

# Polygonal spline spaces and the numerical solution of the Poisson equation

Michael S. Floater\*      Ming-Jun Lai†

March 9, 2015

## Abstract

It is known that generalized barycentric coordinates (GBCs) can be used to form Bernstein polynomial-like functions over a polygon with any number of sides. We propose to use these functions to form a continuous polygonal spline space of order  $d$  over a partition consisting of polygons which is able to reproduce all polynomials of degree  $d$ . Locally supported basis functions (polygonal finite elements) for the space are constructed for order  $d \geq 2$ . Our quadratic elements are simpler than the ‘serendipity’ elements that have appeared in the recent literature. Also, the basis functions are fewer than those of the virtual element method. We use them to solve interpolation problems. In addition, we implement them for the numerical solution of the Poisson equation on two special types of non-triangular partitions to present a proof of concept for solving the equation over mixed partitions. Numerical solutions based on quadrangulations and pentagonal partitions are demonstrated to show the efficiency and effectiveness of these polygonal splines. They appear to better (using less degrees of freedom to find a more accurate solution) than the traditional continuous polynomial finite element method.

**Key Words:** Polygonal Finite Elements, Generalized Barycentric Coordinates, Numerical Solution of PDE

## 1 Introduction

In this paper, we are interested in constructing piecewise-defined functions over an arbitrary polygonal partition  $\Delta$  of a domain and present an application. It is known that one can define bivariate spline functions over a triangulation  $\triangle$  consisting of triangles and use them for the numerical solution of partial differential equations (PDEs), scattered data fitting, etc., see, e.g. [2], [14], [15], and [16]. If a triangulation  $\triangle$  is replaced by a collection  $\Delta$  of polygons, say convex quadrilaterals or pentagons or even a mix of arbitrary polygons such as a Voronoi diagram of a bounded domain, can we extend the bivariate splines to  $\Delta$ ? The answer is yes. The method of virtual finite elements is one approach, see [5] and [6] based on polynomials and see [18] based on GBCs of first order. The construction of serendipity quadratic finite elements in [22] provides another approach, using GBCs. Motivated by these studies we use GBCs to define polygonal splines of order  $d \geq 2$  on a partition  $\Delta$  of convex polygons. Some advantages of these splines are 1) we can partition a polygonal domain  $\Omega$  using various  $n$ -sided polygons including triangles, even though triangles are very convenient for partitioning  $\Omega$ ; 2) as there are various GBCs which include polynomial, rational, trigonometric, logarithmic functions,

---

\*Department of Mathematics, University of Oslo, PO Box 1053, Blindern, 0316 Oslo, Norway, *email: michael@math.uio.no*

†Department of Mathematics, University of Georgia, Athens, GA *email: mjlai@math.uga.edu*

etc., we are able to use more general functions than polynomials to approximate known and unknown functions such as solutions to PDEs (see a summary of various GBCs in a recent survey article [10]); 3) there are redundant functions over each polygon when the side number  $n > 3$  which may be useful for certain applications; and 4) our numerical solutions of the Poisson equation indicate that they are sometimes better than the standard finite element method; see Section 5.2.

The potential advantages of this method come, however, at a price because

- (i) the number of GBC-based functions on each polygon grows rapidly as the number of sides of the polygon increases, and
- (ii) the functions are not in general linearly independent, making it difficult to choose suitable degrees of freedom. For example, the number of quadratic functions generated by GBCs in an  $n$ -gon is  $n(n + 1)/2$ , which grows quadratically with  $n$ .
- (iii) The evaluation of functions based on GBCs is not as efficient as for polynomials.

One way to curtail the redundancy of the polygonal Bernstein-Bézier functions is to use a reduced space of functions still form a continuous polygonal spline when pieced together, and still reproduce all polynomials of degree  $\leq d$ . This is what we do in this paper. This is not a new idea and such reduced elements were constructed in the quadratic case in [22], based on the observation that many of the quadratic functions are zero on the boundary of the polygon, and thus do not contribute to inter-element continuity. By forming various linear combinations of the quadratic functions they obtained ‘reduced’ quadratic elements, having only  $2n$  degrees of freedom, but retaining inter-element continuity. Their construction includes the usual 8-node serendipity element on squares and parallelograms as a special case [23] and [1]. However, their construction is not trivial. One of the purposes of this paper is to offer a simpler way of constructing quadratic elements of this kind. Our main idea is to use not only GBCs but also *local* barycentric coordinates with respect to successive triples of vertices. Similar to [22], this generates  $2n$  functions in the quadratic case. From these functions we also recover the quadratic rational interpolatory element constructed by Wachspress [24], but in explicit form.

To obtain polygonal functions of higher order we specialize to the rational GBCs of Wachspress and use additional degrees of freedom in the interior of the polygon. The dimension of the space of order  $d$  over an  $n$ -gon is then

$$dn + (d - 1)(d - 2)/2. \tag{1}$$

The space is able to reproduce all polynomials of degree  $d$ . This dimension is smaller than that of the virtual finite element method to be explained in Section 4.1. See Remark 1. Once we have these functions over each polygon, we can piece them together to have a polygonal spline space  $S_d(\Delta)$  over a collection  $\Delta$  of mixed polygons  $P_n$  with various  $n \geq 3$ . Hence, the dimension of the polygonal spline space of degree  $d$  over a mixed partition  $\Delta$  can be established as in Theorem 4 when  $d = 2$  for any generalized barycentric coordinates and when  $d \geq 3$  for Wachspress coordinates.

When  $P$  is a triangle ( $n = 3$ ), the number in (1) is  $(d+2)(d+1)/2$ , which is the number of Bernstein polynomials of degree  $d$  in  $P$ . When  $\Delta$  is a triangulation, the dimension in (30) in Theorem 4 agrees with the dimension of the continuous spline space of degree  $d$  explained in [14].

In addition to the construction of a locally supported basis, i.e., finite elements, we also study the standard interpolation problem using these polygonal spline functions, and as one might expect, we can use the space  $S_d(\Delta)$  to do interpolation.

Finally, we describe how to implement these functions for the numerical solution of the Poisson equation. Our numerical results demonstrate the efficiency and effectiveness of this approach. In particular, we introduce a new concept called efficiency which is the product of the number of polygons and the  $L_2$  norm of the numerical solution against the exact solution. The smaller the efficiency

value the better. In this sense, our numerical results show that the quadratic polygonal splines over quadrilaterals are better than the standard quadratic polynomial elements for the same size of partition. In addition, we introduce a novel refinement scheme for pentagons and show that the quadratic polygonal spline solution over pentagonal partition is better than that of quadratic polygonal splines over quadrilaterals and hence is even better than the standard quadratic finite element method over triangulations. See Section 5.2.

The major contributions of this paper are as follows:

1. We provide a construction of serendipity quadratic finite elements that is simpler than the one in [22] and a construction of higher order finite elements using Wachspress coordinates.
2. The number of basis functions per polygon is smaller than those of the virtual elements of [5] and [6] for order  $d \geq 2$ .
3. We have used our elements to solve the Poisson equation numerically, which extends the study in [18] where polygonal finite elements of order  $d = 1$  were used.
4. We introduce a refinement scheme for pentagons which is new to the best of our knowledge. Our numerical experiments show that the quadratic polygonal splines based on both Wachspress coordinates and mean value coordinates over uniformly refined pentagonal partitions are more efficient than the traditional quadratic finite elements over uniformly refined triangulations.

However, the generation of the mass and stiffness matrices for these polygonal splines as well as their evaluation takes much more time than for polynomial finite elements. More research needs to be done to overcome this difficulty so that the polygonal spline method can be useful for practice.

## 2 Background on GBCs

There are many ways to define barycentric coordinates in a polygon with  $n$  sides,  $n \geq 3$ ; see [10]. We will only consider convex polygons in this paper. Let  $P = \langle \mathbf{v}_1, \dots, \mathbf{v}_n \rangle$  be a convex polygon. Any functions  $\phi_i : P \rightarrow \mathbb{R}$ ,  $i = 1, \dots, n$ , will be called generalized barycentric coordinates (GBCs) if, for all  $\mathbf{x} \in P$ ,  $\phi_i(\mathbf{x}) \geq 0$  and

$$\sum_{i=1}^n \phi_i(\mathbf{x}) = 1, \quad \text{and} \quad \sum_{i=1}^n \phi_i(\mathbf{x}) \mathbf{v}_i = \mathbf{x}. \quad (2)$$

When  $P$  is a triangle, the coordinates  $\phi_1, \phi_2, \phi_3$  are the usual ones. For  $n \geq 3$ , the  $\phi_i$  are not uniquely determined by (2) but they share the basic property that they are piecewise linear on the boundary of  $P$ :  $\phi_i(\mathbf{v}_j) = \delta_{ij}$  and

$$\phi_i((1 - \mu)\mathbf{v}_j + \mu\mathbf{v}_{j+1}) = (1 - \mu)\phi_i(\mathbf{v}_j) + \mu\phi_i(\mathbf{v}_{j+1}), \quad \mu \in [0, 1]. \quad (3)$$

Here, and throughout the paper, we use the cyclic indexing,  $\phi_{n+1} = \phi_1$ , and so on.

For convenience, let us describe three well known GBCs, which are all smooth in the interior of  $P$ .

**Example 1** (Inverse bilinear coordinates). *For  $n = 4$ , the quadrilateral  $P$  is the image of a bilinear map from the unit square  $[0, 1]^2$ , and for each  $\mathbf{x} \in P$ , there exists a unique pair  $(\alpha, \beta) \in [0, 1]^2$  such that*

$$\mathbf{x} = (1 - \alpha)(1 - \beta)\mathbf{v}_1 + \alpha(1 - \beta)\mathbf{v}_2 + \alpha\beta\mathbf{v}_3 + (1 - \alpha)\beta\mathbf{v}_4.$$

Thus the four functions

$$\phi_1(\mathbf{x}) = (1 - \alpha)(1 - \beta), \quad \phi_2(\mathbf{x}) = \alpha(1 - \beta), \quad \phi_3(\mathbf{x}) = \alpha\beta, \quad \phi_4(\mathbf{x}) = (1 - \alpha)\beta \quad (4)$$

are GBCs. An explicit formula is given in [10].

**Example 2** (Wachspress (rational) coordinates). For general  $n \geq 3$ , let  $\mathbf{n}_i \in \mathbb{R}^2$  be the outward unit normal to the edge  $e_i = [\mathbf{v}_i, \mathbf{v}_{i+1}]$ ,  $i = 1, \dots, n$ , and for any  $\mathbf{x} \in P$  let  $h_i(\mathbf{x})$  be the perpendicular distance of  $\mathbf{x}$  to the edge  $e_i$ , so that

$$h_i(\mathbf{x}) = (\mathbf{v}_i - \mathbf{x}) \cdot \mathbf{n}_i = (\mathbf{v}_{i+1} - \mathbf{x}) \cdot \mathbf{n}_i. \quad (5)$$

Let

$$w_i(\mathbf{x}) = d_i \prod_{j \neq i-1, i} h_j(\mathbf{x}), \quad (6)$$

where  $d_i$  is the scalar cross product

$$d_i = \mathbf{n}_{i-1} \times \mathbf{n}_i = \begin{vmatrix} n_{i-1}^x & n_i^x \\ n_{i-1}^y & n_i^y \end{vmatrix},$$

and  $\mathbf{n}_j = (n_j^x, n_j^y)$ . Then, with

$$W = \sum_{j=1}^n w_j, \quad (7)$$

the functions  $\phi_i = w_i/W$ , are GBCs, which are rational of degree  $(n-2, n-3)$ .

**Example 3** (Mean value coordinates). For general  $n \geq 3$ , and for  $\mathbf{x}$  in the interior of  $P$ , let  $\alpha_i$  be the angle at  $\mathbf{x}$  of the triangle  $\langle \mathbf{v}_i, \mathbf{x}, \mathbf{v}_{i+1} \rangle$ ,  $i = 1, \dots, n$ , and define

$$w_i(\mathbf{x}) = \frac{\tan(\alpha_{i-1}/2) + \tan(\alpha_i/2)}{\|\mathbf{x} - \mathbf{v}_i\|},$$

with  $\|\cdot\|$  the Euclidean norm in  $\mathbb{R}^2$ . Then the functions  $\phi_i = w_i / \sum_{j=1}^n w_j$ , which extend continuously to the boundary of  $P$ , are GBCs. See [9].

Various other smooth GBCs have been constructed such as Gordon-Wixom coordinates and maximum entropy coordinates. We note also that both Wachspress and mean value coordinates have been generalized to 3D (polyhedra); see [10] for a summary.

### 3 Polygonal Bernstein-Bézier functions

We now study generalizations of Bernstein-Bézier polynomials to a convex polygon, dealing first with the ‘full’ space of functions that has been studied in the literature, and then construct a reduced space. We mainly use the reduced space to construct corresponding polygonal splines in the next section.

### 3.1 The full space

It is well-known that the standard barycentric coordinates on a triangle in  $\mathbb{R}^2$  can be used to generate Bernstein-Bézier polynomials of degree  $d$  via the multinomial theorem (cf. [8]). Similarly, on a convex polygon in  $\mathbb{R}^2$  one can use GBCs to generate analogous functions, whose span will contain polynomials of degree  $\leq d$ . Such functions were studied by Loop and DeRose [17] with a view to constructing multi-sided surface patches, ‘S-patches’, for modelling geometry.

For a convex polygon  $P_n$  with  $n \geq 3$  sides, let  $\phi_1, \dots, \phi_n$  be a set of GBCs. For any  $d \geq 0$ , and any multi-index  $\mathbf{j} = (j_1, \dots, j_n) \in \mathbb{N}_0^n$ , with  $|\mathbf{j}| := j_1 + \dots + j_n = d$ , let

$$B_{\mathbf{j}}^d(\mathbf{x}) = \frac{d!}{j_1! \cdots j_n!} \phi_1^{j_1}(\mathbf{x}) \cdots \phi_n^{j_n}(\mathbf{x}), \quad \mathbf{x} \in P_n. \quad (8)$$

In the triangular case,  $n = 3$ ,  $B_{\mathbf{j}}^d$  is a Bernstein-Bézier polynomial of degree  $d$  (cf. [14]). For  $n > 3$ ,  $B_{\mathbf{j}}^d$  is no longer a polynomial in general and its function type will depend on the choice of the GBCs  $\phi_i$ . Nevertheless, for any  $n$ , the linear space  $\Phi_d(P_n)$  of functions of the form

$$s(\mathbf{x}) = \sum_{|\mathbf{j}|=d} c_{\mathbf{j}} B_{\mathbf{j}}^d(\mathbf{x}), \quad \mathbf{x} \in P_n, \quad (9)$$

with  $c_{\mathbf{j}} \in \mathbb{R}$ , inherits two important properties from the  $\phi_i$ .

(P1)  $\Pi_d \subset \Phi_d(P_n)$ , where  $\Pi_d$  is the space of polynomials of degree  $\leq d$ . This is because, due to (2), any linear polynomial can be expressed as a linear combination of the  $\phi_i$ .

(P2) Due to (3), the function  $s$  in (9) is a univariate polynomial of degree  $\leq d$  on each edge of the polygon.

In general, the functions  $B_{\mathbf{j}}^d$ ,  $|\mathbf{j}| = d$ , are not linearly independent, and, as we will see, the dimension of  $\Phi_d(P_n)$  is less than the number of these polygonal Bernstein-Bézier functions. Indeed, let us consider the case  $d = 2$ . Among the pairwise products  $\phi_i \phi_j$ ,  $i, j = 1, \dots, n$ , there are  $n$  products of the form  $\phi_i^2$ ,  $i = 1, \dots, n$  and  $\binom{n}{2}$  of the form  $\phi_i \phi_j$ ,  $i \neq j$ . So the total number of  $B_{\mathbf{j}}^2$ ,  $|\mathbf{j}| = 2$ , is

$$D_n = n + \binom{n}{2} = \binom{n+1}{2}. \quad (10)$$

So,

$$D_3 = 6, \quad D_4 = 10, \quad D_5 = 15, \quad D_6 = 21,$$

and so on. However, the dimension of  $\Phi_2(P_n)$  is lower than  $D_n$  when  $n \geq 4$ . We propose an open problem:

**What is the dimension of  $\Phi_d(P_n)$  for various GBCs and  $d \geq 2$  and any  $n > 3$ ?**

For  $d = 2$  and for Wachspress coordinates, we can show

**Theorem 1.**

$$\dim(\Phi_2(P_n)) = 2n + \frac{(n-2)(n-3)}{2}. \quad (11)$$

*Proof.* The products  $\phi_i^2$ ,  $\phi_i \phi_{i+1}$ ,  $i = 1, \dots, n$ , are linearly independent due to their values at  $\mathbf{v}_i$ ,  $(\mathbf{v}_i + \mathbf{v}_{i+1})/2$ ,  $i = 1, \dots, n$ . Therefore we can express  $\Phi_2(P_n)$  as

$$\Phi_2(P_n) = B_n \oplus I_n,$$

where

$$B_n = \text{span}\{\phi_i^2, \phi_i\phi_{i+1}, i = 1, \dots, n\} \quad \text{and} \quad I_n = \text{span}\{\phi_i\phi_j : |i - j| \geq 2\}.$$

The sum is direct because all the functions in  $I_n$  are zero on the entire boundary  $\partial P_n$  while the only function in  $B_n$  with this property is 0.

Since  $\dim(B_n) = 2n$ , it remains to determine the dimension of  $I_n$ . Recalling  $h_j$  in (6), let  $b$  be the bubble function

$$b(\mathbf{x}) = \prod_{k=1}^n h_k(\mathbf{x}). \quad (12)$$

When  $|i - j| \geq 2$ , the four indices  $i - 1, i, j - 1, j$  are distinct and so from (6),

$$\phi_i\phi_j = \frac{w_i w_j}{W^2} = \frac{bd_i d_j \nu_{ij}}{W^2},$$

where

$$\nu_{ij} = \prod_{k \neq i-1, i, j, j-1} h_k. \quad (13)$$

Therefore,  $\dim(I_n) = \dim(H_n)$ , where

$$H_n = \text{span}\{\nu_{ij} : |i - j| \geq 2\}.$$

Observe that  $\nu_{ij} \in \Pi_{n-4}$  since  $h_k$  is a linear polynomial and so  $H_n \subset \Pi_{n-4}$ . In fact we will show that

$$H_n = \Pi_{n-4}, \quad (14)$$

from which we deduce that  $\dim(H_n) = (n - 2)(n - 3)/2$  which yields (30). We now present a proof of (14). For  $n = 4$  it is easy as (13) is an empty product, i.e.,  $\nu_{13} = \nu_{24} = 1$ . For  $n \geq 5$ , notice that each edge of the polygon can be parallel with at most one other edge. Therefore, any three lines  $h_k = 0$  intersect in at least two points from which it follows that any three  $h_k$  are linearly independent, and thus form a basis for  $\Pi_1$ . In the case  $n = 5$ , the  $\nu_{ij}$  are linear polynomials, which are  $\nu_{i,i+2} = h_{i+3}$ ,  $i = 1, \dots, 5$ . Since any three of them span  $\Pi_1$  (14) follows.

Finally, for  $n \geq 6$ , consider the monomial  $x^{n-4}$  where  $\mathbf{x} = (x, y)$ . For any distinct  $r, s, t$  in  $\{1, \dots, n\}$ , we can write it as

$$x^{n-4} = x x^{n-5} = (c_r h_r + c_s h_s + c_t h_t) x^{n-5} = c_r h_r x^{n-5} + c_s h_s x^{n-5} + c_t h_t x^{n-5},$$

for coefficients  $c_r, c_s, c_t \in \mathbb{R}$ , and a similar decomposition can be applied to any other monomial of degree  $n - 4$ , and so if, for any  $r = 1, \dots, n$ , we can express  $x^{n-5}$  as a linear combination of the polynomials

$$\mu_{i,j} = \prod_{k \neq r, i-1, i, j-1, j} h_k,$$

where  $r, i - 1, i, j - 1, j$  are distinct, equation (14) will follow. Thus it is sufficient to show that the polynomials  $\mu_{ij}$  span  $\Pi_{n-5}$ , and we may assume that  $r$  is fixed and  $r = n$ . The possible indices of the  $\mu_{ij}$ , with  $i < j$ , are  $i = 2, \dots, n - 3$  and  $j = i + 2, \dots, n - 1$ , and thus the number of  $\mu_{ij}$  is  $(n - 4)(n - 3)/2$ , which is precisely the dimension of  $\Pi_{n-5}$ . So, showing that the  $\mu_{ij}$  span  $\Pi_{n-5}$  is equivalent to showing that they are linearly independent.

We thus complete the proof by showing that the  $\mu_{ij}$  are linearly independent. We prove this by induction on  $n$ ,  $n \geq 5$ . It is true for  $n = 5$  since there is only one  $\mu_{ij}$ , namely  $\mu_{2,4}$ . For  $n > 5$  suppose that

$$\sum_{i=2}^{n-3} \sum_{j=i+2}^{n-1} c_{ij} \mu_{i,j}(\mathbf{x}) = 0, \quad \mathbf{x} \in \mathbb{R}^2. \quad (15)$$

Let  $l_k$  denote the line  $h_k = 0$  for each  $k = 1, \dots, n-1$ . Suppose first that  $l_1$  and  $l_{n-1}$  are not parallel, and let  $\mathbf{x} = l_1 \cap l_{n-1}$ . Then all the  $\mu_{ij}$  are zero at  $\mathbf{x}$  except  $\mu_{2,n-1}$ , and so  $c_{2,n-1} = 0$  in (15). Otherwise,  $l_1$  and  $l_{n-1}$  are parallel, in which case let  $\mathbf{x}_0 = (\mathbf{v}_1 + \mathbf{v}_n)/2$  and let  $\mathbf{d}$  be the unit vector parallel to  $l_1$  and  $l_{n-1}$  in the direction away from the polygon, and consider the point  $\mathbf{x} = \mathbf{x}_0 + \lambda \mathbf{d}$  as  $\lambda \rightarrow \infty$ . From (5),  $h_1(\mathbf{x})$  and  $h_{n-1}(\mathbf{x})$  are constant in  $\lambda$ , while

$$h_k(\mathbf{x}) \sim -\lambda \mathbf{d} \cdot \mathbf{n}_k, \quad k = 2, \dots, n-2.$$

Therefore, since

$$\frac{\mu_{i,j}}{\mu_{2,n-1}} = \frac{h_1 h_2 h_{n-2} h_{n-1}}{h_{i-1} h_i h_{j-1} h_j},$$

we see that if  $(i, j) \neq (2, n-1)$ ,

$$\frac{\mu_{i,j}(\mathbf{x})}{\mu_{2,n-1}(\mathbf{x})} \rightarrow 0 \quad \text{as } \lambda \rightarrow \infty,$$

and dividing (15) by  $\mu_{2,n-1}(\mathbf{x})$  and letting  $\lambda \rightarrow \infty$  shows again that  $c_{2,n-1} = 0$ .

Next, let  $\mathbf{x} = l_2 \cap l_{n-1}$  if  $l_2$  and  $l_{n-1}$  are not parallel. Then all the  $\mu_{ij}$  are zero at  $\mathbf{x}$  except  $\mu_{2,n-1}$  and  $\mu_{3,n-1}$ . Since  $c_{2,n-1} = 0$  this means that  $c_{3,n-1} = 0$  as well. The previous argument for parallel lines also applies to show that  $c_{3,n-1} = 0$  also if  $l_2$  and  $l_{n-1}$  are parallel. We now continue in this way with the pairs of lines  $l_i$  and  $l_{n-1}$  for  $i = 3, \dots, n-4$ , to show sequentially that  $c_{i,n-1} = 0$  for  $i = 4, \dots, n-3$ . This reduces (15) to

$$h_{n-1}(\mathbf{x}) \sum_{i=2}^{n-4} \sum_{j=i+2}^{n-2} c_{ij} \hat{\mu}_{i,j}(\mathbf{x}) = 0, \quad \mathbf{x} \in \mathbb{R}^2, \quad (16)$$

where the  $\hat{\mu}_{ij}$  are the analogs of  $\mu_{ij}$  for the convex polygon with vertices  $\mathbf{v}_1, \dots, \mathbf{v}_{n-1}$ . By the induction hypothesis, the  $\hat{\mu}_{ij}$  are linearly independent, and therefore the  $c_{ij}$  in the sum are zero.  $\square$

Thus at least for Wachspress coordinates, the dimension of  $\Phi_2(P_n)$ , while lower than the number of functions  $B_{\mathbf{j}}^2$ ,  $|\mathbf{j}| = 2$ , nevertheless grows quadratically with  $n$ . In fact, in this case, the redundancy of the  $B_{\mathbf{j}}^2$ ,  $|\mathbf{j}| = 2$ , is

$$D_n - \dim(\Phi_2(P_n)) = n - 3.$$

We still do not know the dimension of  $\Phi_2(P_n)$  for other GBCs.

### 3.2 A reduced space

Since the dimension of the whole space  $\Phi_d(P_n)$  is high for  $d \geq 2$ , we would like to construct a subspace  $\Psi_d(P_n)$  that satisfies the same two properties (P1) and (P2): a ‘reduced’ space. Our idea for this is to expand any polynomial of degree  $\leq d$  from its polar form. Recall that any  $p \in \Pi_d$  has a unique,  $d$ -variate polar form (blossom)  $\mathcal{P}(\mathbf{x}_1, \dots, \mathbf{x}_n)$  for points  $\mathbf{x}_1, \dots, \mathbf{x}_d \in \mathbb{R}^2$  [20], [19], uniquely defined by the three properties: (i) it is symmetric in the variables  $\mathbf{x}_1, \dots, \mathbf{x}_d$ ; (ii) it is multi-affine, i.e., affine in each variable while the others are fixed; and (iii) it has the diagonal property,

$$\mathcal{P}(\mathbf{x}^{[d]}) := \mathcal{P}(\underbrace{\mathbf{x}, \dots, \mathbf{x}}_d) = p(\mathbf{x}). \quad (17)$$

Given a polynomial  $p \in \Pi_d$ , we use (17) and expand one of the  $\mathbf{x}$  variables using the GBCs  $\phi_i$  and the remaining  $d - 1$  variables using the local barycentric coordinates,  $\lambda_{i,j}$ ,  $j = -1, 0, 1$ , defined by

$$\mathbf{x} = \sum_{j=-1}^1 \lambda_{i,j}(\mathbf{x}) \mathbf{v}_{i+j}, \quad 1 = \sum_{j=-1}^1 \lambda_{i,j}(\mathbf{x}). \quad (18)$$

Consider first the quadratic case.

**Theorem 2.** *Let*

$$F_i = \phi_i \lambda_{i,0}, \quad F_{i,1} = \phi_i \lambda_{i,1} + \phi_{i+1} \lambda_{i+1,-1}, \quad i = 1, \dots, n,$$

and

$$\Psi_2(P_n) = \text{span}\{F_i, F_{i,1}, i = 1, \dots, n\}. \quad (19)$$

Then  $\Pi_2 \subset \Psi_2(P_n) \subset \Phi_2(P_n)$ .

*Proof.* For any  $p \in \Pi_2$ , expanding one of the two  $\mathbf{x}$  variables in (17) using the GBCs  $\phi_i$  gives

$$p(\mathbf{x}) = \mathcal{P}(\mathbf{x}, \mathbf{x}) = \sum_{i=1}^n \phi_i(\mathbf{x}) \mathcal{P}(\mathbf{v}_i, \mathbf{x}). \quad (20)$$

Then expanding the remaining  $\mathbf{x}$  variable using the coordinates  $\lambda_{i,j}$  gives

$$p(\mathbf{x}) = \sum_{i=1}^n \phi_i(\mathbf{x}) \sum_{j=-1}^1 \lambda_{i,j}(\mathbf{x}) \mathcal{P}(\mathbf{v}_i, \mathbf{v}_{i+j}). \quad (21)$$

Considering the two cases  $i = j$  and  $i \neq j$ , and since  $\mathcal{P}(\mathbf{v}_i, \mathbf{v}_i) = p(\mathbf{v}_i)$ , we thus obtain

$$p(\mathbf{x}) = \sum_{i=1}^n p(\mathbf{v}_i) F_i(\mathbf{x}) + \sum_{i=1}^n \mathcal{P}(\mathbf{v}_i, \mathbf{v}_{i+1}) F_{i,1}(\mathbf{x}),$$

with the  $F_i$  and  $F_{i,1}$  as in (19). This shows that  $\Pi_2 \subset \Psi_2(P_n)$ . Since each  $\lambda_{i,j}$  can be expressed as a linear combination of the  $\phi_k$ ,  $k = 1, \dots, n$ , it also follows that  $F_i, F_{i,1} \in \Phi_2(P_n)$ .  $\square$

On the edge  $[\mathbf{v}_i, \mathbf{v}_{i+1}]$ , the only  $F$ -functions that are non-zero are  $F_i$ ,  $F_{i,1}$ , and  $F_{i+1}$ , and if

$$\mathbf{x} = (1 - \mu) \mathbf{v}_i + \mu \mathbf{v}_{i+1}, \quad 0 \leq \mu \leq 1, \quad (22)$$

then

$$F_i(\mathbf{x}) = (1 - \mu)^2, \quad F_{i,1}(\mathbf{x}) = 2\mu(1 - \mu), \quad F_{i+1}(\mathbf{x}) = \mu^2, \quad (23)$$

Thus the  $2n$  functions  $F_i$  and  $F_{i,1}$  are linearly independent.

In constructing a reduced space for order  $d \geq 3$  the polar form expansion involves some functions that are zero on the boundary of  $P$ , and some of these could be linearly dependent. However, by specializing the coordinates  $\phi_i$  to be Wachspress coordinates, we are able to make a precise statement about these dependencies and ascertain the function space and its dimension.

**Theorem 3.** For  $d \geq 3$  and with Wachspress coordinates  $\phi_i$ , let

$$F_i = \phi_i \lambda_{i,0}^{d-1}, \quad i = 1, \dots, n,$$

$$F_{i,k} = \binom{d-1}{k} \phi_i \lambda_{i,1}^k \lambda_{i,0}^{d-1-k} + \binom{d-1}{k-1} \phi_{i+1} \lambda_{i+1,0}^{k-1} \lambda_{i+1,-1}^{d-k}, \quad i = 1, \dots, n, \quad k = 1, \dots, d-1,$$

and

$$\Psi_d(P_n) := \text{span}\{F_i\} \oplus \text{span}\{F_{i,k}\} \oplus \frac{b}{W} \Pi_{d-3}, \quad (24)$$

with  $W$  as in (7) and  $b$  in (12). Then  $\Pi_d \subset \Psi_d(P_n) \subset \Phi_d(P_n)$ .

*Proof.* For any  $p \in \Pi_d$  with  $d \geq 3$ . expanding one of the  $\mathbf{x}$  variables in (17) using any GBCs  $\phi_i$  gives

$$p(\mathbf{x}) = \sum_{i=1}^n \phi_i(\mathbf{x}) q_i(\mathbf{x}), \quad (25)$$

where

$$q_i(\mathbf{x}) = \mathcal{P}(\mathbf{v}_i, \mathbf{x}^{[d-1]}).$$

Then expanding the remaining  $\mathbf{x}$ 's using the  $\lambda_{ij}$  gives

$$q_i(\mathbf{x}) = \sum_{j_1, \dots, j_{d-1} = -1}^1 \lambda_{i,j_1}(\mathbf{x}) \cdots \lambda_{i,j_{d-1}}(\mathbf{x}) \mathcal{P}(\mathbf{v}_i, \mathbf{v}_{i+j_1}, \dots, \mathbf{v}_{i+j_{d-1}}),$$

and so  $q_i$  is the Bernstein-Bézier polynomial

$$q_i(\mathbf{x}) = \sum_{|\mathbf{j}|=d-1} \mathcal{P}(\mathbf{v}_{i-1}^{[j_1]}, \mathbf{v}_i^{[j_2+1]}, \mathbf{v}_{i+1}^{[j_3]}) B_{\mathbf{j}}^{d-1}(\mathbf{x}),$$

where

$$B_{\mathbf{j}}^{d-1} = \frac{(d-1)!}{j_1! j_2! j_3!} \lambda_{i,-1}^{j_1} \lambda_{i,0}^{j_2} \lambda_{i,1}^{j_3}.$$

By grouping together the cases  $\mathbf{j} = (0, d-1, 0)$ ,  $\mathbf{j} = (k, d-1-k, 0)$  and  $\mathbf{j} = (0, d-1-k, k)$ ,  $k = 1, \dots, d-1$ , we obtain

$$\begin{aligned} q_i(\mathbf{x}) &= \mathcal{P}(\mathbf{v}_i^{[d]}) \lambda_{i,0}^{d-1}(\mathbf{x}) \\ &+ \sum_{j=-1,1}^{d-1} \sum_{k=1}^{d-1} \mathcal{P}(\mathbf{v}_i^{[d-k]}, \mathbf{v}_{i+j}^{[k]}) \binom{d-1}{k} \lambda_{i,0}^{d-1-k}(\mathbf{x}) \lambda_{i,j}^k(\mathbf{x}) \\ &+ \lambda_{i,-1}(\mathbf{x}) \lambda_{i,1}(\mathbf{x}) r_i(\mathbf{x}) \end{aligned} \quad (26)$$

for some polynomial  $r_i \in \Pi_{d-3}$ .

Now, for Wachspress coordinates  $\phi_i$ , the polynomial weight function  $w_i$  in (6) contains the factors  $h_j$ ,  $j \neq i-1, i$ , and  $\lambda_{i,-1}$  contains the factor  $h_i$  and  $\lambda_{i,+1}$  contains the factor  $h_{i-1}$ , and so there is some constant  $K_i$  such that

$$\phi_i(\mathbf{x}) \lambda_{i,-1}(\mathbf{x}) \lambda_{i,1}(\mathbf{x}) = K_i \frac{b(\mathbf{x})}{W(\mathbf{x})},$$

where  $b$  is the bubble function in (12), and  $W$  as in (7). Therefore, substituting (26) into (25) gives

$$p = \sum_{i=1}^n p(\mathbf{v}_i) F_i + \sum_{i=1}^n \sum_{k=1}^{d-1} \mathcal{P}(\mathbf{v}_i^{[d-k]}, \mathbf{v}_{i+1}^{[k]}) F_{i,k} + \frac{b}{W} r, \quad (27)$$

with the  $F_i$  and  $F_{i,k}$  as in (24) and  $r$  some polynomial in  $\Pi_{d-3}$ . This shows that  $\Pi_d \subset \Psi_d$ . The sum in (24) is direct due to  $b$  being zero on the edges of  $P_n$  and the  $F_{i,k}$  being zero at the vertices of  $P_n$ .  $\square$

In analogy to the quadratic case, if  $\mathbf{x}$  is the boundary point in (22), then

$$F_i(\mathbf{x}) = b_0^d(\mu), \quad F_{i,k}(\mathbf{x}) = b_k^d(\mu), \quad k = 1, \dots, d-1, \quad F_{i+1}(\mathbf{x}) = b_d^d(\mu),$$

where

$$b_k^d(\mu) = \binom{d}{k} \mu^k (1-\mu)^{d-k}, \quad k = 0, 1, \dots, d, \quad (28)$$

the univariate Bernstein polynomials of degree  $d$  in  $\mu$ , and all remaining  $F_i$ 's are zero on  $e_i$ .

By the linear independence of the  $b_k^d$ , the  $n$  functions  $F_i$  and the  $(d-1)n$  functions  $F_{i,k}$  are linearly independent. By the dimension of  $\Pi_{d-3}$ , we have

$$\dim \Psi_d(P_n) = dn + (d-1)(d-2)/2.$$

**Remark 1.** We note that this dimension is less than that of the virtual element space of [5] and [6] which is

$$dn + d(d-1)/2.$$

See, e.g. (4.3) in [6].

## 4 Polygonal spline spaces

Recall that Bernstein-Bézier polynomials on triangles can be pieced together (continuously and even smoothly) over a collection of triangles to form a bivariate spline function (cf. e.g. [8, 14]). Similarly, polygonal Bernstein-Bézier functions can be pieced together continuously over a collection of convex polygons to form a bivariate polygonal spline function.

Let  $\Delta$  be a collection of convex polygons and let  $\Omega = \bigcup_{P \in \Delta} P$  be the domain consisting of these polygons in  $\Delta$ . We assume that the interiors of any two polygons  $P$  and  $\tilde{P}$  do not intersect and the intersection of  $P$  and  $\tilde{P}$  is either their common edge or common vertex if the intersection is not empty. Then let

$$\mathcal{S}_d(\Delta) = \{s \in C(\Omega), s|_{P_n} \in \Psi_d(P_n), \forall P_n \in \Delta\}, \quad (29)$$

where, for general GBCs,

$$\Psi_1(P_n) := \Phi_1(P_n) = \text{span}\{\phi_i, i = 1, \dots, n\},$$

and  $\Psi_2(P_n)$  is defined by (19), and for Wachspress coordinates and  $d \geq 3$ ,  $\Psi_d(P_n)$  is defined in (24).

### 4.1 Space dimension

By the results of the previous section, we have

**Theorem 4.** For any GBCs,

$$\dim(\mathcal{S}_1(\Delta)) = \#(V), \quad \dim(\mathcal{S}_2(\Delta)) = \#(V) + \#(E),$$

and for Wachspress coordinates and  $d \geq 3$ ,

$$\dim(\mathcal{S}_d(\Delta)) = \#(V) + (d-1)\#(E) + \frac{(d-1)(d-2)}{2}\#(\Delta), \quad (30)$$

where  $V$  and  $E$  are the sets of vertices and edges of  $\Delta$ .

*Proof.* We prove this only for the quadratic case,  $d = 2$ . The other cases are similar. For each vertex  $\mathbf{v} \in \Delta$ , let  $\mathcal{P}(\mathbf{v})$  be the set of polygons in  $\Delta$  sharing the vertex  $\mathbf{v}$ . For each  $P \in \mathcal{P}(\mathbf{v})$ , recall the functions  $F_i$  and  $F_{i,1}$  of (19). Since  $\mathbf{v}$  is a vertex of  $P$ , we let  $F_{\mathbf{v},P}$  be the function  $F_i$  associated with vertex  $\mathbf{v}$ . Then we define the vertex spline

$$S_{\mathbf{v}}(\mathbf{x}) = \begin{cases} F_{\mathbf{v},P}(\mathbf{x}) & \text{if } \mathbf{x} \in P \in \mathcal{P}_{\mathbf{v}} \\ 0. & \text{otherwise} \end{cases} \quad (31)$$

Its support is the union of the polygons in  $\mathcal{P}_{\mathbf{v}}$ . We now show that  $S_{\mathbf{v}}$  is continuous on  $\Omega$ . For any two polygons  $P$  and  $\tilde{P}$  in  $\mathcal{P}_{\mathbf{v}}$  which share a common edge  $[\mathbf{v}, \mathbf{u}]$ , we see from (23) that at the point  $\mathbf{x} = (1 - \mu)\mathbf{v} + \mu\mathbf{u}$ , with  $\mu \in [0, 1]$ ,

$$F_{\mathbf{v},P}(\mathbf{x}) = (1 - \mu)^2 = F_{\mathbf{v},\tilde{P}}(\mathbf{x}).$$

As all  $F_{\mathbf{v},P}$  are zero on the edges of  $P$  which do not connect to  $\mathbf{v}$ , we know that  $S_{\mathbf{v}}$  is zero on the boundary of the union of the polygons in  $\mathcal{P}_{\mathbf{v}}$ , and thus  $S_{\mathbf{v}}$  is continuous.

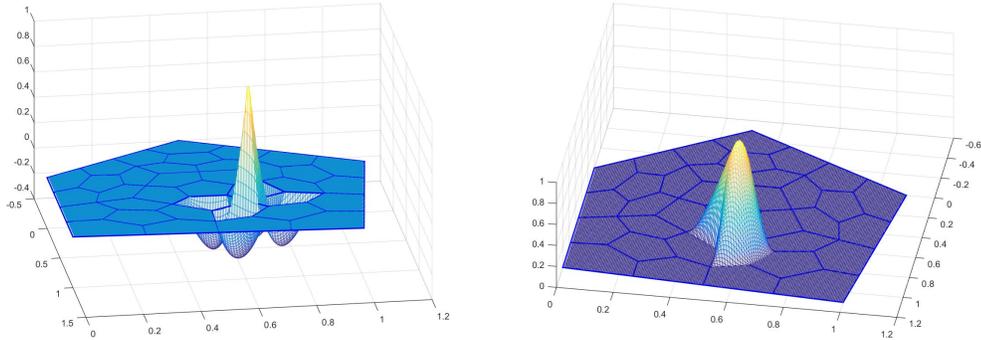


Figure 1: Vertex and edge splines

Next for each interior edge  $e = [\mathbf{v}, \mathbf{u}]$  in  $\Delta$ , there are two polygons  $P$  and  $\tilde{P}$  sharing  $e$ . Let  $F_{e,P}$  be the function  $F_{i,1}$  of (19) on  $P$  associated with  $e$ , and  $F_{e,\tilde{P}}$  the one on  $\tilde{P}$  associated with  $e$ . Then we define the edge spline

$$S_e(\mathbf{x}) = \begin{cases} F_{e,P}(\mathbf{x}), & \text{if } \mathbf{x} \in P \\ F_{e,\tilde{P}}(\mathbf{x}), & \text{if } \mathbf{x} \in \tilde{P} \\ 0, & \text{otherwise,} \end{cases} \quad (32)$$

whose support is  $P \cup \tilde{P}$ . Since  $S_e$  is zero on the boundary of  $P \cup \tilde{P}$ , and since at the point  $\mathbf{x} = (1 - \mu)\mathbf{v} + \mu\mathbf{u}$ ,  $\mu \in [0, 1]$ ,

$$F_{e,P}(\mathbf{x}) = 2\mu(1 - \mu) = F_{e,\tilde{P}}(\mathbf{x}),$$

it follows that  $S_e$  is continuous on  $\Omega$ . Fig. 1 shows examples of  $S_{\mathbf{v}}$  and  $S_e$ .

For a boundary edge  $e$ , we define  $S_e$  similarly. Since the functions  $S_{\mathbf{v}}$  and  $S_e$  form a basis for  $S_2(\Delta)$ , the dimension of  $S_2(\Delta)$  follows.  $\square$

## 4.2 Interpolation

We now explain how to use the functions in  $S_d(\Delta)$  for interpolation. Given the values  $f(\mathbf{v})$  of some function  $f$  defined at the vertices  $\mathbf{v} \in V$  of  $\Delta$  there is a unique function  $S_f \in S_1(\Delta)$  satisfying

$$S_f(\mathbf{v}) = f(\mathbf{v}), \quad \mathbf{v} \in V, \quad (33)$$

namely

$$S_f = \sum_{\mathbf{v} \in V} f(\mathbf{v}) S_{\mathbf{v}}, \quad (34)$$

where  $S_{\mathbf{v}}$  is a vertex spline in  $S_1(\Delta)$ , analogous to (31). From the definition of GBCs,  $S_f$  reproduces linear polynomials. We now extend the study to find Lagrange functions for interpolation for degree  $d \geq 2$ .

Consider  $d = 2$ . Let us start with a single polygon  $P$ . Following an observation of [22] and [21] we can transform the functions  $F_i$  and  $F_{i,1}$  into an interpolatory basis, interpolating at the vertices  $\mathbf{v}_j$  and the midpoints

$$\mathbf{v}_{j,1} := (\mathbf{v}_j + \mathbf{v}_{j+1})/2.$$

Indeed, since

$$F_i(\mathbf{v}_j) = \delta_{ij}, \quad F_i(\mathbf{v}_{i-1,1}) = F_i(\mathbf{v}_{i,1}) = 1/4, \quad F_i(\mathbf{v}_{j,1}) = 0, \quad j \neq i-1, i,$$

and

$$F_{i,1}(\mathbf{v}_j) = 0, \quad F_{i,1}(\mathbf{v}_{j,1}) = \delta_{ij}/2,$$

we obtain

**Theorem 5.** *Fix a polygon  $P$  in  $\Delta$ . The functions defined by*

$$\begin{aligned} L_i &= -\frac{1}{2}F_{i-1,1} + F_i - \frac{1}{2}F_{i,1}, \quad i = 1, \dots, n, \\ L_{i,1} &= 2F_{i,1}, \quad i = 1, \dots, n, \end{aligned}$$

are an interpolatory basis for  $S_2(P)$ , i.e., for any continuous function  $f$  over  $P$ , define  $s_f \in S_2(P)$  by

$$s_f(\mathbf{x}) = \sum_{i=1}^n f(\mathbf{v}_i) L_i(\mathbf{x}) + \sum_{i=1}^n f(\mathbf{v}_{i,1}) L_{i,1}(\mathbf{x}), \quad \mathbf{x} \in P. \quad (35)$$

Then  $s_f = f$  for all  $f \in S_2(P)$ .

We are now ready to consider  $d \geq 3$  with Wachspress coordinates. Let  $\ell_i^d$ ,  $i = 0, 1, \dots, d$  be the univariate Lagrange fundamental polynomials of degree  $\leq d$  w.r.t. the points  $j/d$ ,  $j = 0, 1, \dots, d$ , satisfying  $\ell_i^d(j/d) = \delta_{ij}$ . Then the inner polynomials in (28) can be expressed as

$$b_i^d = \sum_{j=1}^{d-1} b_i^d(j/d) \ell_j^d, \quad i = 1, \dots, d-1.$$

Then if  $A = [a_{ij}]$  is the inverse of the matrix  $[b_i^d(j/d)]_{i,j=1,\dots,d}$ , then we know that

$$\ell_i^d = \sum_{j=1}^{d-1} a_{ij} b_j^d, \quad i = 1, \dots, d-1.$$

It follows that the functions

$$\begin{aligned} L_{i,k} &= \sum_{r=1}^{d-1} a_{kr} F_{i,r}, \quad i = 1, \dots, n, \quad k = 1, \dots, d-1, \\ L_i &= F_i - \sum_{k=1}^{d-1} (1-k/d)^d (L_{i,k} + L_{i-1,d-k}), \quad i = 1, \dots, n, \end{aligned}$$

are an interpolatory basis for the ‘boundary’ part

$$S_{d,B}(P) := \text{span}\{F_i, i = 1, \dots, n\} \oplus \text{span}\{F_{i,k}, i = 1, \dots, n, k = 1, \dots, d-1\},$$

of  $S_d(P)$ , with respect to the side nodes,

$$\mathbf{v}_{i,k} := \frac{(d-k)\mathbf{v}_i + k\mathbf{v}_{i+1}}{d}, \quad k = 1, \dots, d-1.$$

In other words

$$I_B f(\mathbf{x}) := \sum_{i=1}^n f(\mathbf{v}_i) L_i(\mathbf{x}) + \sum_{i=1}^n \sum_{k=1}^{d-1} f(\mathbf{v}_{i,k}) L_{i,k}(\mathbf{x}), \quad \mathbf{x} \in P,$$

is the unique interpolant to  $f$  in  $S_{d,B}$ . For example, when  $d = 3$ ,

$$\begin{bmatrix} b_1^3 \\ b_2^3 \end{bmatrix} = \frac{2}{9} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} l_1^3 \\ l_2^3 \end{bmatrix},$$

and then

$$\begin{bmatrix} l_1^3 \\ l_2^3 \end{bmatrix} = \frac{3}{2} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} b_1^3 \\ b_2^3 \end{bmatrix},$$

and so the interpolatory basis is

$$\begin{aligned} L_{i,1} &= \frac{3}{2}(2F_{i,1} - F_{i,2}), & L_{i,2} &= \frac{3}{2}(-F_{i,1} + 2F_{i,2}), & i &= 1, \dots, n, \\ L_i &= F_i - \frac{8}{27}(L_{i,1} + L_{i-1,2}) - \frac{1}{27}(L_{i,2} + L_{i-1,1}), & i &= 1, \dots, n. \end{aligned}$$

Interpolation in the whole space  $S_d(P)$  can now be carried out in two steps: by first interpolating  $f$  using  $S_{d,B}(P)$  and then by interpolating the error on a set of unisolvent points in the interior of  $P$  and combining the two terms. Let  $\xi_{\mathbf{j}} \in P$ ,  $\mathbf{j} \in \mathbb{N}_0^3$ ,  $|\mathbf{j}| = d-3$  be a set of distinct points which is unisolvent for polynomial interpolation in  $\Pi_{d-3}$ . Let  $\tilde{L}_{\mathbf{j}} \in \Pi_{d-3}$ ,  $|\mathbf{j}| = d-3$ , be the Lagrange fundamental polynomials for these points, i.e.,  $\tilde{L}_{\mathbf{j}}(\xi_{\mathbf{k}}) = \delta_{\mathbf{j},\mathbf{k}}$ . Then

$$S_f := I_B f + \frac{b}{W} \sum_{|\mathbf{j}|=d-3} \frac{W(\xi_{\mathbf{j}})}{b(\xi_{\mathbf{j}})} (f(\xi_{\mathbf{j}}) - I_B f(\xi_{\mathbf{j}})) \tilde{L}_{\mathbf{j}}, \quad (36)$$

interpolates  $f$  in  $S_d(P)$ .

For  $d = 3$  we could choose any single point  $\xi_{000}$  in  $P$ . For  $d \geq 4$  we could choose a triangular grid of points

$$\xi_{\mathbf{j}} := \frac{j_1 \mathbf{p}_1 + j_2 \mathbf{p}_2 + j_3 \mathbf{p}_3}{d-3}, \quad |\mathbf{j}| = d-3,$$

where  $T = [\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3]$  is a triangle inside  $P$ .

Figure 2 shows possible configurations of boundary and interior points for interpolation.

This explains how to use polygonal Bernstein-Bézier functions to do interpolation on a single polygon. Clearly, we can extend the interpolation scheme to a polygonal partition  $\Delta$  so that we can use the spline functions in  $S_d(\Delta)$  to interpolate any continuous function  $f$ .

**Remark 2.** *Certainly, it is interesting to study a smooth polygonal spline space, e.g.,*

$$S_d^r(\Delta) = \{s \in C^r(\Omega), s|_{P_n} \in \Psi_d(P_n), \forall P_n \in \Delta\}. \quad (37)$$

*This is one of our future research topics.*

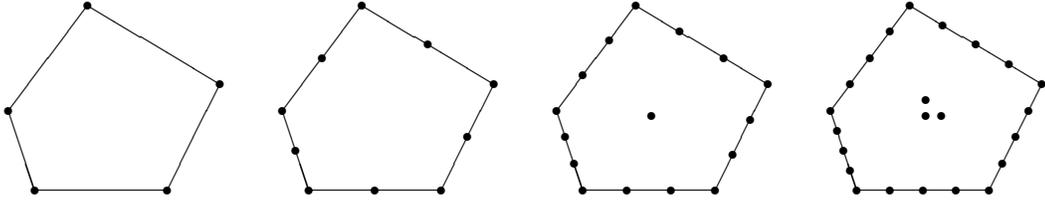


Figure 2: Linear, quadratic, cubic, and quartic elements

## 5 Numerical Solution of Poisson Equation

Let us use the quadratic and cubic polygonal finite elements constructed in the previous section to solve the following Poisson equation with Dirichlet boundary value problem:

$$\begin{cases} -\Delta u = f, & x \in \Omega \subset \mathbb{R}^2 \\ u = g, & x \in \partial\Omega, \end{cases} \quad (38)$$

where  $\Omega$  is a polygonal domain and  $\Delta$  is the standard Laplace operator. Our main point is to provide a proof of concept that these finite elements can be used to solve partial differential equations numerically just like any standard finite elements, macro finite elements, mortar elements, virtual finite elements, multivariate splines, and polygonal finite element of order 1, etc.. See, e.g. in [7], [4], [3], [2], [5], [6], [18] and etc..

For simplicity, let us illustrate our approach with some simple partitions. For example, we shall use  $\mathbf{\Delta}$  which consists of all convex quadrilaterals. How to obtain a quadrangulation of a given domain  $\Omega$  can be seen, e.g. in [14]. For another example,  $\mathbf{\Delta}$  can be a collection of all pentagons. We shall explain a refinement scheme to subdivide pentagons later in this section. Finally, we conclude this paper with an example over a partition of mixed polygons.

Let  $S_2(\mathbf{\Delta})$  be the collection of polygonal spline functions of degree 2 over  $\mathbf{\Delta}$ . Let  $B(u, v) = \int_{\Omega} \nabla u \cdot \nabla v dx$  be the standard bilinear form associated with Poisson equation. Our numerical method is to solve the following weak solution:

$$B(u_h, v) = \langle f, v \rangle, \forall v \in S_d(\mathbf{\Delta}) \cap H_0^1(\Omega). \quad (39)$$

Using the standard argument in the finite element method (see the proof of the Céa Lemma), we have

**Lemma 1.** *Let  $u_h$  be approximate weak solution satisfying (39) in  $V_h = S_d(\mathbf{\Delta}) \cap H_0^1(\Omega)$ . Then*

$$\|u - u_h\|_{H_0^1(\Omega)} \leq \frac{1}{m} \min_{v \in V_h} \|u - v\|_{H_0^1(\Omega)},$$

where  $m > 0$  is a constant dependent on the domain  $\Omega$ .

As  $S_d(\mathbf{\Delta})$  is a space containing all piecewise polynomials of degree  $d$  over  $\mathbf{\Delta}$ . It is known, i.e. by Bramble-Hilbert Lemma that if  $u \in H^{d+1}(\Omega)$ ,

$$\|u - S_u\|_{L_2(\Omega)} \leq C_1 h^{d+1} \text{ and } \|\nabla(u - S_u)\|_{L_2(\Omega)} \leq C_2 h^d \quad (40)$$

for some positive constants  $C_1$  and  $C_2$  which is dependent on the chunkiness of the underlying polygons  $\mathbf{\Delta}$ (cf. [3], [14]). Thus, one has

**Theorem 6.** (The serendipity finite element approximation of the weak solution) Suppose that the weak solution  $u \in H^{d+1}(\Omega)$ . Then

$$\|u - u_h\|_{H_0^1(\Omega)} \leq \frac{C_2}{m} h^d,$$

where  $C_2$  is a positive constant dependent on the chunkiness of the underlying polygons  $\Delta$ .

We have implemented these polygonal spline functions of order  $d$  with  $d = 2, d = 3$  over arbitrary partition  $\Delta$  in MATLAB for numerical solution of Poisson equations. Our implementation follows the approach discussed in [2] on how to use multivariate splines for numerical solutions of many other linear and nonlinear PDE. See similar approaches in [16] and [13].

Let us explain our implementation more detail. For convenience, let us focus on  $d = 2$ . Mainly, we use a vector  $\mathbf{c}$  of coefficients ( $2n$  per polygon with  $n$  sides) over the list of polygons in  $\Delta$  to represent a polygonal spline function in  $S_2(\Delta)$ . Note that this is different from traditional finite element method. The vector  $\mathbf{c}$  is redundant. An advantage of such vector  $\mathbf{c}$  is that it allows us to compute the mass and stiffness matrix in parallel. When two polygons share a common edge  $e$ , there are 3 pairs of the polygonal finite element functions supported on these two polygons and each pair have the same value on the common edge  $e$ . For each pair, we consider that they are belong to the same locally supported polygonal spline function and thus, the coefficients associated with each pair should be same. In this way we build a continuity condition matrix  $H$  such that  $H\mathbf{c} = 0$  represents all these the continuity conditions among the coefficients in  $\mathbf{c}$ . We also form mass and stiffness matrices by using tensor product of Gauss quadrature formula with order 5 or larger. Note that our mass and stiffness matrices are blockly diagonal, e.g. the mass matrix  $M = [M_p]_{p \in \Delta}$  with  $M_p = [M_{p,i,j}]_{i,j=1,\dots,2n}$  with  $n = n(p)$  being the number of sides of  $p$ , where

$$M_{p,i,j} = \int_p L_i L_j dx dy$$

with  $L_i = L_{i,p}$  being polygonal finite elements over  $p$ . When  $n = 4$ , we simply use tensor product of Gauss quadrature formula with order 5. When  $n > 4$ , we divide  $p$  into a few quadrilaterals and apply for tensor product of Gauss quadrature formula to each quadrilateral and add the numerical values together. Similar for the stiffness matrix  $S$ . We use the interpolatory scheme discussed in a previous section to find polygonal interpolant  $I_f$  and then use the mass matrix to generate an approximation to the right-hand side  $\langle f, L_{i,p} \rangle$  in (39) for  $v = L_{i,p}$ . That is, if  $I_f = \sum_{p \in \Delta} \sum_{i=1}^{n(p)} f(\mathbf{v}_i(p)) L_{2i-1} + f((\mathbf{v}_i(p) + \mathbf{v}_{i+1}(p))/2) L_{2i}$ , we compute

$$\langle f, L_{i,p} \rangle = \sum_{j=1}^{n(p)} f(\mathbf{v}_j(p)) \int_p L_{2j-1} L_{i,p} dx dy + f((\mathbf{v}_i(p) + \mathbf{v}_{i+1}(p))/2) \int_p L_{2j} L_{i,p} dx dy,$$

where the two integrals are already calculated in the mass matrix. These are entries for  $\mathbf{F}$ . Boundary conditions are also generated by using interpolation scheme. We can write  $B\mathbf{c} = G$ . Finally we solve the minimization

$$\min_{\mathbf{c}} \frac{1}{2} \mathbf{c}^\top S \mathbf{c} - \mathbf{F} M \mathbf{c}, H \mathbf{c} = 0, B \mathbf{c} = G, \quad (41)$$

where  $H\mathbf{c} = 0$  represents the all continuity conditions and  $B\mathbf{c} = G$  all the boundary conditions,  $M$  and  $S$  are mass and stiffness matrices. This minimization problem can be solved using the iterative method discussed in [2] in a few iterative step. In our computation, we only did 5 iterations. The rest is to evaluate the polygonal spline solution at 1,000,000 points equally spaced points over the minimal rectangular containing  $\Omega$  and compare the numerical solution with exact solution to approximate the  $L_2$  norm over  $\Omega$  using the rooted mean squared error. Although this is not the most efficient way to implement, it provides a fast enough computation for generating our numerical results in this paper.

## 5.1 A Refinement Scheme of Pentagonal Partitions

Refinements of a triangulation and a quadriangulation are well-known. Let us first explain how to refine a pentagon partition  $\Delta$ . We start with a pentagon  $P = \langle \mathbf{v}_1, \dots, \mathbf{v}_5 \rangle$ . Let  $\mathbf{p} = (\mathbf{v}_1 + \dots + \mathbf{v}_5)/5$  be the geometric center of  $P$ . Write  $\mathbf{u}_i = (\mathbf{v}_i + \mathbf{v}_{i+1})/2$  for  $i = 1, \dots, 5$  with  $\mathbf{v}_6 := \mathbf{v}_1$  and  $\mathbf{w}_i = (\mathbf{u}_i + \mathbf{p})/2$ ,  $i = 1, \dots, 5$  as shown in Fig. 3. Connect these  $\mathbf{v}_i, \mathbf{u}_i, \mathbf{w}_i, i = 1, \dots, 5$  as shown to form a uniform refinement of the pentagon  $P$ .

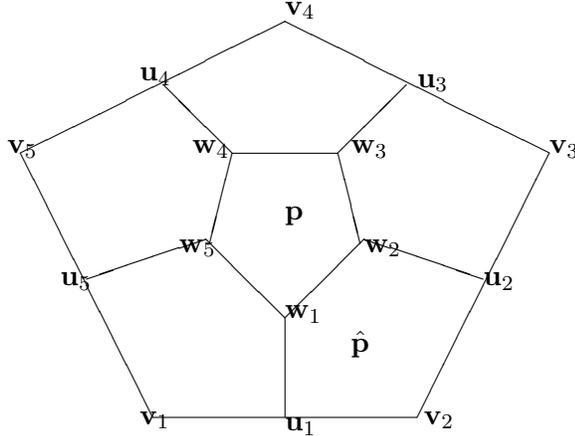


Figure 3: An illustration of the pentagon refinement with vertices  $\mathbf{v}_i, \mathbf{u}_i, \mathbf{w}_i, i = 1, \dots, 5$

For a collection  $\Delta$  of pentagons, we apply the above refinement scheme to each pentagon from  $\Delta$ . Let us now define the size  $|\Delta|$  of  $\Delta$  to be the longest of the lengths of edges in  $\Delta$ . An elementary proof shows

**Lemma 2.** *Let  $\Delta_1$  be the refinement of pentagon partition  $\Delta$ . Then*

$$|\Delta_1| \leq \frac{1}{2} |\Delta|. \quad (42)$$

If we let  $d(P)$  to be the smallest of the diameter of the circle containing  $P$ , then we know that  $|P| \rightarrow 0$  is equivalent to  $d(P) \rightarrow 0$ . Another elementary result is the following

**Lemma 3.** *Let  $\Omega$  be a convex polygonal region with  $n$  corner points. Then  $\Omega$  can be divided into a collection of convex pentagons if  $n \geq 5$ .*

So far we still do not know if any polygon  $\Omega$  can be partitioned into a collection of strictly convex pentagons although it can be partitioned into a collection of non-strictly convex pentagons. When a pentagon  $P$  is not convex, we do not know how to refine it. We leave these two problems to the interested reader.

## 5.2 Numerical Results of Polygonal Finite Elements

We now use our MATLAB codes to present some numerical results of Poisson equations using our polygonal finite elements. We have tested that our codes for  $d = 2$  and  $d = 3$  can reproduce all polynomials of degree 2 and degree 3, respectively based on Wachspress coordinates. We shall use the

No. of pentagons	DoFs	$\ u_1 - u_{1,h}\ _2$	rates	$\ u_2 - u_{2,h}\ _2$	rates	$\ u_3 - u_{3,h}\ _2$	rates
6	35	1.0947e-03	0	1.0524e-02	0	1.0979e-01	0
36	165	8.6966e-05	3.65	8.9303e-04	3.55	6.7861e-03	4.01
216	905	7.4922e-06	3.53	7.5053e-05	3.57	5.6689e-04	3.58
1296	5265	6.4614e-07	3.53	6.3632e-06	3.56	5.0677e-05	3.48
77760	31265	5.9470e-08	3.44	5.5677e-07	3.51	4.4986e-06	3.49

Table 1: Convergence of  $C^0$  quadratic serendipity finite elements over repeatedly refined pentagon partitions in Fig. 4

Root Mean Square Error (RMSE) over 1,000,000 equally spaced points over the minimal rectangle containing  $\Omega$  to show its performance. Any value located inside the rectangle and outside of  $\Omega$  will be nan (not a number) in MATLAB.

**Example 4.** We have done several tests on smooth solutions:  $u_1(x, y) = 1/(1 + x + y)$ ,  $u_2(x, y) = x^4 + y^4$  and  $u_3(x, y) = \sin(\pi(x^2 + y^2)) + 1$  to check the performance of our polygonal finite elements of order 2 based on partitions obtained by repeated pentagonal refinements as in Fig. 4.

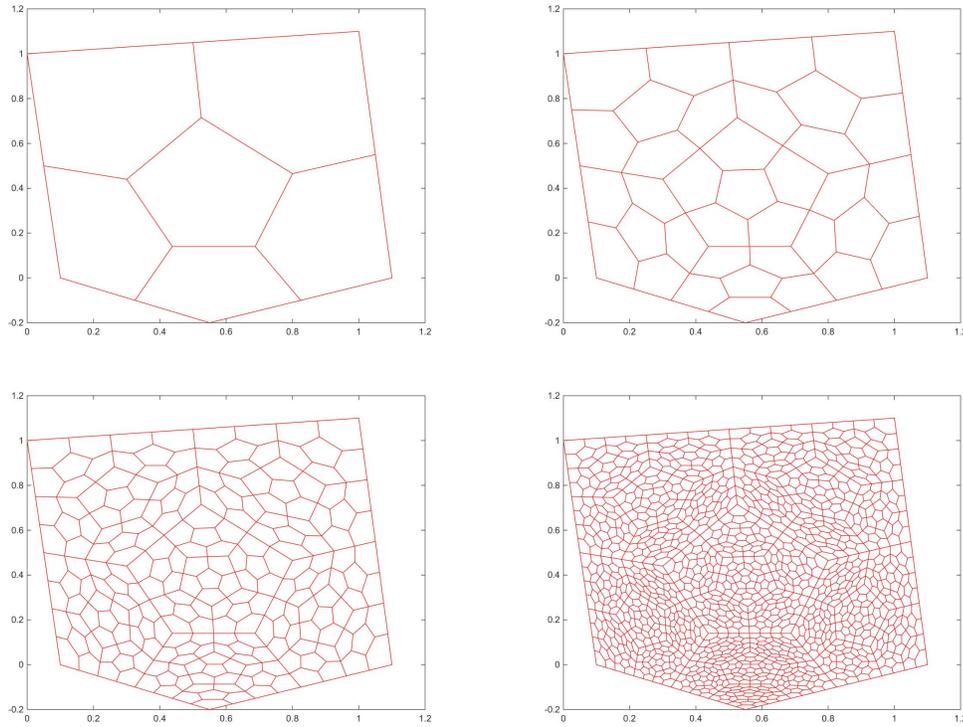


Figure 4: A Pentagon Partition (top-left) and its refinements

Numerical results are shown in Table 1, where the rate of convergence is measured by using  $\log_2(\text{err}_\ell/\text{err}_{\ell-1})$  for  $\ell \geq 2$ , where  $\text{err}_\ell = \|u - u_\ell\|_2$  stands for the RMSE error of numerical solution  $u_\ell$  at refinement level  $\ell$  against the exact solution  $u$ . Indeed, as  $\text{err}_\ell = O(1/2^\ell)^\alpha$  for some  $\alpha > 0$ , one is interested in the rate of  $\alpha$ .

These numerical results in Table 1 show that the quadratic finite elements over pentagon partitions work very well with  $\alpha \approx 3.5$  even though we are not able to prove the boundedness of the chunkiness

of refined pentagon partitions.

**Example 5.** To see that the refinement of pentagonal partitions indeed works better, we present a comparison of numerical solution of quadratic finite elements over triangulation, quadrangulation and pentagonal partitions as shown in Fig. 5.

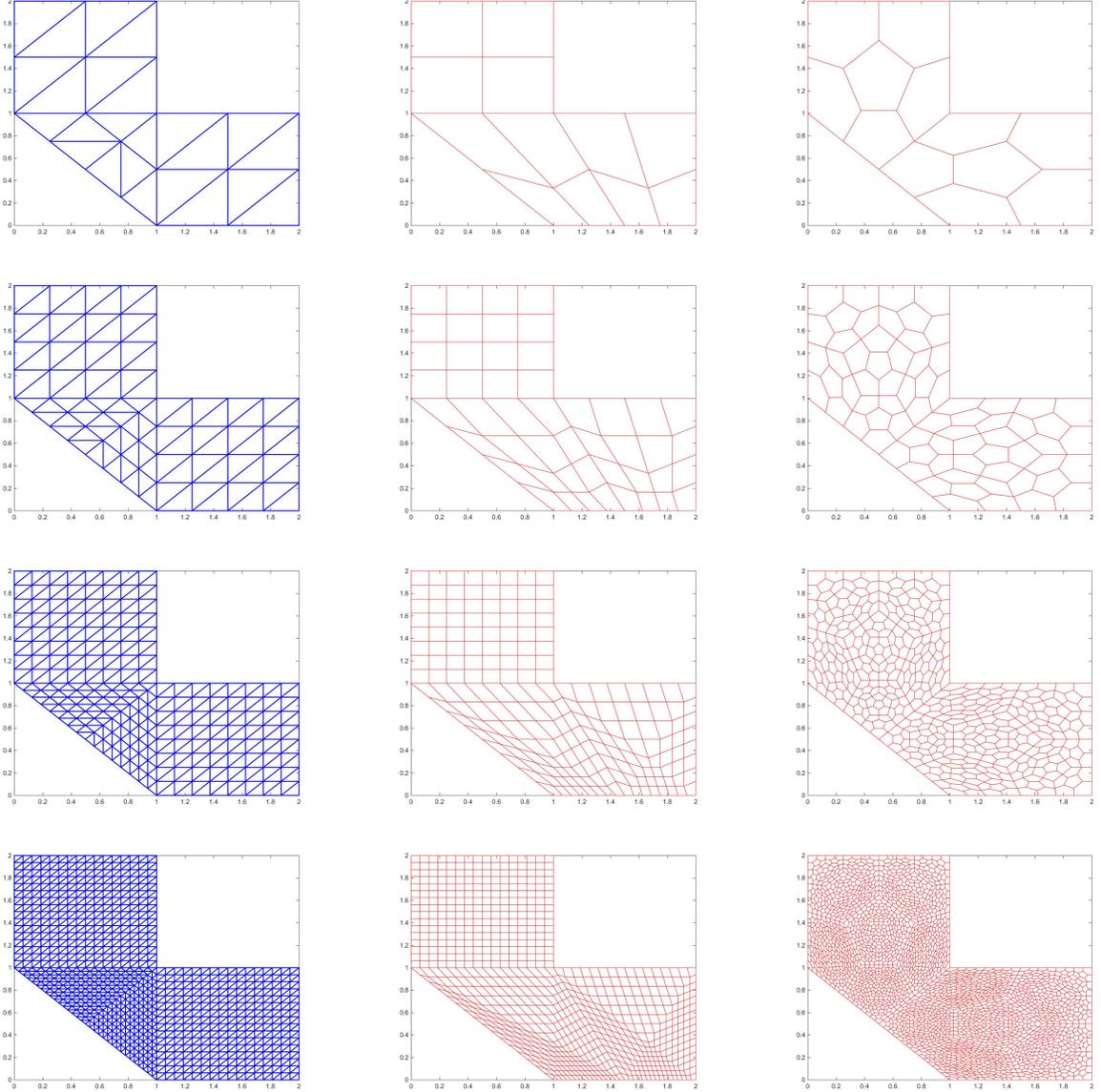


Figure 5: Three Different Partitions of an  $L$ -domain  $\Omega$

See numerical results shown in Table 2, where the rate of convergence is measured by using  $\log_2(\text{err}_l/\text{err}_{l-1})$  for  $l \geq 2$  as in Example 4.

It is clear from Table 2 that the quadratic finite element method using quadrangulations is more efficient than the triangular finite element method. Also, the quadratic polygonal element method based on pentagon partitions is more efficient than the polygonal finite elements based on quadrilaterals and triangular finite elements.

Table 2: Numerical Comparison of Quadratic Finite Elements over Triangulations, Quadrangulations and Pentagon Partitions. In the following, T's stands for the number of triangles, Q's for number of quadrilaterals, and P's for number of pentagons.

T's	$\ u_3 - u_{3,T}\ _2$	rate	Q's	$\ u_3 - u_{3,Q}\ _2$	rate	P's	$\ u_3 - u_{3,P}\ _2$	rate
24	1.9195	0	12	5.8332e-01	0	12	8.1060e-01	0
96	5.1101e-01	2.84	48	6.3392e-02	3.20	72	4.0100e-02	4.33
384	7.1105e-02	3.00	192	7.1610e-03	3.14	432	3.1625e-03	3.66
1536	9.6360e-03	3.10	768	8.2142e-04	3.12	2592	2.6070e-04	3.60
6144	1.4255e-03	3.03	3072	9.9943e-05	3.04	15552	2.149e-05	3.60

We have also done the same test for using the mean value coordinates and the convergence rate is  $\alpha = 3.5$  for the pentagonal partitions. Furthermore, let us use the polygonal finite element of order 3 to show that numerical solution based on repeatedly refined pentagonal partitions also works very well.

**Example 6.** We repeat the same experiment as in Example 5 based on the polygonal finite element of order 3 over the same three partitions as in Fig. 5.

Table 3: Numerical Comparison of Cubic Finite Elements over Triangulations, Quadrangulations and Pentagon Partitions.

T's	$\ u_3 - u_{3,T}\ _2$	rate	Q's	$\ u_3 - u_{3,Q}\ _2$	rate	P's	$\ u_3 - u_{3,P}\ _2$	rate
24	3.8339e-01		12	3.3958e-01		12	4.0742e-01	
96	1.8035e-02	4.41	48	1.0199e-02	5.05	72	7.6599e-03	5.73
384	1.2795e-03	3.81	192	6.3810e-04	3.99	432	2.5641e-04	4.90
1536	8.2478e-05	3.95	768	4.3051e-05	3.89	2592	1.0628e-05	4.59
6144	5.1569e-06	3.99	3072	2.7495e-06	3.96	15552	4.9148e-07	4.43

In order to compare the efficiency of these three numerical methods let us define the effectiveness of a finite element method based on a partition  $\Delta_h$  by

$$e(\Delta_h) = \#(\Delta_h) \|u - u_{\Delta_h}\|_2, \quad (43)$$

where  $u_{\Delta_h}$  is the finite element solution of the PDE over  $\Delta_h$  of the domain  $\Omega$ ,  $\Delta_h$  is either a triangulation of size  $h$  or a quadrangulation of size  $h$  or a pentagon partition of size  $h$ , and  $\#(\Delta_h)$  stands for the number of triangles or quadrilaterals or pentagons in the partition  $\Delta_h$ . Here  $\|u - u_{\Delta_h}\|_2$  stands for the  $L_2$  norm of  $u - u_{\Delta_h}$ . We say that a finite element method over  $\Delta_h$  is more effective if the value  $e(\Delta_h)$  is smaller as a method with smaller  $e(\Delta_h)$  has either less number of polygons or more accurate or both. From Example 6 we can see that the cubic polygonal finite element over pentagonal partitions is most effective.

## 6 Conclusions and Future Research Problems

We proposed to define a continuous polygonal spline space  $S_d(\Delta)$  of order  $d \geq 1$  over a mixed partition of convex polygons with arbitrary number of sides which possesses a property of reproducing polynomials of degree  $d$ . Then we presented a construction of locally supported basis functions for  $S_d(\Delta)$  over the mixed partition  $\Delta$  of convex polygons for  $d \geq 2$ . For  $d = 2$ , our construction is simpler

than the one in [22]. For  $d \geq 3$ , our construction is only based on Wachspress coordinates. We then explained how to use these spline functions to solve the Poisson equation numerically over a partition of quadrilaterals or pentagons. In particular, we found a refinement scheme for pentagons and showed that this refinement leads to a higher order of convergence rate.

There are many problems remaining open. For example, when  $d \geq 3$ , we are only able to construct polygonal splines of order  $d$  using Wachspress coordinates. It is interesting to see if one can use other generalized barycentric coordinates to construct higher order splines. Also, we would like to know the approximation properties of our interpolation schemes. In particular, is it possible that the polygonal splines studied in this paper have higher approximation power than standard polynomial splines? Our numerical results suggest that the refinement scheme for pentagons may lead to higher accuracy than the uniform refinement of triangulations and quadrangulations. We would like to understand this phenomenon better.

In addition, we would like to extend our study to a higher dimensional setting. Currently we are working on constructing polyhedral splines over a 3D partition of mixed polyhedra and using them for solving partial differential equations.

Finally, we would like to construct polygonal splines which are  $C^1$  or smoother in the 2D and 3D settings using GBCs in [12], [11], [25], and etc.. These are all under investigation. Of course, we welcome any interested reader to join our efforts and understand polygonal splines better.

## References

- [1] D. N. Arnold and G. Awanou, The serendipity family of finite elements, *Found. Comput. Math.* 11 (2011), no. 3, 337344.
- [2] Awanou, G., Lai, M. J. and Wenston, P., *The Multivariate Spline Method for Numerical Solution of Partial Differential Equations, Wavelets and Splines*, Nashboro Press, (2006) edited by G. Chen and M. J. Lai, pp. 24–74.
- [3] Brenner, S. C. and L. R. Scott, *The mathematical theory of finite element methods*, Springer Verlag, New York, 1994.
- [4] F. Brezzi and M. Fortin, *Mixed and Hybrid Finite Element Methods*. Springer-Verlag, New York, 1991.
- [5] L. Beirao da Veiga, K. Lipnikov, and G. Manzini, Arbitrary-order nodal mimetic discretizations of elliptic problems on polygonal meshes. *SIAM Journal on Numerical Analysis*, 49(5):1737–1760, 2011.
- [6] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. D. Marini, and A. Russo. Basic principles of virtual element methods. *Math. Models Methods Appl. Sci.*, 23(1):199–214, 2013.
- [7] P. Ciarlet, *The Finite Element Method*, North Holland, 1978.
- [8] G. Farin, Triangular Bernstein-Bézier patches, *Comput. Aided Geom. Design*, 3 (1986), 83–127.
- [9] M. Floater, Mean value coordinates, *Comput. Aided Geom. Des.* 20 (2003), 19-27.
- [10] M. Floater, Generalized barycentric coordinates and applications, *Acta Numerica* (2015), to appear.

- [11] M. Floater, K. Hormann, and G. Kós, A general construction of barycentric coordinates over convex polygons, *Adv. Comput. Math.* 24 (2006), no. 1, 311-331.
- [12] M. S. Floater, G. Kos, and M. Reimers, Mean value coordinates in 3D, *Comput. Aided Geom. Design* 22 (2005), 623-631.
- [13] Hu, X., Han, D. and Lai, M. J., Bivariate Splines of Various Degrees for Numerical Solution of PDE, *SIAM Journal of Scientific Computing*, vol. 29 (2007) pp. 1338–1354.
- [14] M. J. Lai and L. L. Schumaker, *Spline Functions on Triangulations*, Cambridge University Press, 2007.
- [15] Lai, M. J. and Schumaker, L. L., Domain Decomposition Method for Scattered Data Fitting, *SIAM Journal on Numerical Analysis*, vol. 47 (2009) pp. 911–928.
- [16] M. J. Lai and Wenston, P., Bivariate Splines for Fluid Flows, *Computers and Fluids*, vol. 33 (2004) pp. 1047–1073.
- [17] C. T. Loop and T. D. DeRose, A multisided generalization of Bézier surfaces, *ACM Trans. Graph.* 8 (1989), 204–234.
- [18] G. Manzini, A. Russo, and N. Sukumar, New perspectives on polygonal and polyhedral finite element methods. *Math. Models Methods Appl. Sci.* 24 (2014), no. 8, 1665-1699.
- [19] H. Prautzsch, W. Boehm, and M. Paluszny, *Bézier and b-spline techniques*, Springer, 2002.
- [20] L. Ramshaw, Blossoms are polar forms, *Computer Aided Geometric Design*, 6(1989), 323–358.
- [21] A. Rand, A. Gillette and C. Bajaj, Interpolation error estimates for mean value coordinates over convex polygons, *Adv. Comput. Math.* 39 (2013), 327-347.
- [22] A. Rand, A. Gillette, and C. Bajaj, Quadratic serendipity finite elements on polygons using generalized barycentric coordinates, *Math. Comp.* 83 (2014), 2691–2716.
- [23] B. Szabó and I. Babuska, *Finite element analysis*, A Wiley-Interscience Publication, John Wiley & Sons Inc., New York, 1991.
- [24] E. L. Wachspress, A rational finite element basis, *Mathematics in Science and Engineering*, vol. 114, Academic Press, New York, 1975.
- [25] E. L. Wachspress, Barycentric coordinates for polytopes, *Comput. Math. Appl.* 61 (2011), 3319-3321.