rays $L_{j1},\ldots,L_{j\alpha}$ and Ψ_j^P is defined by the product of each of the 2α basic PU functions*

$$\Psi_{j1}^R, \Psi_{j1}^L; \ldots; \Psi_{j\alpha}^R, \Psi_{j\alpha}^L,$$

that is 1 on Q_j^{flt}.

Then, $\left\{\Psi_j^P | j = 1,\ldots,m\right\}$ is a partition of unity on Ω corresponding to the background mesh $\{Q_j | j = 1,\ldots,m\}$ of the domain Ω. Moreover, Q_j^{flt} is the flat-top part of supp$\left(\Psi_j^P\right)$ for $j = 1,2,\ldots,m$.

Note that the number of basic PU functions used for the product function Ψ_j^P is the same as the number of lines surrounding the patch Q_j (excluding those sides of Q_j that lie on the boundary of Ω). Thus, the PU function Ψ_j^P is said to be *the product partition of unity function*. We also note that even though the partitioning lines are broken (L_1 and L_6 in the right-hand side diagram of Fig. 9), the above rule for the construction of the product PU function holds.

Proof. From (26), for each $k = 1,\ldots,n$, we have

$$\Psi_k^L(x,y) + \Psi_k^R(x,y) = 1, \quad \text{for } (x,y) \in \mathbb{R}^2.$$

and hence,

$$\prod_{k=1}^{n}\left(\Psi_k^L + \Psi_k^R\right)(x,y) = 1, \quad \text{for all } (x,y) \in \mathbb{R}^2. \tag{27}$$

The expansion of (27) yields 2^n-terms, each of which consists of the product of n basic PU functions. Each term is a non-negative valued function and the sum of these 2^n-terms is 1 on $\mathbb{R}^2$ and there also exist terms that are identically zero on $\mathbb{R}^2$. Moreover, each non-zero term of these 2^n-terms is a product of n functions and corresponds to one patch Q_j. However, if we drop those among these n functions that are identically 1 on

$$Q_j^\delta = \{(x,y) \in \Omega | \mathrm{dist}((x,y),Q_j) < \delta\},$$

the number of remaining functions is the same as the number of lines or rays surrounding Q_j.

For brevity, we adopt the following notation: for $k = 1,\dots,n$,

$$\Psi_k = \Psi_k^R, \quad \Psi_k^\star = \Psi_k^L = 1 - \Psi_k.$$

In the arguments preceding Theorem 4.1, we proved Theorem 4.1 for those five cases of partitions shown in Figs. 3–7.

Now, we show that a proper combination of five cases (Figs. 3–7) leads to a product partition of unity for a general background mesh of Ω.

We consider two illustrative examples:

(a) *The combination of Cases I and III*: From the Case I of Fig. 3 and the Case III of Fig. 5, we have the following:

$$\begin{aligned}
1 &= \left[\left(\Psi_1^P\right)_I + \left(\Psi_2^P\right)_I + \left(\Psi_3^P\right)_I + \left(\Psi_4^P\right)_I\right]\left[\left(\Psi_1^P\right)_{III} + \left(\Psi_2^P\right)_{III} + \left(\Psi_3^P\right)_{III}\right] \\
&= \left(\Psi_1^P\right)_I + \left(\Psi_2^P\right)_I\left[\left(\Psi_1^P\right)_{III} + \left(\Psi_2^P\right)_{III} + \left(\Psi_3^P\right)_{III}\right] + \left(\Psi_3^P\right)_I\left[\left(\Psi_1^P\right)_{III} + \left(\Psi_2^P\right)_{III} + \left(\Psi_3^P\right)_{III}\right] \\
&\quad + \left(\Psi_4^P\right)_I\left[\left(\Psi_1^P\right)_{III} + \left(\Psi_2^P\right)_{III} + \left(\Psi_3^P\right)_{III}\right] \\
&= \left(\Psi_1^P\right)_I + \left[\Psi_2^\star\Psi_1^\star\Psi_3 + \Psi_2^\star\Psi_1^\star\Psi_3^\star\Psi_4^\star + \Psi_2^\star\Psi_1^\star\Psi_3^\star\Psi_4\right] + \left[\Psi_1\Psi_2^\star\Psi_3 + \Psi_1\Psi_2^\star\Psi_3^\star\Psi_4^\star + \Psi_1\Psi_2^\star\Psi_3^\star\Psi_4\right] \\
&\quad + \left[\Psi_1\Psi_2\Psi_3 + \Psi_1\Psi_2\Psi_3^\star\Psi_4^\star + \Psi_1\Psi_2\Psi_3^\star\Psi_4\right] \\
&= \Psi_1^\star\Psi_2 + \left[\Psi_2^\star\Psi_1^\star\Psi_3 + \Psi_1^\star\Psi_3^\star + 0\right] + \left[\Psi_1\Psi_2^\star\Psi_3 + \Psi_1\Psi_3^\star\Psi_4^\star + \Psi_2^\star\Psi_3^\star\Psi_4\right] + \left[\Psi_1\Psi_2\Psi_3 + 0 + \Psi_2\Psi_3^\star\right] \\
&= \Psi_1^P + \left[\Psi_2^P + \Psi_5^P\right] + \left[\Psi_3^P + \Psi_6^P + \Psi_7^P\right] + \left[\Psi_4^P + \Psi_8^P\right].
\end{aligned} \tag{28}$$

Hence, from the schematic diagram in Fig. 8, one can see that, for $j = 1,\dots,8$,

$$\Psi_j^P \text{ is the product of each of eight PU functions } \Psi_k,\ \Psi_k^\star,\ k = 1,\dots,4,$$
$$\text{that is 1 on } Q_j^{flt}, \text{ but not 1 on } Q_j^{n-flt} \left(\text{that is, drop those which are 1 on supp } \Psi_j^P\right).$$

(b) *Multiple combinations of Cases I and V*: Without loss of generality, we can assume that the domain Ω is partitioned into 20 patches by seven lines with various slopes shown in Fig. 9. Let

$$\Psi_k = \Psi_k^R, \quad \Psi_k^\star = 1 - \Psi_k, \quad k = 1,\dots,7,$$

be the seven pairs of PU defined by using (17) and (18) with respect to the seven lines $L_1, L_2, \dots, L_7$, shown in Fig. 9. Then for all point in the domain, the following functions are zero :

$$\Psi_1\Psi_2^\star,\ \Psi_1\Psi_3^\star,\ \Psi_1\Psi_4^\star;\quad \Psi_2\Psi_3^\star,\ \Psi_2\Psi_4^\star;\ \Psi_3\Psi_4^\star, \tag{29}$$
$$\Psi_5^\star\Psi_6,\ \Psi_5^\star\Psi_7;\quad \Psi_6^\star\Psi_7. \tag{30}$$

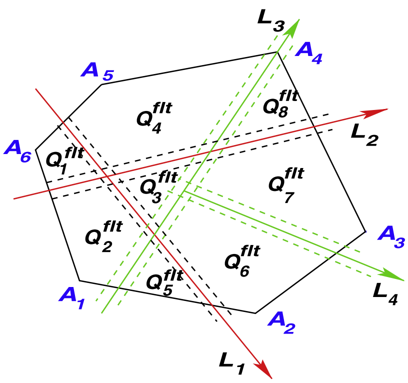

Fig. 8. The diagram of the combination of Cases I and III.

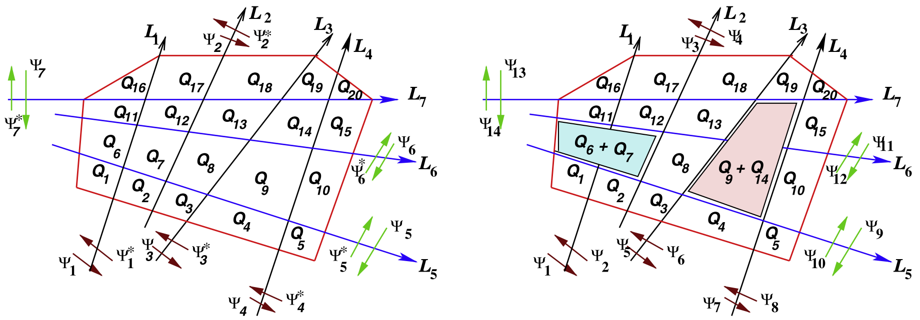

Fig. 9. (Left) Combination of Cases I and V. (Right) Combining product PU functions to make larger patches.

However, if any two lines among the four vertically tilted lines (L_1, L_2, L_3, L_4) are crossing inside Ω, some functions in the list (29) are not identically zero. Whereas, if any two lines among the horizontally tilted lines (L_5, L_6, L_7) are crossing inside Ω, some functions in the list (30) are not identically zero.

Using (29) and (30), Eq. (27) can be expanded as follows:

$$
\begin{aligned}
1 &= \prod_{k=1}^{7}\left(\Psi_k + \Psi_k^\star\right)\\
&= \left(\Psi_1\Psi_2 + \Psi_1^\star\Psi_2 + \left[\Psi_1\Psi_2^\star \equiv 0\right] + \Psi_1^\star\Psi_2^\star\right)\prod_{k=3}^{7}\left(\Psi_k + \Psi_k^\star\right)\\
&= \left(\Psi_1\Psi_2\Psi_3 + \Psi_1^\star\Psi_2\Psi_3 + \Psi_1^\star\Psi_2^\star\Psi_3 + \left[\left(\Psi_1\Psi_2\Psi_3^\star + \Psi_1^\star\Psi_2\Psi_3^\star\right) \equiv 0\right] + \Psi_1^\star\Psi_2^\star\Psi_3^\star\right)\prod_{k=4}^{7}\left(\Psi_k + \Psi_k^\star\right)\\
&= \Big(\Psi_1\{\Psi_2\Psi_3\Psi_4\} + \Psi_1^\star\Psi_2\{\Psi_3\Psi_4\} + \{\Psi_1\}\Psi_2^\star\Psi_3\{\Psi_4\} + \{\Psi_1^\star\Psi_2^\star\}\Psi_3^\star\Psi_4\\
&\quad + \left[\left(\Psi_1\Psi_2\Psi_3\Psi_4^\star + \Psi_1^\star\Psi_2\Psi_3\Psi_4^\star + \Psi_1^\star\Psi_2^\star\Psi_3\Psi_4^\star\right) \equiv 0\right] + \{\Psi_1^\star\Psi_2^\star\Psi_3^\star\}\Psi_4^\star\Big)\prod_{k=5}^{7}\left(\Psi_k + \Psi_k^\star\right)\\
&= \left(\Psi_1 + \Psi_1^\star\Psi_2 + \Psi_2^\star\Psi_3 + \Psi_3^\star\Psi_4 + \Psi_4^\star\right)\prod_{k=5}^{7}\left(\Psi_k + \Psi_k^\star\right)\\
&= \left(\Psi_1\Psi_5^\star + \Psi_1^\star\Psi_2\Psi_5^\star + \Psi_2^\star\Psi_3\Psi_5^\star + \Psi_3^\star\Psi_4\Psi_5^\star + \Psi_4^\star\Psi_5^\star\right)\prod_{k=6}^{7}\left(\Psi_k + \Psi_k^\star\right)\\
&\quad + \left(\Psi_1\Psi_5 + \Psi_1^\star\Psi_2\Psi_5 + \Psi_2^\star\Psi_3\Psi_5 + \Psi_3^\star\Psi_4\Psi_5 + \Psi_4^\star\Psi_5\right)\prod_{k=6}^{7}\left(\Psi_k + \Psi_k^\star\right)\\
&= \left(\Psi_1\Psi_5^\star + \Psi_1^\star\Psi_2\Psi_5^\star + \Psi_2^\star\Psi_3\Psi_5^\star + \Psi_3^\star\Psi_4\Psi_5^\star + \Psi_4^\star\Psi_5^\star\right)\\
&\quad + \left(\Psi_1\Psi_5\Psi_6 + \Psi_1^\star\Psi_2\Psi_5\Psi_6 + \Psi_2^\star\Psi_3\Psi_5\Psi_6 + \Psi_3^\star\Psi_4\Psi_5\Psi_6 + \Psi_4^\star\Psi_5\Psi_6\right)\left(\Psi_7 + \Psi_7^\star\right)\\
&\quad + \left(\Psi_1\Psi_5\Psi_6^\star + \Psi_1^\star\Psi_2\Psi_5\Psi_6^\star + \Psi_2^\star\Psi_3\Psi_5\Psi_6^\star + \Psi_3^\star\Psi_4\Psi_5\Psi_6^\star + \Psi_4^\star\Psi_5\Psi_6^\star\right)\left(\Psi_7 + \Psi_7^\star\right)\\
&= \left(\Psi_1\Psi_5^\star + \Psi_1^\star\Psi_2\Psi_5^\star + \Psi_2^\star\Psi_3\Psi_5^\star + \Psi_3^\star\Psi_4\Psi_5^\star + \Psi_4^\star\Psi_5^\star\right)\\
&\quad + \left(\Psi_1\{\Psi_5\}\Psi_6\Psi_7 + \Psi_1^\star\Psi_2\{\Psi_5\}\Psi_6\Psi_7 + \Psi_2^\star\Psi_3\{\Psi_5\}\Psi_6\Psi_7 + \Psi_3^\star\Psi_4\{\Psi_5\}\Psi_6\Psi_7 + \Psi_4^\star\{\Psi_5\}\Psi_6\Psi_7\right)\\
&\quad + \left(\Psi_1\{\Psi_5\}\Psi_6\Psi_7^\star + \Psi_1^\star\Psi_2\{\Psi_5\}\Psi_6\Psi_7^\star + \Psi_2^\star\Psi_3\{\Psi_5\}\Psi_6\Psi_7^\star + \Psi_3^\star\Psi_4\{\Psi_5\}\Psi_6\Psi_7^\star + \Psi_4^\star\{\Psi_5\}\Psi_6\Psi_7^\star\right)\\
&\quad + \left[\left(\Psi_1\Psi_5\Psi_6^\star + \Psi_1^\star\Psi_2\Psi_5\Psi_6^\star + \Psi_2^\star\Psi_3\Psi_5\Psi_6^\star + \Psi_3^\star\Psi_4\Psi_5\Psi_6^\star + \Psi_4^\star\Psi_5\Psi_6^\star\right)\Psi_7 \equiv 0\right]\\
&\quad + \Big(\Psi_1\Psi_5\Psi_6^\star\{\Psi_7^\star\} + \Psi_1^\star\Psi_2\Psi_5\Psi_6^\star\{\Psi_7^\star\} + \Psi_2^\star\Psi_3\Psi_5\Psi_6^\star\{\Psi_7^\star\}\\
&\quad + \Psi_3^\star\Psi_4\Psi_5\Psi_6^\star\{\Psi_7^\star\} + \Psi_4^\star\Psi_5\Psi_6^\star\{\Psi_7^\star\}\Big) \qquad (31)\\
&= \left(\Psi_1\Psi_5^\star + \Psi_1^\star\Psi_2\Psi_5^\star + \Psi_2^\star\Psi_3\Psi_5^\star + \Psi_3^\star\Psi_4\Psi_5^\star + \Psi_4^\star\Psi_5^\star\right)\\
&\quad + \left(\Psi_1\Psi_7 + \Psi_1^\star\Psi_2\Psi_7 + \Psi_2^\star\Psi_3\Psi_7 + \Psi_3^\star\Psi_4\Psi_7 + \Psi_4^\star\Psi_7\right)\\
&\quad + \left(\Psi_1\Psi_6\Psi_7^\star + \Psi_1^\star\Psi_2\Psi_6\Psi_7^\star + \Psi_2^\star\Psi_3\Psi_6\Psi_7^\star + \Psi_3^\star\Psi_4\Psi_6\Psi_7^\star + \Psi_4^\star\Psi_6\Psi_7^\star\right)\\
&\quad + \left(\Psi_1\Psi_5\Psi_6^\star + \Psi_1^\star\Psi_2\Psi_5\Psi_6^\star + \Psi_2^\star\Psi_3\Psi_5\Psi_6^\star + \Psi_3^\star\Psi_4\Psi_5\Psi_6^\star + \Psi_4^\star\Psi_5\Psi_6^\star\right)\\
&= \left(\sum_{i=1}^{5}\Psi_i^P\right) + \left(\sum_{i=16}^{20}\Psi_i^P\right) + \left(\sum_{i=11}^{15}\Psi_i^P\right) + \left(\sum_{i=6}^{10}\Psi_i^P\right). \qquad (32)
\end{aligned}
$$

In (31), $\Psi_5 = 1$ on the support of $\Psi_i^P,\ i = 11, \ldots, 20$; and $\Psi_7^\star = 1$ on the support of $\Psi_i^P,\ i = 6, \ldots, 10$. Thus, $\Psi_5(\Psi_7^\star)$ is dropped from the product functions for $\Psi_i^P,\ i = 11, \ldots, 20,\ (\Psi_i^P,\ i = 11, \ldots, 20,)$, respectively. From the schematic diagram in Fig. 9, one can see that

$$\Psi_j^P \text{ is the product of each of 14 PU functions } \Psi_k, \Psi_k^\star,\ k = 1,\dots,7,$$
$$\text{that is 1 on } Q_j^{flt},\ \text{but not 1 on } Q_j^{nflt}\ \left(\text{that is, drop those which are 1 on supp } \Psi_j^P\right).$$

If some of the partitioning lines are broken lines, then the sum of several terms makes 1 on the whole Q_j^{flt}. For example, suppose we break the line L_6 to make a lager patch in the right hand side diagram of Fig. 9, then the new product PU function corresponding to this enlarged patch, $Q_9 \cup Q_{14}$, is $\left(\Psi_9^P + \Psi_{14}^P\right) = \Psi_3^\star\Psi_4\Psi_5\Psi_6^\star\{\Psi_7^\star\} + \Psi_3^\star\Psi_4\Psi_5\Psi_6\{\Psi_7^\star\} = \Psi_3^\star\Psi_4\Psi_5\Psi_7^\star$, since $\Psi_6 + \Psi_6^\star = 1$.

If we break the line L_1 to join Q_6 with Q_7 (Fig. 9), then the product PU function corresponding to $Q_6 \cup Q_7$ is $\Psi_6^P + \Psi_7^P = (\Psi_1\Psi_5\Psi_6^\star\{\Psi_7^\star\} + \Psi_1^\star\Psi_2\Psi_5\Psi_6^\star\{\Psi_7^\star\}) = \left(\Psi_1\{\Psi_2\}\Psi_5\Psi_6^\star + \Psi_1^\star\Psi_2\Psi_5\P si_6^\star\right) = \Psi_2\Psi_5\Psi_6^\star$, since $\{\Psi_2\}$ and $\{\Psi_7^\star\}$ are 1 on the support of $\Psi_1\Psi_5\Psi_6^\star$. $\square$

Suppose two vertical lines $x = a(L_1);\ x = b(L_2)$ with $a < b$ and two horizontal lines $y = c(L_3);\ y = d(L_4)$ with $c < d$ intersect to make a rectangular patch $Q_{rec} = [a,b] \times [c,d]$. Then, by the rule defined in Theorem 4.1, the PU function corresponding to Q_{rec} is the same as tensor product of two one-dimensional PU functions defined by (7). In other words, we have

$$\Psi_1^\star\Psi_2\Psi_3^\star\Psi_4 = \psi_{[a,b]}^{\delta,n-1}(x) \times \psi_{[c,d]}^{\delta,n-1}(y),$$

that justify why partition of unity in Theorem 4.1 is coined to be generalized product PU.

Note that a similar idea is also used in RPEM (reproducing particle element methods) [7–11].

In the following two examples, we consider simple partition of unity in which the generalized product method constructing PU, stated in the Theorem 4.1, does not hold.

Example 4.1. Suppose the domain Ω is partitioned into four patches, $Q_i,\ i = 1,\dots,4$, by the line, L_1 and two rays, $L_2,\ L_3$, shown in the left figure of Fig. 10.

Unlike the above case (IV), both two rays, L_2 and L_3, are on the one side of the line L_1 in this example, as shown in Fig. 10. Then we have the following:

$$\begin{aligned}
1 &= (\Psi_1 + \Psi_1^\star)(\Psi_2 + \Psi_2^\star)(\Psi_3 + \Psi_3^\star) = \left[\Psi_1(\Psi_2 + \Psi_2^\star) + \Psi_1^\star(\Psi_2 + \Psi_2^\star)\right](\Psi_3 + \Psi_3^\star) \\
&= \left[\Psi_1\Psi_2(\Psi_3 + \Psi_3^\star) + \Psi_1\Psi_2^\star(\Psi_3 + \Psi_3^\star) + \Psi_1^\star\right] = \left[\Psi_1\Psi_2\Psi_3 + \Psi_1\Psi_2\Psi_3^\star + \Psi_1\Psi_2^\star + \Psi_1^\star\right] \\
&= \Psi_2\Psi_4^P + \Psi_1\Psi_3^P + \Psi_2^P + \Psi_1^P \neq \Psi_4^P + \Psi_3^P + \Psi_2^P + \Psi_1^P.
\end{aligned}$$

The globally defined functions, $\Psi_2\Psi_4^P, \Psi_1\Psi_3^P, \Psi_2^P, \Psi_1^P$, become a partition of unity on $\mathbb{R}^2$, however, they do not satisfy the rules stated in Theorem 4.1 for the generalized product partition of unity. Note that $\mathrm{supp}\left(\Psi_4^P\right) - \mathrm{supp}\left(\Psi_2\Psi_4^P\right) = \triangle abc$ in Fig. 10.

Example 4.2. Suppose the domain Ω is partitioned into five patches, $Q_1,\ Q_2,\ Q_3,\ Q_{4A},\ Q_{4B}$ by two lines, $L_1,\ L_2$, and a ray L_3, shown in the right figure of Fig. 10.

In the Case I of Fig. 3, we add a ray L_3 to make the partition shown in the right hand side of Fig. 10 so that three lines meet at one point. Then, we obtain a partition of unity that does not satisfy the rules for the product partition of unity as shown below:

$$\begin{aligned}
1 &= \Psi_1^P(= \Psi_1^\star\Psi_2) + \Psi_2^P(= \Psi_1^\star\Psi_2^\star) + \Psi_3^P(= \Psi_1\Psi_2^\star) + \Psi_4^P(= \Psi_1\Psi_2) = \left[\Psi_1^P + \Psi_2^P + \Psi_3^P + \Psi_4^P(= \Psi_1\Psi_2)\right](\Psi_3 + \Psi_3^\star) \\
&= \Psi_1^P + \Psi_2^P + \Psi_3^P + \left[\Psi_1\Psi_2\Psi_3 + \Psi_1\Psi_2\Psi_3^\star\right] = \Psi_1^P + \Psi_2^P + \Psi_3^P + \left[\Psi_2\Psi_{4B}^P + \Psi_1\Psi_{4A}^P\right],
\end{aligned}$$

in which the last two PU functions do not satisfy the rules for the generalized product method constructing PU.

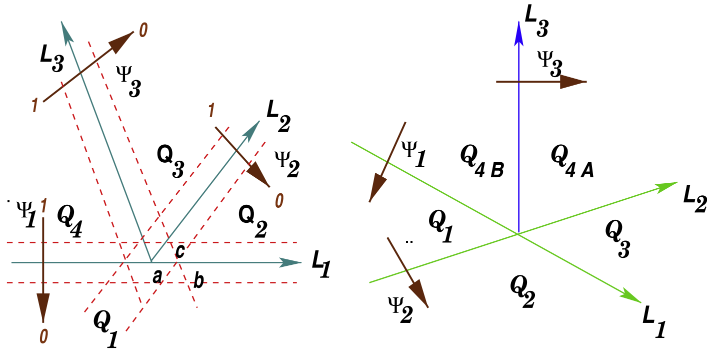

Fig. 10. Diagrams of non-allowable cases.

4.2. The generalized product partition of unity for three-dimensional domain

Using (9) and (10), we obtain four *basic three-dimensional $\mathscr{C}^{n-1}$-PU functions* defined by

$$\begin{aligned} &\Psi_x^R(x,y,z) = \psi_0^R(x) \quad \text{and} \quad \Psi_x^L(x,y,z) = \psi_0^L(x), \quad \text{for all } (x,y,z) \in \mathbb{R}^3; \\ &\Psi_z^R(x,y,z) = \psi_0^R(z) \quad \text{and} \quad \Psi_z^L(x,y,z) = \psi_0^L(z), \ \text{for all } (x,y,z) \in \mathbb{R}^3, \end{aligned} \tag{33}$$

such that

$$\begin{aligned} &\Psi_x^R(x,y,z) + \Psi_x^L(x,y,z) = 1, \quad \text{for all } (x,y,z) \in \mathbb{R}^3; \\ &\Psi_z^R(x,y,z) + \Psi_z^L(x,y,z) = 1, \quad \text{for all } (x,y,z) \in \mathbb{R}^3. \end{aligned}$$

The schematic diagram for Ψ_x^R and Ψ_x^L are shown in Fig. 11.

Theorem 4.2. *Let*

$$\Gamma_1^x, \ldots, \Gamma_{N_x}^x$$

be (vertically slanted or vertical) planes whose intersections with the x–y plane are those lines and rays allowed in Theorem 4.1, and

$$\Gamma_1^z, \ldots, \Gamma_{N_z}^z,$$

be (horizontal or horizontally slanted) planes that are mutually disjoint within the domain $\Omega \subset \mathbb{R}^3$. Suppose Ω is partitioned into the M-number of convex subregions $Q_1, \ldots, Q_M$, by vertically slicing with planes, $\Gamma_1^x, \ldots, \Gamma_{N_x}^x$, and by horizontally slicing with planes, $\Gamma_1^z, \ldots, \Gamma_{N_z}^z$, so that, for each $I = 1, \ldots, M, Q_I^{flt} = \{(x,y,z) \in Q_I | dist((x,y,z), \Gamma_k) > \delta, \ k = 1, \ldots, N = (N_x + N_z)\}$, the flat-top part of Q_I, has a positive measure. We assume the following rules and definitions:

1. *The orientations of planes are as usual: the normal vector v_k to the plane Γ_k has the same direction as the vector product of two non-co-line vectors on the plane Γ_k, in the counter clockwise direction. For $k = 1, \ldots, N$, T_k is an affine transformation on $\mathbb{R}^3$ that maps the plane Γ_k onto the x–y-plane so that orientations can be matched.*
2. *We define the 2N basic PU functions by*
$$\Psi_k^R = \Psi_x^R \circ T_k \quad \text{and} \quad \Psi_k^L = \Psi_x^L \circ T_k, \quad k = 1, \ldots, N = (N_x + N_z). \tag{34}$$
3. *Let Ψ_I^P be the product of each of those 2N basic PU functions*
$$\Psi_k^R, \Psi_k^L = 1 - \Psi_k^R, \quad k = 1, \ldots, N$$

that is 1 on Q_I^{flt}, but not identically 1 on $Q_I^\delta = \{(x,y,z) \in \Omega | dist((x,y,z), Q_I) < \delta\}$.

Then $\left\{\Psi_I^P | I = 1, \ldots, M\right\}$ is a partition of unity and the flat-top subregion of the support of Ψ_I^P is Q_I^{flt}. The PU function Ψ_I^P is the product of the same number of basic PU functions as the number of planes surrounding the patch Q_I.

We call the PU function Ψ_I^P that is the product of basic PU functions corresponding to planes surrounding the convex subset Q_I by *the product partition of unity function.*

Proof. Using arguments similar to Theorem 4.1, we have the vertical PU functions such that

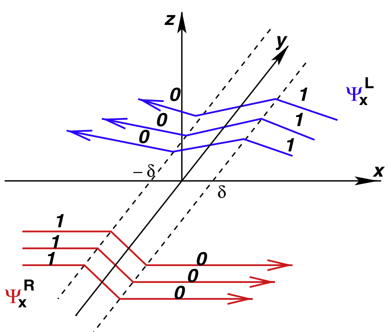

Fig. 11. Schematic diagram of basic PU functions Ψ_x^R and Ψ_x^L in dimension three.

$$\sum_{l=1}^{M_x}\left(\Psi_l^P\right)_v = 1. \tag{35}$$

Here, M_x is the number of the subregions of Ω divided by the vertically slanted planes. Using an argument similar to Case V in the proof of Theorem 4.1, we have the horizontal PU functions such that

$$\sum_{l=1}^{M_z}\left(\Psi_l^P\right)_h = 1, \tag{36}$$

where M_z is the number of the subregions of Ω divided by the horizontally slanted planes. Then we have

$$1 = \sum_{t=1}^{M_y}\left[\sum_{s=1}^{M_x}(\Psi_s^P)_v\right]\left(\Psi_t^P\right)_h = \sum_i^M \Psi_i^P. \qquad \square \tag{37}$$

Example 4.3. Suppose a polygonal domain Ω is partitioned into two wedges and six hexahedrons as shown in Fig. 12 by the planes: $\Gamma_1(\overline{stiu\,vb}\,\text{half plane})$, $\Gamma_2(\overline{BcGJhE})$, $\Gamma_3(\overline{pqdrsg})$, $\Gamma_4(\overline{acefgh})$. Note that Γ_1 does not cut through the domain Ω. That is, the eight patches in Fig. 12 are

$$Q_1 = \text{hexahedron} : \overline{AstEabih},\ Q_2 = \ \text{wedge} : \overline{tsBibc},\ Q_3 = \text{hexahedron} : \overline{EBqphcdg}, Q_4 = \text{hexahedron} : \overline{pqCDgdef},\ Q_5 = \ \text{hexahedron} : \overline{abihF\,vuJ},\ Q_6 = \text{wedge}\,\overline{ibcu\,vG}, Q_7 = \text{hexahedron} : \overline{hcdgJGfs},\ Q_8 = \ \text{hexahedron} : \overline{gdefsrHI}.$$

Consider the following affine transformations (rotation and translation) on $\mathbb{R}^3$:

1. T_s^t moves the line $s \to t$ in Fig. 12 onto the positive y-axis while the x-axis is on the bottom plane;
2. T_B^E moves the line $B \to E$ in Fig. 12 onto the positive y-axis while the x-axis is on the bottom plane;
3. T_q^p moves the line $q \to p$ in Fig. 12 onto the positive y-axis while the x-axis is on the bottom plane;
4. T_{cen} moves the middle face $\overline{abcdefgh}$ in Fig. 12 onto the x—y plane.

In order to construct PU functions for the eight patches, using the affine transformations and (33), we now construct *basic PU functions* defined on $\mathbb{R}^3$ as follows:

$$\begin{aligned}
\Psi_1 &= \Psi_x^R \circ T_s^t, \quad \Psi_1^\star = 1 - \Psi_1,\\
\Psi_2 &= \Psi_x^R \circ T_B^E, \quad \Psi_2^\star = 1 - \Psi_2,\\
\Psi_3 &= \Psi_x^R \circ T_q^p, \quad \Psi_3^\star = 1 - \Psi_3,\\
\Psi_4 &= \Psi_z^R \circ T_{cen}, \quad \Psi_4^\star = 1 - \Psi_4.
\end{aligned}$$

We define the flat-top part of a patch Q_j as follows:

$$Q_j^{flt} = \{(x,y,z) \in Q_j |\ \mathrm{dist}(\Gamma_k,(x,y,z)) > \delta, k = 1,2,3,4\},$$

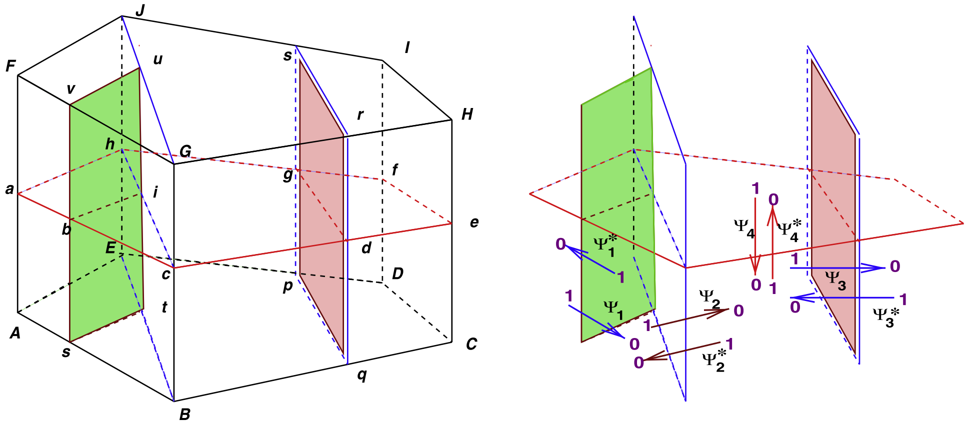

Fig. 12. Three-dimensional domain and partition into eight patches (left); diagram of eight basic PU functions (right).

where $\Gamma_k,\ k=1,2,3,4$, are the four planes partitioning Ω in Fig. 12. Then, by Theorem 4.2, we obtain the smooth product PU functions with flat-top as follows:

1. $\Psi_1^P=\Psi_1\Psi_2\Psi_4^\star$ (the product of those basic PU functions that are one on Q_1^{flt}) is a PU function corresponding to patch Q_1.
2. $\Psi_2^P=\Psi_1^\star\Psi_2\Psi_4^\star$ (the product of those basic PU functions that are one on Q_2^{flt}) is a PU function corresponding to patch Q_2.
3. $\Psi_3^P=\Psi_2^\star\Psi_3\Psi_4^\star$ (the product of those basic PU functions that are one on Q_3^{flt}) is a PU function corresponding to patch Q_3.
4. $\Psi_4^P=\Psi_3^\star\Psi_4^\star$ (the product of those basic PU functions that are one on Q_4^{flt}) is a PU function corresponding to patch Q_4.
5. $\Psi_5^P=\Psi_1\Psi_2\Psi_4$ (the product of those basic PU functions that are one on Q_5^{flt}) is a PU function corresponding to patch Q_5.
6. $\Psi_6^P=\Psi_1^\star\Psi_2\Psi_4$ (the product of those basic PU functions that are one on Q_6^{flt}) is a PU function corresponding to patch Q_6.
7. $\Psi_7^P=\Psi_2^\star\Psi_3\Psi_4$ (the product of those basic PU functions that are one on Q_7^{flt}) is a PU function corresponding to patch Q_7.
8. $\Psi_8^P=\Psi_3^\star\Psi_4$ (the product of those basic PU functions that are one on Q_8^{flt}) is a PU function corresponding to patch Q_8.

We can prove directly that these eight functions become a partition of unity. Actually, from the definition of basic PU functions, for all points in $\mathbb{R}^3$, we have

$$\begin{aligned}
1 &= (\Psi_1+\Psi_1^\star)(\Psi_2+\Psi_2^\star)(\Psi_3+\Psi_3^\star)(\Psi_4+\Psi_4^\star)\\
&= \Psi_1\Psi_2\Psi_3\Psi_4+\Psi_1\Psi_2\Psi_3^\star\Psi_4+\Psi_1^\star\Psi_2\Psi_3\Psi_4+\Psi_1^\star\Psi_2\Psi_3^\star\Psi_4+\Psi_1\Psi_2^\star\Psi_3\Psi_4+\Psi_1^\star\Psi_2^\star\Psi_3\Psi_4+\Psi_1\Psi_2^\star\Psi_3\Psi_4^\star\\
&\quad+\Psi_1^\star\Psi_2^\star\Psi_3\Psi_4^\star+\Psi_1\Psi_2^\star\Psi_3^\star\Psi_4+\Psi_1^\star\Psi_2^\star\Psi_3^\star\Psi_4+\Psi_1\Psi_2^\star\Psi_3^\star\Psi_4^\star+\Psi_1^\star\Psi_2^\star\Psi_3^\star\Psi_4^\star+\Psi_1\Psi_2\Psi_3\Psi_4^\star+\Psi_1\Psi_2\Psi_3^\star\Psi_4^\star\\
&\quad+\Psi_1^\star\Psi_2\Psi_3\Psi_4^\star+\Psi_1^\star\Psi_2\Psi_3^\star\Psi_4^\star\\
&= [\Psi_1\Psi_2\Psi_4(\Psi_3+\Psi_3^\star)]+[\Psi_1^\star\Psi_2\Psi_4(\Psi_3+\Psi_3^\star)]+[\Psi_2^\star\Psi_3\Psi_4(\Psi_1+\Psi_1^\star)]+[\Psi_2^\star\Psi_3\Psi_4^\star(\Psi_1+\Psi_1^\star)]\\
&\quad+[\Psi_2^\star\Psi_3^\star\Psi_4(\Psi_1+\Psi_1^\star)]+[\Psi_2^\star\Psi_3^\star\Psi_4^\star(\Psi_1+\Psi_1^\star)](\text{Note } \Psi_2^\star\equiv 1\text{on } \Psi_3^\star\Psi_4^\star,\Psi_3^\star\Psi_4)+[\Psi_1\Psi_2\Psi_4^\star(\Psi_3+\Psi_3^\star)]\\
&\quad+[\Psi_1^\star\Psi_2\Psi_4^\star(\Psi_3+\Psi_3^\star)]\\
&= \Psi_5^P+\Psi_6^P+\Psi_7^P+\Psi_3^P+\Psi_8^P+\Psi_4^P+\Psi_1^P+\Psi_2^P.
\end{aligned}$$

Remark 4.1. The above generalized product method constructing PU functions can be applied to most practical background meshes for meshless methods. The product partition of unity is one of the most simple effective partition of unity for meshless methods, especially for the patchwise RPPM (reproducing polynomial particle method) to be described in the following section.

5. Numerical examples

5.1. Patchwise reproducing polynomial particle methods (RPPM)

Consider a model problem

$$-\Delta u+u=f\quad\text{in }\Omega, \tag{38}$$
$$u=g_d \text{ along } \Gamma_D, \tag{39}$$
$$\nabla u\cdot\mathbf{n}=g_t \text{ along } \Gamma_N, \tag{40}$$

where Ω is a polygonal domain in $\mathbb{R}^d$, $\mathbf{n}$ is the outward normal vector along Γ_N and $\Gamma_D\cup\Gamma_N=\partial\Omega$. We assume that the partition of Ω into patches and the partition of unity for Ω are those constructed in the previous section. Then the variational formulation of the model problem is: find $u\in H^1(\Omega)$ such that $u=g_d$ on Γ_D and

$$\int_\Omega\nabla v\cdot\nabla u d\Omega-\int_{\Gamma_D}v\nabla u\cdot\mathbf{n}d\Gamma+\int_\Omega uvd\Omega=\int_\Omega vfd\Omega+\int_{\Gamma_N}vg_td\Gamma, \tag{41}$$

for all $v\in H^1(\Omega)$.

Now, the patchwise RPPM [21,22] for a numerical solution for this problem in two-dimensional case is described as follows:

1. (Dividing Ω into patches). To construct PU functions with flat-top and smooth local approximation functions for numerical solutions of (41), the domain Ω is partitioned into large quadrangular patches and large triangular patches, denoted by Q_I, by using lines and rays satisfying rules in Theorem 4.1. Unlike conventional FEM mesh, the partition of Ω allows hanging nodes.
2. (Construction of partition of unity functions with flat-top). For $I=1,\ldots,M$, let Ψ_I^P be the generalized product PU function with flat-top corresponding to the patches in the background mesh, and ω_I be the support of Ψ_I^P.

3. (Planting particles). Let $\widehat{Q}^{(t)}$ and $\widehat{Q}^{(q)}$ be the reference triangular patch and the reference rectangular patch, respectively. Suppose $\hat{p}_k,\ k=1,\dots,N$ are arbitrary (or uniformly) distributed particles on the reference patches. Let $\varphi_I^t\,\widehat{Q}^{(t)} \to \omega_I$ (if Q_I is a triangular patch) and $\varphi_I^q\,\widehat{Q}^{(r)} \to \omega_I$ (if Q_I is a quadrangular patch) be the patch mappings. Then, through the patch mappings φ_I^t or φ_I^q, we are able to plant particles $p_{Ik},\ k=1,\dots,N_I$, in ω_I, that are distributed arbitrary (or uniformly). The conformal mapping $\varphi(z)=z^2$ can be used to generate singular local approximation functions [19,22]
4. (Local approximation functions). Suppose $\hat{h}_k,\ k=1,\dots,N$ are smooth RPP shape functions corresponding to the particles planted in the reference patch $\widehat{Q}$ that satisfy Kronecker delta property. Then these RPP shape functions on the reference patch can be used to build the local approximation functions on the physical patch ω_I as follows:

$$h_{Ik} = \hat{h}_k \circ \left(\varphi_I^q\right)^{-1} \ (\text{or } \hat{h}_k \circ \left(\varphi_I^t\right)^{-1}), \quad k=1,\dots,N_I. \tag{42}$$

For example, the following RPP shape function can be used for the reference shape functions $\hat{h}_k,\ k=1,\dots,N$:

- the smooth piecewise polynomial RPP shape functions introduced in [22] can be used. In this case, some particles can be located outside Q^r and particles are uniformly distributed.
- Tensor product of Lagrange interpolation functions corresponding to arbitrary spaced N_x numbers of nodes in the x-direction and those corresponding to arbitrary spaced N_y numbers of nodes in the y-direction. These are RPP shape functions and satisfy the Kronecker delta property. The complete polynomials of degree $k, \{x^i y^j | 0 \leqslant i,j \leqslant k\}$, can be used for local approximation functions, however they do not satisfy the Kronecker delta property.

5. (Smooth global RPP basis functions). Now the global approximation functions with compact support are constructed as follows:

$$\Phi_{Ik}(x,y) = \Psi_I^P(x,y)\cdot h_{Ik}(x,y), \quad I=1,\dots,M;\ k=1,\dots,N_I. \tag{43}$$

These global approximation functions are highly smooth and are corresponding to the particles:

$$p_{Ik}, \quad I=1,2,\dots,M;\ k=1,2,\dots,N_I.$$

We also assume that each set $\{h_{Ik}|k=1,\dots,N_I\}$, of local approximation functions has the polynomial reproducing property, and satisfies the Kronecker delta property at the particles corresponding to local approximation functions on ω_I, namely, $h_{(Ii)}(p_{(Ij)}) = \delta_i^j$.

6. (RPP approximation space). The vector space spanned by those approximation functions defined by (43), denoted by $\mathscr{V}^{RPP}$, is said to be *the RPP approximation space*. The Galerkin approximation method with use of this RPP approximation space $\mathscr{V}^{RPP}$ is coined to be *patchwise Reproducing Polynomial Particle Method* (patchwise RPPM) [22]. Then an RPP approximate solution of the model problem (38)–(40), can be written as

$$u^{RPP}(x,y) = \sum_I \sum_j c_{Ij}\cdot\Phi_{Ij}(x,y) = \sum_I \Psi_I^P(x,y)\left[\sum_j c_{Ij}\cdot h_{Ij}(x,y)\right]. \tag{44}$$

Since the PU functions Ψ_I^P used in (43) have flat-top, the associated stiffness matrix is expected to have a small condition number.

In order to show the effectiveness of the generalized product PU functions for meshless methods, in particular, the patchwise RPPM, we consider two numerical examples: one two-dimensional example and one three-dimensional example.

Example 5.1. Consider Poisson's equation $-\Delta u = f$ on the a polygonal domain Ω shown in Fig. 13, whose analytic solution is

$$u(x,y) = (1-x)ye^{x+y} \tag{45}$$

The Dirichlet boundary conditions are imposed along the entire boundary of Ω.

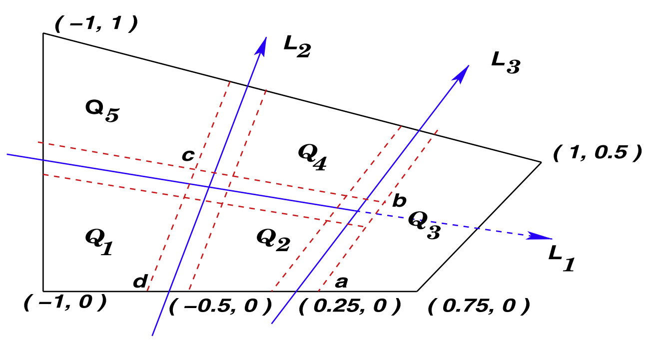

Fig. 13. Diagram of the domain for Example 5.1. The slopes of partitioning lines $L_1,\ L_2$, and L_3, respectively, are $-1/8$, $2/8$, and $4/8$. The support of the product PU function Ψ_2^P is the quadrangle $\overline{abcd}$.

Table 1
Two-dimensional Poisson's equation, $-\Delta u = f$, with analytic solution $u(x,y) = (1-x)ye^{x+y}$.

$-\Delta u = f$ on non-rectangular domain with essential BC				
RPP-order	$\|\text{Abs. err}\|_\infty$	$\|\text{Rel. err}\|_{eng}$	Computed Eng	Matrix Condition No.
2	6.65E−02	3.69E−02	9.7799002E−00	6.30175E+01
4	3.01E−04	1.71E−03	9.7932129E−00	3.48345E+04
6	5.75E−07	3.39E−05	9.7932415E−00	2.23594E+07
∞			9.7932416E−00	

From this example, we observe the following:

1. The absolute errors of computed solutions of Example 5.1 by patchwise RPPM with use of the generalized product PU functions are shown in Table 1.
2. $\delta = 0.05$ is used for those results in Table 1.
3. Unlike other meshless methods, the matrix condition numbers in Table 1 are small since the product PU functions have wide flat-top.
4. To deal with essential boundary conditions for meshless methods, the Lagrange multiplier methods, the penalty methods and Nitche's methods are commonly used in the literature. These methods modify the variational formulation which lead to large matrix condition numbers when the penalty parameters are increased. On the other hand, since our local approximation functions used in patchwise RPPM satisfy the Kronecker delta property, imposing essential boundary condition is almost the same as the conventional FEM. If the partitioning lines meet at a (convex or non-convex) corner point, we need to use almost everywhere partition of unity [21] as well as generalized product partition of unity. This issue on imposing Dirichlet boundary conditions in meshless methods will be elaborated in a forthcoming paper.

Example 5.2. Consider three-dimensional Poisson's equation $-\Delta u = f$ on the polyhedral domain Ω shown in Fig. 14, whose analytic solution is

$$u(x,y,z) = xyze^{x+y+z} \tag{46}$$

The Dirichlet boundary conditions are imposed along the entire boundary of Ω. For numerical test of this problem, we use $\delta = 0.05$ for the construction of the generalized product partition of unity.

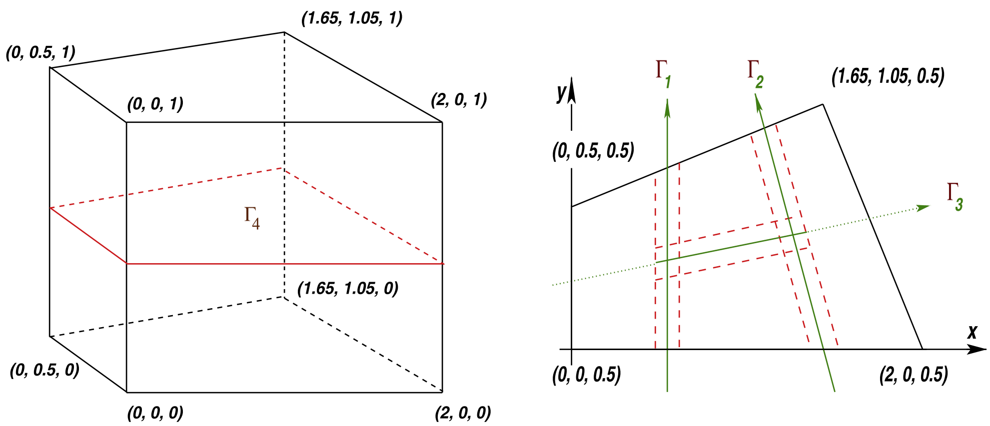

Fig. 14. (Left) Diagram of the domain for Example 5.2 and the horizontally partitioning plane Γ_4. (Right) Diagram of the plane section of the vertically partitioning planes Γ_1, Γ_2, Γ_3 of Ω.

Table 2
Three-dimensional Poisson's equation, $-\Delta u = f$, with analytic solution $u(x,y,z) = xyze^{x+y+z}$.

$-\Delta u = f$ on Polyheral domain with essential BC				
RPP-order	$\|\text{Abs. err}\|_\infty$	$\|\text{Rel. err}\|_{eng}$	Computed Eng	Matrix Condition No.
2	6.45E−01	4.99E−01	495.7366095938	1.36E+03
4	1.50E−03	3.32E−02	490.8555005157	9.65E+06
6	9.73E−08	8.59E−06	490.8505588268	1.80E+10
∞			490.8505586819	

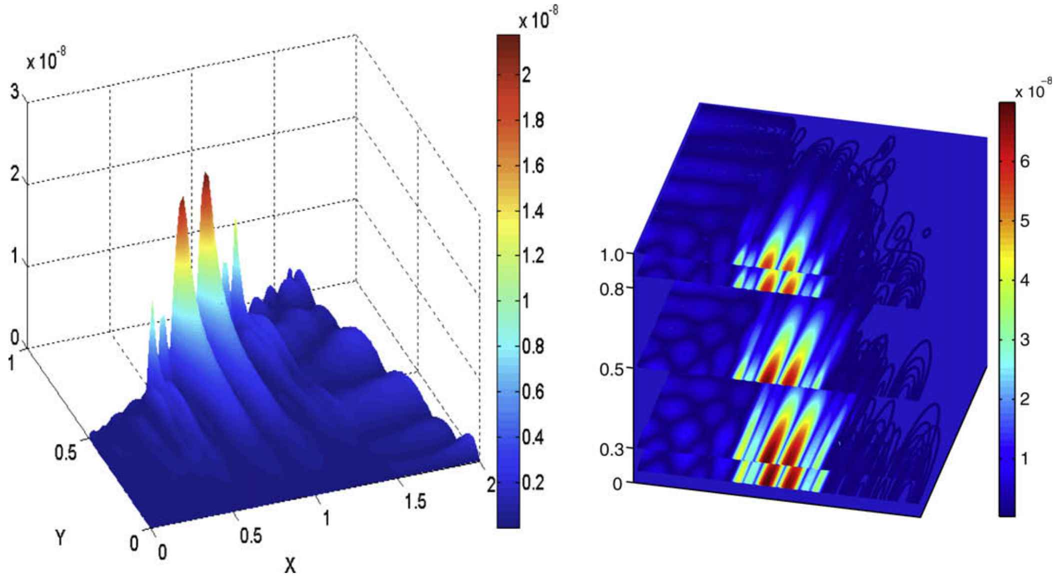

Fig. 15. (Left) Absolute errors on the plane section by the x—y plane when RPP order = 6. The largest error (9.73E−08) occurs along the vertically slated boundary (the north side boundary in Fig. 14). (Right) The contour curves of absolute errors (when RPP order = 6) on five different plane sections by the planes $z = 0,\ z = 0.3,\ z = 0.5,\ z = 0.8,\ z = 1$. The sections are rotated by 180° so that the larger errors can be shown on the front side.

In Table 2, we show the absolute errors in maximum norm and the relative errors in energy norm of computed solutions of Example 5.2 by patchwise RPPM with use of the generalized product PU functions. The absolute errors (when RPP = 6) on the horizontal slices of the domain shown in Fig. 14 are plotted in Fig. 15.

For brevity, in Fig. 14, the three-dimensional domain Ω is partitioned into eight hexahedral patches by the three vertical planes $\Gamma_1,\ \Gamma_2,\ \Gamma_3$ and one horizontal plane Γ_4. However, we can partition the domain Ω into hexahedral patches by vertically tilted planes, $\Gamma_1,\ \Gamma_2,\ \Gamma_3$, and horizontally tilted plane Γ_3 to get similar results.

6. Concluding remarks

The proposed method constructing a highly smooth piecewise polynomial (closed form) partition of unity with flat-top is simple and efficient for the p-version type partition of unity FEM (PUFEM). However, it may have limitations to use for the adaptive meshless methods because the width δ of the non-flat-top parts, ω^{nflt}, can not be too small for the convergence of meshless methods. This is because the error estimates proved in [21] are as follows:

$$\|u - u^{approx}\|_{L^2(\Omega)} \leqslant \sqrt{\kappa}\left(\sum_{I=1}^{M}\left(\varepsilon_I^{(0)}\right)^2\right)^{1/2} \quad \text{and} \quad \|\nabla(u - u^{approx})\|_{L^2(\Omega)} \leqslant \sqrt{2\kappa}\left(\sum_{I=1}^{M}\left(\frac{C}{\delta}\right)^2\left(\varepsilon_{I,nflat}^{(0)}\right)^2 + \left(\varepsilon_I^{(1)}\right)^2\right)^{1/2},$$

when

1. the domain Ω is partitioned into M patches, ω_I is the supports of PU functions Ψ_I^p, for $I = 1, \ldots, M$, and $\mathrm{card}\{I | x \in \omega_I\} \leqslant \kappa$, for all $x \in \Omega$;
2. we are given a collection of local approximation spaces $\mathscr{V}_I \subset H^1(\Omega)$ that have the following local approximation properties: on each $\omega_I,\ I = 1, \ldots, M$, the function u can be approximated by a function $v_I \in \mathscr{V}_I$ such that $\|u - v_I\|_{0,\omega_I} \leqslant \epsilon_I^{(0)},\ \|u - v_I\|_{0,\omega_I^{nflat}} \leqslant \epsilon_{I,nflat}^{(0)},\ \|u - v_I\|_{1,\omega_I} \leqslant \epsilon_I^{(1)}$.

In using the Shepard PU functions, it is not easy to determine automatically the intersection of the supports of adjacent PU functions for numerical integrals. Whereas it is easy to determine the intersection of the supports of adjacent product PU functions, their flat-top parts, and their non-flat-top parts for numerical integrals of the product PU functions in two-dimensional cases. However, it is not simple to determine automatically their intersections in three-dimensional cases.

References

[1] S. Atluri, S. Shen, The Meshless Method, Tech Science Press, 2002.
[2] I. Babuška, U. Banerjee, J.E. Osborn, Survey of Meshless and Generalized Finite Element Methods: A Unified Approach, Acta Numerica, Cambridge Press, 2003. pp. 1–125.
[3] P.G. Ciarlet, The Finite Element Method for Elliptic Problems, North-Holland, 1978.

[4] C.A. Duarte, J.T. Oden, An hp adaptive method using clouds, Comput. Meth. Appl. Mech. Eng. 139 (1996).
[5] M. Griebel, M.A. Schweitzer, A particle-partition of unity methods part VII: adaptivity, Meshfree Methods for Partial Differential Equations, Lecture Notes in Computer Science and Engineering III, vol. 57, Springer, 2007.
[6] W. Han, X. Meng, Error analysis of reproducing kernel particle method, Comput. Meth. Appl. Mech. Eng. 190 (2001) 6157–6181.
[7] W.K. Liu, Y. Chen, R.A. Uras, C.T. Chang, Generalized multiple scale reproducing kernel particle methods, Comput. Meth. Appl. Mech. Eng. 139 (1996) 91–158.
[8] S. Li, W.K. Liu, Moving least square reproducing kernel method. Part II: Fourier analysis, Comput. Meth. Appl. Mech. Eng. 139 (1996) 159–194.
[9] W. Han, W.K. Liu, Flexible Piecewise Approximations Based on Partition of Unity, Advances in Computational Mathematics, Kluwer Academic Publishers, Netherlands, 2004.
[10] H. Lu, S. Li, D.C. Simkins, W.K. Liu, J. Cao, Reproducing kernel element method. Part III. Generalized enrichment and applications, Comput. Meth. Appl. Mech. Eng. 193 (2004) 9891011.
[11] D.C. Simkins, S. Li, H. Lu, W.K. Liu, Reproducing kernel element method. Part IV. Globally conforming $C^n(n > 1)$ triangular hierarchy, Comput. Meth. Appl. Mech. Eng. 193 (2004) 10131034.
[12] S. Li, W.K. Liu, Meshfree Particle Methods, Springer-Verlag, 2004.
[13] S. Li, H. Lu, W. Han, W.K. Liu, D.C. Simkins Jr., Reproducing kernel element method. Part II. Globally conforming I^m/C^n hierarchies, Comput. Meth. Appl. Mech. Eng. 193 (2004) 953–987.
[14] W.K. Liu, W. Han, H. Lu, S. Li, J. Cao, Reproducing kernel element method: part I. Theoretical formulation, Comput. Meth. Appl. Mech. Eng. 193 (2004) 933–951.
[15] W.K. Liu, S. Jun, Y.F. Zhang, Reproducing kernel particle methods, Int. J. Numer. Meth. Fluids 20 (1995) 1081–1106.
[16] W.K. Liu, S. Liu, S. Jun, S. Li, J. Adee, T. Belytschko, Reproducing kernel particle methods for structural dynamics, Int. J. Numer. Meth. Eng. 38 (1995) 1655–1679.
[17] W.K. Liu, S. Li, T. Belytschko, Moving least square reproducing kernel method part I: methodology and convergence, Comput. Meth. Appl. Mech. Eng. 143 (1997) 422–453.
[18] J.M. Melenk, I. Babuška, The partition of unity finite element method: theory and application, Comput. Meth. Appl. Mech. Eng. 139 (1996) 239–314.
[19] H.-S. Oh, I. Babuška, The method of auxiliary mapping for the finite element solutions of plane elasticity problems containing singularities, J. Comput. Phys. 121 (1995) 193–212.
[20] H.-S. Oh, B.S. Jang, Y. Jou, The weighted Ritz–Galerkin method for elliptic boundary value problems on unbounded domains, Numer. Meth. Partial Differen. Equat. 19 (2003) 301–326.
[21] H.-S. Oh, J.W. Jeong, Almost everywhere partition of unity to deal with essential boundary conditions in meshless methods, Comput. Meth. Appl. Mech. Eng. 198 (2009) 3299–3312.
[22] H.-S. Oh, J.W. Jeong, Reproducing polynomial (singularity) particle methods and adaptive meshless methods for two-dimensional elliptic boundary value problems, Comput. Meth. Appl. Mech. Eng. (2008).
[23] H.-S. Oh, J.G. Kim, W.T. Hong, The piecewise polynomial partition of unity shape functions for the generalized finite element methods, Comput. Meth. Appl. Mech. Eng. 197 (2008) 3702–3711.
[24] H.-S. Oh, J.G. Kim, J.W. Jeong, The closed form reproducing polynomial particle shape functions for meshfree particle methods, Comput. Meth. Appl. Mech. Eng. 196 (2007) 3435–3461.
[25] 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 particles methods, Comput. Mech. 40 (2007) 569–594.
[26] H.-S. Oh, J.W. Jeong, J.G. Kim, The reproducing singularity particle shape function for problems containing singularities, Comput. Mech. 41 (2007) 135–157.
[27] T. Strouboulis, K. Copps, I. Babuska, Generalized finite element method, Comput. Meth. Appl. Mech. Eng. 190 (2001) 4081–4193.
[28] T. Strouboulis, L. Zhang, I. Babuska, Generalized finite element method using mesh-based handbooks: application to problems in domains with many voids, Comput. Meth. Appl. Mech. Eng. 192 (2003) 3109–3161.
[29] B. Szabo, I. Babuska, Finite Element Analysis, John Wiley, 1991.