Quadratic Serendipity Discontinuous Finite Element Discretization for S_N Transport on Arbitrary Polygonal Grids

Michael W. Hackemack[*,a], Jean C. Ragusa[b]

[a]*Naval Nuclear Laboratory, PO Box 1072, Schenectady, NY 12301, USA*
[b]*Department of Nuclear Engineering, Texas A&M University, College Station, TX 77843, USA*

Abstract

In this paper, we present a quadratic serendipity discontinuous Galerkin finite element (DGFEM) discretization of the S_N transport equation for arbitrary polygonal grids. The quadratic serendipity functions are constructed from products of linear Generalized Barycentric Coordinates (GBC) and are fully compatible with arbitrary polygonal grids. The piecewise linear (PWL) functions are a GBC that have been previously utilized for DGFEM transport. We employ these PWL functions, and others for comparison, as the underlying linear functions in this work. Once constructed, the quadratic serendipity functions span the $\{1, x, y, x^2, xy, y^2\}$ space of functions and grow by $2n$ on a mesh element where n is the number of the polygon's vertices. Numerical tests confirm that the functions capture exactly quadratic solutions and appropriate convergence rates are observed, including a test case involving spatial adaptive mesh refinement. Finally, the functions are analyzed for diffusive problems and retain full resolution in the thick diffusion limit.

Key words: Radiation Transport, Arbitrary Polygonal Grids, Discontinuous Finite Element, Serendipity Basis Functions

1. Introduction

The solution of the radiation transport equation is important for the fields of high energy density physics, astrophysics, medical imaging and treatment, reactor physics, and others. High-fidelity solutions are difficult to obtain because of the large phase space of the radiation distribution: three for space, two for angular direction on the unit sphere, one for energy, and one for time for unsteady problems [1]. The angular flux solution of the radiation transport equation, $\psi(\vec{r}, \vec{\Omega}, E)$, about position $\vec{r}$, about energy E, and traveling in a cone about direction $\vec{\Omega}$ is given by

---

[*]Corresponding author
 Email address: `michael.hackemack@unnpp.gov` (Michael W. Hackemack)

$$\vec{\Omega} \cdot \vec{\nabla}\psi(\vec{r}, \vec{\Omega}, E) + \sigma_t(\vec{r}, E)\psi(\vec{r}, \vec{\Omega}, E) = Q(\vec{r}, \vec{\Omega}, E)$$
$$+ \int_0^\infty dE' \int_{4\pi} d\Omega' \sigma_s(\vec{r}, \vec{\Omega}' \to \vec{\Omega}, E' \to E)\psi(\vec{r}, \vec{\Omega}', E'), \tag{1}$$

where σ_t and σ_s are the total and scattering cross sections, Q is an angularly-dependent distributed source, and we have omitted temporal dependence. Various numerical techniques have been developed to appropriately discretize Eq. (1) in space, energy, and angle.

In this work, we focus on using solution techniques developed for Eq. (1), also referred to as the first-order form of the transport equation. Namely, we restrict ourselves to the one-speed transport equation, the discrete ordinates (S_N) approximation in angle, and the discontinuous Galerkin finite element method (DGFEM) in space. The DGFEM technique is a Galerkin approach of the more general discontinuous finite element method (DFEM). The DGFEM discretization of the S_N transport equation dates back to Reed and Hill [2] as well as Lesaint and Raviart [3]. It is a common spatial discretization of the transport equation because it is amenable to efficient iterative techniques [4] and scales well on massively parallel architectures [5]. Most of the DGFEM S_N codes use first-order (linear) functions on simplical (triangles and tetrahedra) and tensor (quadrilaterals and hexahedra) grids, including DENOVO [6], PARTISN [7], and ATTILA [8]. Jaguar [9] and PDT [5] differ from other codes by solving up to first-order functions on unstructured polytope grids, where Jaguar uses the slice balance approach (SBA) and linear discontinuous (LD) discretizations and PDT uses the piecewise linear discontinuous (PWLD) discretization.

The PWLD discretization of the S_N transport equation is a DGFEM utilizing piecewise linear (PWL) basis functions [10, 11, 12]. These PWL functions are the only DGFEM transport discretization technique for arbitrary polygonal (as well as polyhedral) cells compatible with the thick diffusion limit [11, 13]. The thick diffusion limit is an important asymptotic limit of the transport equation that requires the use of spatial discretizations with certain properties to prevent "locking" to an incorrect solution [13]. Our work includes using these PWL functions, along with other linear polygonal functions.

For the most part, linear (including bilinear and trilinear) representations have been employed to solve the S_N transport equation using DGFEM discretizations (see previously listed transport codes). Finite-difference methods (such as the diamond-difference method) and DGFEM discretizations of the S_N typically perform global matrix-free operations by use of the transport sweep [5]. In the sweep operations, local linear system of equations are solved on each spatial cell, with finite-difference usually being computationally cheaper than DGFEM discretizations. This possibly hinders the use of high-order DGFEM methods for radiation transport, with a few exceptions. Maginot et. al [14, 15] and Hennart and De Valle [16, 17] have analyzed higher-order functions for 1D slab problems. In 2D, the TRIPLET code [18] used higher-order Lagrange elements on triangles and Wang and Ragusa [19, 20] used hierarchical triangular elements. Other methods besides the DFEM discretization have utilized higher-order approximations including AHOT for

nodal and characteristic transport [21, 22] and Wachspress' functions for curved mesh cell boundaries [23].

In this paper, we seek to analyze the DGFEM S_N transport equation with a higher-order quadratic serendipity spatial discretization on arbitrary polygonal grids. With advances in computer architectures, local computations are becoming cheaper than data movement in memory along with communication times between processors for parallel problems. For this reason, along with superior convergence rates, there is a growing benefit for using higher-order functions for the DGFEM transport equation. Likewise, we seek to analyze the DGFEM transport equation on arbitrary polygonal grids because of the flexibility and benefits they provide:

1. Polygonal mesh cells are now being employed in other physics communities - notably computational fluid dynamics (CFD) [24] and solid mechanics [25];

2. They can reduce cell/face counts which can reduce algorithm wallclock times depending on the solution method;

3. They can allow for transition elements between different portions of the domain (e.g., triangular elements bordering quadrilateral elements at the border of the boundary layer);

4. They can easily be split along cut planes - allowing the mesh to be partitioned into regular or irregular divisions as well as be generated by simplical meshing techniques across processor sets in parallel;

5. Hanging nodes from non-conforming meshes, like those that naturally arise from locally refined/adapted meshes using spatial adaptive mesh refinement (AMR) as seen in Figure 1, are not necessary.
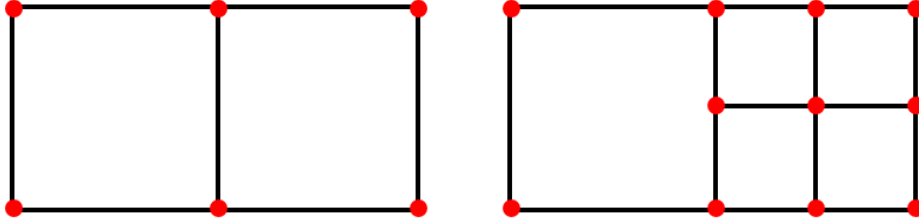

Figure 1: Local mesh refinement of an initial quadrilateral cell (left) leads to a degenerate pentagonal cell (right) without the use of a hanging node.

The higher-order discretization that we employ uses the quadratic serendipity space of functions developed by Rand et al. [26]. They proposed a process where linear functions, compatible with polygonal mesh elements, are converted into a quadratic serendipity space of functions that span $\{1, x, y, x^2, xy, y^2\}$ on strictly-convex polygons. Their procedure forms $O(n^2)$ pairwise quadratic products in 2D, where n is the number of the polygon's vertices, and subsequently eliminates interior degrees of freedom to yield $2n$ serendipity functions, hence reducing the total number of basis functions. We summarize their process in this work, and provide our methodology that is compatible with weakly-convex polygons (i.e., collinear vertices) so that we could perform AMR calculations using degenerate polygons without hanging nodes.

3

65 Finally, we note that the basis functions derived in this work are not solely compatible with the first-order

66 form of the transport equation, but can be used with the second-order form of the transport equation, such

67 as the self-adjoint and even-parity forms [27, 28].

68   The remainder of the paper is as follows. We provide a brief description of the DGFEM S_N transport

69 equation along with its theoretical spatial convergence rate in Section 2. We then review families of functions

70 that are linearly-complete on polygons in Section 3. This includes the PWL functions, along with three

71 other functions for comparative purposes. In Section 4, we present the methodology to convert the linear

72 polygonal basis functions presented in Section 3 into a serendipity space of basis functions with quadratic-

73 completeness. Section 5 briefly describes the methodology that will be employed to perform spatial AMR

74 with the DGFEM transport equation. Finally, Section 6 provides a set of numerical experiments to verify

75 the polynomial interpolation of the functions as well as their ability to retain full resolution in the thick

76 diffusion limit.

2. The DGFEM S_N Discretization of the Transport Equation

78   In this section, we review the DGFEM S_N transport equation. Given an angular quadrature set,

79 $\{w_m, \vec{\Omega}_m\}_{m=1}^{M}$, the monoenergetic S_N transport equation with isotropic scattering within an open, con-

80 vex spatial domain $\mathcal{D}$, with boundary, $\partial\mathcal{D}$, is written as

$$\left(\vec{\Omega}_m \cdot \vec{\nabla} + \sigma_t(\vec{r})\right)\psi_m(\vec{r}) = \frac{\sigma_s(\vec{r})}{4\pi}\phi(\vec{r}) + Q_m(\vec{r}), \tag{2}$$

81 where $\psi_m(\vec{r}) = \psi(\vec{r}, \vec{\Omega}_m)$ is the angular flux at position $\vec{r}$ and in direction $\vec{\Omega}_m$ and Q_m is a distributed source

82 for angle m. The scalar flux is defined as

$$\phi(\vec{r}) = \int\limits_{4\pi} d\Omega\, \psi(\vec{r}, \vec{\Omega}) \approx \sum_{m=1}^{M} w_m \psi_m(\vec{r}). \tag{3}$$

83 We then lay down an unstructured mesh, $\mathbb{T}_h \in \mathbb{R}^d$, over the spatial domain, where d is the spatial dimension

84 of the problem (for this work, $d = 2$). This mesh consists of non-overlapping spatial elements to form a

85 complete union over the entire spatial domain: $\mathcal{D} = \bigcup_{K \in \mathbb{T}_h} K$. The 2D domain is meshed with arbitrary

86 polygonal elements.

87   As is standard in the discontinuous Galerkin scheme, the angular flux unknown of Eq. (2) is expanded

88 using discontinuous basis functions over each mesh cell K:

$$\psi_m(\vec{r}) \approx \sum_{j=1}^{n_K} \psi_{m,j} b_j(\vec{r}). \tag{4}$$

89 The basis functions are nonzero over a single element and discontinuous across the element interfaces. The

90 DGFEM weak form for the transport equation is obtained by multiplying Eq. (2) by any test function b,

4

integrating over element K, applying the divergence theorem to the streaming term, and using the standard upwind technique for the angular flux at the element's boundaries. A compact notation for the weak form over element K, for an angular direction $\vec{\Omega}_m$, is as follows:

$$
\begin{aligned}
\left\langle \left(\vec{\Omega}_m \cdot \vec{n}\right) b, \psi_m \right\rangle_{\partial K^+} &- \left(\vec{\Omega}_m \cdot \vec{\nabla} b, \psi_m\right)_K + \left(\sigma_t b, \psi_m\right)_K \\
&= \left(\frac{\sigma_s}{4\pi} b, \phi\right)_K + (b, Q_m)_K - \left\langle \left(\vec{\Omega}_m \cdot \vec{n}\right) b, \psi_m^\uparrow \right\rangle_{\partial K^-}
\end{aligned}
\tag{5}
$$

In Eq. (5), $\partial K^\pm$ represents the outflow/inflow boundary of element K (with respect to direction $\vec{\Omega}_m$),

$$
\partial K^\pm = \left\{ \vec{r} \in \partial K \mid \vec{\Omega}_m \cdot \vec{n}_K(\vec{r}) \gtrless 0 \right\},
\tag{6}
$$

with $\vec{n}_K(\vec{r})$ the outward unit normal vector at location $\vec{r}$, $\psi_m^\uparrow$ are the angular flux values on an inflow face taken from the upwind neighbor element, and the inner products

$$
\left(u, v\right)_K \equiv \int_K u\, v\, dr \qquad \text{and} \qquad \left\langle u, v \right\rangle_{\partial K} \equiv \int_{\partial K} u\, v\, ds
\tag{7}
$$

correspond to integrations over the cell volume and faces, respectively, where $dr \in \mathbb{R}^d$ is within the cell and $ds \in \mathbb{R}^{d-1}$ is along the cell boundary. In Eq. (5), all angular flux unknowns "live on" element K, except for the upwinded values which are taken from the upwind neighboring element:

$$
\psi_m^\uparrow(\vec{r}) = \lim_{s \to 0^+} \psi_m(\vec{r} + s\vec{n}_K(\vec{r})), \quad \vec{r} \in \partial K^-.
\tag{8}
$$

The spatial convergence of DGFEM methods for hyperbolic systems has been extensively studied [3, 29, 30, 19]. With the discretized flux solution, $\phi_h \in W_\mathcal{D}^h$, corresponding to our unstructured mesh, $\mathbb{T}_h$, we can define an error for our DGFEM transport solution with the L_2-norm as

$$
||\phi_h - \phi_{exact}||_{L_2} \leq C \frac{h^q}{(p+1)^{q-1/2}}.
\tag{9}
$$

In Eq. (9), $q = \min(p+1, r)$, h is the maximum diameter of all the mesh elements, p is the polynomial completeness of the finite element function space, r is the regularity index of the transport solution, and C is a constant that is independent of the mesh employed. Therefore, the DGFEM transport convergence rates are limited by the regularity, r, of the solution space. With sufficiently smooth data the solution regularity is $r = 3/2$. However, in the case of a pure absorber or void, the solution regularity diminishes to $r = 1/2$ [19]. From these indices, one would think that the transport solution regularity would impede the use of higher-order finite element spaces. We note, however, that these regularity indices only apply to the asymptotic convergence range, which usually only applies to very fine meshes that are much smaller than typically employed meshes. Therefore, we expect to capture up to order $(p+1)$ in preasymptotic regimes that would be employed for a wide variety of transport problems.
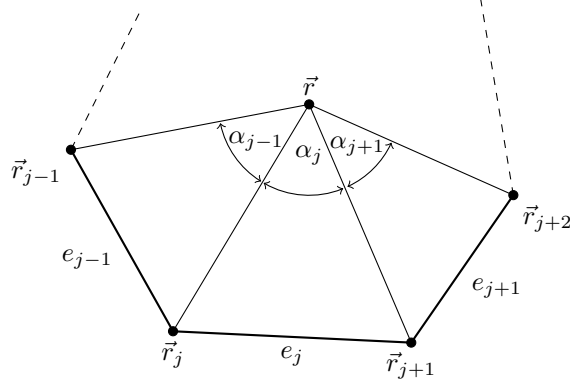
Figure 2: Arbitrary polygon with geometric properties used for 2D basis function generation.

We can also define the DGFEM transport solution convergence rates in terms of the total number of spatial degrees of freedom, N_{dof}, instead of the maximum element diameter, h. We first relate the total number of mesh elements, N_{ele}, to the maximum element diameter: $N_{ele} \propto h^{-d}$. We then assume that the finite element functional space on each element is the serendipity space [31]. This means that the number of degrees of freedom per element is proportional to the polynomial order: $N_{dof} \propto ph^{-d}$ or $h \propto \left(\frac{N_{dof}}{p} \right)^{-1/d}$. Substituting this relation into Eq. (9), yields an L_2-norm convergence rate in terms of the problem's total number of spatial degrees of freedom,

$$||\phi_h - \phi_{exact}||_{L_2} \leq C(p) \left(N_{dof} \right)^{-q/d}, \qquad C(p) = \frac{p^{q/d}}{(p+1)^{q-1/2}}. \tag{10}$$

3. Linear Polygonal Basis Functions

We now describe the linear polygonal functions used here. Geometric notations are given by Figure 2. To achieve the highest convergence rate of Eq. (9), these functions need to exactly interpolate the $\{1, x, y\}$ span of functions. For this work, we utilize a family of linear functions known as Generalized Barycentric Coordinates (GBC) [32]. In particular, we investigate the piecewise linear functions, Wachspress functions, the mean value functions, and the maximum entropy functions. These functions (denoted as $\{\lambda_j\}_{j=1}^n$ on a polygon with n vertices) have the following properties:

1. Positivity: $\lambda_j \geq 0$;

2. Partition of unity: $\sum_{j=1}^{n} \lambda_j = 1$;

3. Affine combination: $\sum_{j=1}^{n} \vec{r}_j \lambda_j(\vec{r}) = \vec{r}$;

4. Lagrange property: $\lambda_i(\vec{r}_j) = \delta_{ij}$;

6

131   5. Piecewise boundary linearity: $\lambda_j((1-\mu)\vec{r}_j + \mu\vec{r}_{j+1}) = (1-\mu)\lambda_j(\vec{r}_j) + \mu\lambda_j(\vec{r}_{j+1})$, $\mu \in [0,1]$.

132   We can see from properties 2 and 3 that the GBCs span the $\{1, x, y\}$ space and can exactly interpolate

133   constant and linear functions.

3.1. Piecewise Linear Functions

135   The first linearly-complete 2D polygonal functions that we analyze are the piecewise linear (PWL)

136   functions proposed by Stone and Adams [33, 10] and Bailey [11]. The 2D PWL functions are defined as

137   combinations of linear triangular functions, with some of them only having measure within a subregion of a

138   polygon. These subregions are formed by triangulating the arbitrary 2D polygon into a set of sub-triangles.

139   Each sub-triangle is defined by two adjacent vertices of the polygon and the polygon's centroid, $\vec{r}_c$. The

140   polygon's centroid can be defined by

$$\vec{r}_c = \sum_{j=1}^{n} \alpha_j \vec{r}_j, \tag{11}$$

141   where α_j are the vertex weights functions and,

$$\sum_{j=1}^{n} \alpha_j = 1. \tag{12}$$

142   For this work, we continue to use the definition for the vertex weight functions from previous works [33, 10,

143   11]: $\alpha_j = 1/n$. Hence, the weights are constant per polygonal cell, and the cell centroid simply becomes

144   the average position of all the vertices. Using these weights, the PWL basis function for vertex j, λ_j^{PWL}, is

145   defined as

$$\lambda_j^{PWL}(\vec{r}) = t_j(\vec{r}) + \alpha_j t_c(\vec{r}). \tag{13}$$

146   In Eq. (13), t_j is the standard 2D linear function with unity at vertex j that linearly decreases to zero at

147   the cell center and each adjoining vertex. t_c is the 2D cell "tent" function located at $\vec{r}_c$ which is unity at the

148   cell center and linearly decreases to zero at each cell vertex. The functional form of Eq. (13) with identical

149   weights means that the PWL function for vertex j linearly decreases to a value of $1/n$ at the polygonal

150   center. From there, the function linearly decreases to zero on all faces that are not connected to vertex j.

3.2. Wachspress Rational Functions

152   The second linearly-complete polygonal functions that we consider are the Wachspress rational func-

153   tions [34]. These rational functions were originally derived for 2D polygons and possess all the properties of

154   the barycentric functions. However, they are only valid interpolants over strictly-convex polygons. Weakly-

155   convex polygons (containing colinear vertices) lose piecewise linearity on the boundary. Concave polygons

7

have regions within their domain that are undefined and divide-by-zero occurs. The Wachspress functions (which we denote as λ^W) have the following form

$$\lambda_j^W(\vec{r}) = \frac{w_j(\vec{r})}{\sum\limits_{i=1}^{n} w_i(\vec{r})}, \tag{14}$$

where the Wachspress weight function for vertex j, w_j, has the following definition:

$$w_j(\vec{r}) = \frac{A(\vec{r}_{j-1}, \vec{r}_j, \vec{r}_{j+1})}{A(\vec{r}, \vec{r}_{j-1}, \vec{r}_j)\, A(\vec{r}, \vec{r}_j, \vec{r}_{j+1})}. \tag{15}$$

In Eq. (15), the terms $A(\vec{a}, \vec{b}, \vec{c})$ denote the signed area of the triangle with vertices $\vec{a}$, $\vec{b}$, and $\vec{c}$. Each of these signed areas can be computed by

$$A(\vec{a}, \vec{b}, \vec{c}) = \frac{1}{2} \begin{vmatrix} 1 & 1 & 1 \\ x_a & x_b & x_c \\ y_a & y_b & y_c \end{vmatrix}. \tag{16}$$

Through observation of Eq. (15), we can see that the denominator of w_j goes to zero on faces e_j and e_{j+1}. This makes direct numerical evaluation of the Wachspress functions impossible on the boundary of the element boundary. However, the functions do have valid limits on the boundary as shown in Appendix A.

3.3. Mean Value Functions

The third linearly-complete 2D polygonal functions that we analyze are the mean value functions (MV) [35]. These functions were derived to approximate harmonic maps on a polygon by a set of piecewise linear maps over a triangulation of the polygon for use in computer aided graphic design. The mean value function at vertex j, λ_j^{MV}, is defined as

$$\lambda_j^{MV}(\vec{r}) = \frac{w_j(\vec{r})}{\sum\limits_{i=1}^{n} w_i(\vec{r})}, \tag{17}$$

where the mean value weight function for vertex j, w_j, has the following definition:

$$w_j(\vec{r}) = \frac{\tan(\alpha_{j-1}/2) + \tan(\alpha_j/2)}{|\vec{r}_j - \vec{r}|}. \tag{18}$$

Similar to the Wachspress functions, we can observe from Eq. (18) that the numerator of w_j tends to infinity on faces e_j and e_{j+1} as $\alpha_j \to \pi/2$. Again, the MV functions have valid limits on the boundary as shown in Appendix A.

8

3.4. Maximum Entropy Functions

    The final linearly-complete 2D polygonal functions that we investigate are the maximum entropy func-

tions (ME) [36]. The principle of maximum entropy stems from the concept of Shannon entropy [37], where

the maximum of the Shannon entropy function, H, will lead to the least-biased statistical inference for

some set of testable constraints [38]. In FEM analysis, these testable constraints are the partition of unity

and the affine combination properties of the barycentric functions. For n discrete probability functions

(corresponding to the n vertex functions), the functional form for the Shannon entropy can be given by

$$H(\vec{b}, \vec{m}, \vec{r}) = -\sum_{j=1}^{n} b_j(\vec{r}) \log \left(\frac{b_j(\vec{r})}{m_j(\vec{r})} \right), \tag{19}$$

where m_j is called the prior distribution. If $(\lambda_j^{ME}(\vec{x}), j = 1, ..., n)$ is the solution of the following constrained

optimization problem

$$\max_{\vec{b}(\vec{r})} H(\vec{b}, \vec{m}, \vec{r}), \tag{20}$$

then the maximum entropy functions can be given by

$$\lambda_j^{ME}(\vec{r}) = \frac{w_j(\vec{r})}{\sum\limits_{i=1}^{n} w_i(\vec{r})}, \tag{21}$$

where the maximum entropy weight function for vertex j, w_j, has the following definition,

$$w_j(\vec{r}) = m_j(\vec{r}) \exp(-\vec{\kappa} \cdot (\vec{r}_j - \vec{r})), \tag{22}$$

and $\vec{\kappa}$ is a vector value of dimension d that will be explained shortly. In the context of Eq. (22), the

prior distribution, m_j, can be viewed as a weight function associated with vertex j. This means that there

is variability that one can employ for these weight functions. For FEM applications [39], an appropriate

functional form for the prior distribution is given by

$$m_j(\vec{r}) = \frac{\pi_j(\vec{r})}{\sum\limits_{k=1}^{n} \pi_k(\vec{r})}, \tag{23}$$

where

$$\pi_j(\vec{r}) = \prod_{k \neq j-1,j}^{n} \rho_k(\vec{r}) = \rho_1(\vec{r}) \ldots \rho_{j-2}(\vec{r}) \rho_{j+1}(\vec{r}) \ldots \rho_n(\vec{r}), \tag{24}$$

and

$$\rho_j(\vec{r}) = ||\vec{r} - \vec{r}_j|| + ||\vec{r} - \vec{r}_{j+1}|| - ||\vec{r}_{j+1} - \vec{r}_j||. \tag{25}$$

In Eqs. (24) and (25), we have defined a new function, ρ_j, that corresponds to face e_j between vertices j and $j + 1$. These face functions are zero along face e_j, but strictly positive elsewhere due to the triangle inequality. This means that the vertex function, π_j, is also non-negative and vanishes on all faces that are not adjacent to vertex j. We note that $m_j(\vec{r}_j) = 1$.

To complete the definition of the ME functions, we now explain how the $\vec{\kappa}$ vector in Eq. (22) is computed. We can see that once $\vec{\kappa}$ is known, the ME functions can be directly calculated. To obtain $\vec{\kappa}$, we solve the constrained optimization problem of Eq. (20) through the use of Lagrange multipliers with a Newton's method. Let κ_0 be the Lagrange multiplier for the constant constraint and $\vec{\kappa}$ be the Lagrangian multiplier for the affine constraint, then the Lagragian for the problem of Eq. (20) is given by

$$\mathcal{L}(\vec{b}; \kappa_0, \vec{\kappa}) = -\sum_{j=1}^{n} b_j \log\left(\frac{b_j}{m_j}\right) - \kappa_0 \left(\sum_{j=1}^{n} b_j - 1\right) - \vec{\kappa} \cdot \left(\sum_{j=1}^{n} b_j \left(\vec{r}_j - \vec{r}\right)\right), \tag{26}$$

where we omitted the spatial parameter $\vec{r}$ for brevity and differentiate between $\vec{b}$ and $\vec{\kappa}$ being vectors of lengths n and d, respectively. If we define $W = \sum_{j=1}^{n} w_j(\vec{r})$, then the solution of Eq. (21) is equivalent to solving the following dual unconstrained optimization problem [40]:

$$\vec{\kappa}^* = \min_{\vec{\kappa}} F(\vec{\kappa}), \qquad F(\vec{\kappa}) = \log W(\vec{\kappa}). \tag{27}$$

3.5. Summary of the Linear Polygonal Basis Functions

For this work, we are analyzing four different linear barycentric functions for use as our FEM basis functions: the piecewise linear functions, the Wachspress rational functions, the mean value functions, and the maximum entropy functions. Table 1 provides a summary of the properties for the different functions. The PWL, MV, and ME functions can exactly interpolate linear functions on degenerate-convex and concave polygons while the Wachspress functions can only exactly interpolate linear functions on strictly-convex polygons. Using geometric information for a given polygonal cell, the PWL, Wachspress, and MV functions can be directly evaluated, whereas the ME functions require an iterative approach with Newton's method since their functional form constitutes a non-linear optimization problem. Finally, the elementary matrices of Eq. (5) can be integrated analytically with the PWL functions. The Wachspress, MV, and ME functions need to be integrated numerically.

We conclude our discussion of the linear barycentric functions by providing some example contour plots. Figure 3 provides the contour plots of the different functions located at vertex $(0, 1)$ on the unit square. Figure 4 provides the contour plots on the degenerate pentagon that is formed by inserting a vertex at $(1/2, 1)$ into the unit square. This re-emphasizes that the Wachspress functions are only valid interpolatory functions on strongly-convex polygons. Finally, Figure 5 provides the contour plots of the PWL, MV, and ME functions on the L-shaped domain at the $(0, 1)$ and $(1/2, 1/2)$ vertices.

Table 1: Summary of the properties of the 2D barycentric functions.

Basis Function	Polygon Type	Evaluation	Integration
Piecewise Linear	Convex/Concave	Direct	Analytical
Wachspress	Convex	Direct	Numerical
Mean Value	Convex/Concave	Direct	Numerical
Max Entropy	Convex/Concave	Iterative	Numerical

(a) PWL (b) Wachspress (c) Mean Value (d) Maximum Entropy
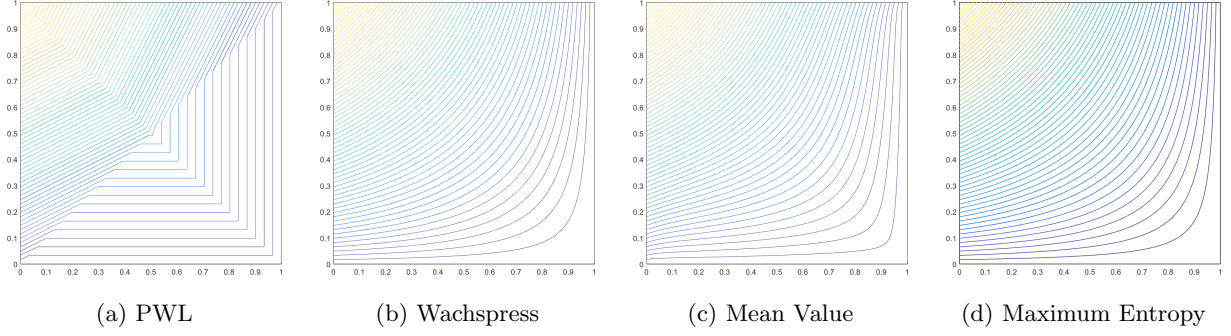
Figure 3: Contour plots of the different linear basis function on the unit square located at vertex (0,1).

(a) PWL (b) Wachspress (c) Mean Value (d) Maximum Entropy

(e) PWL (f) Wachspress (g) Mean Value (h) Maximum Entropy
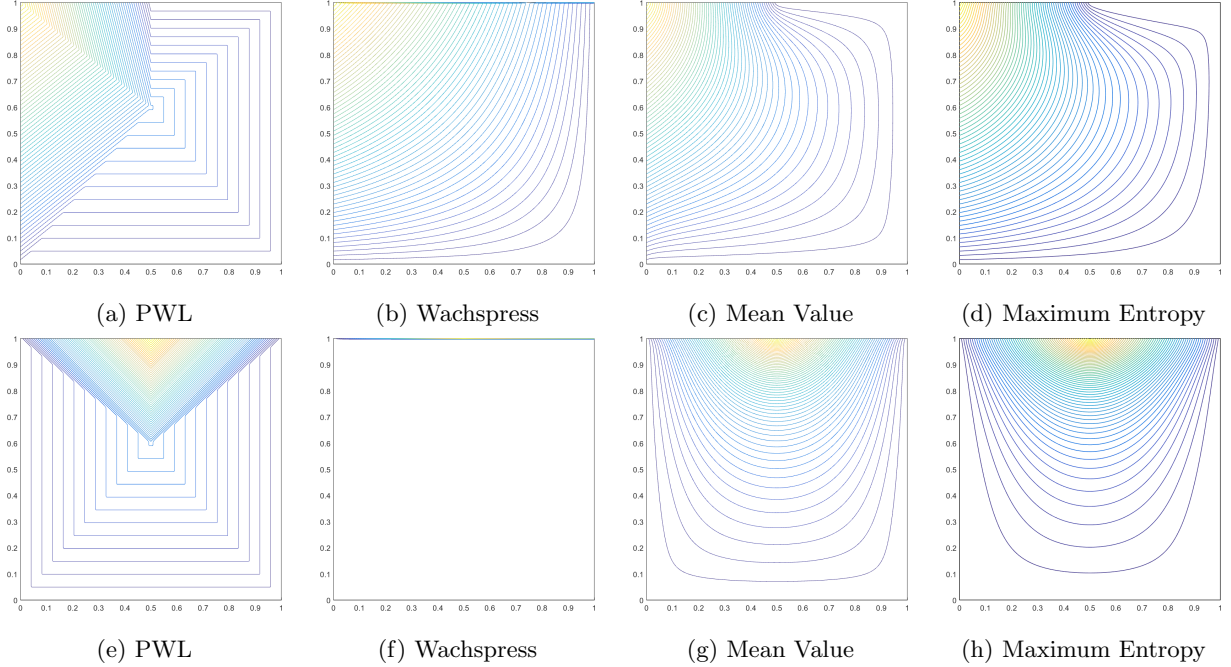
Figure 4: Contour plots of the different linear basis function on the degenerate pentagon located at vertex (0,1) (a-d) and vertex (1/2,0) (e-h). The Wachspress functions clearly fail for the weakly convex case.
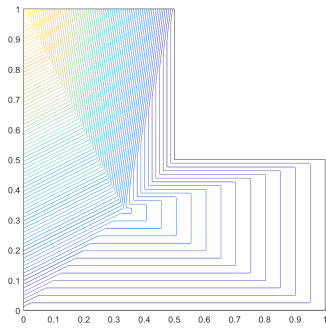
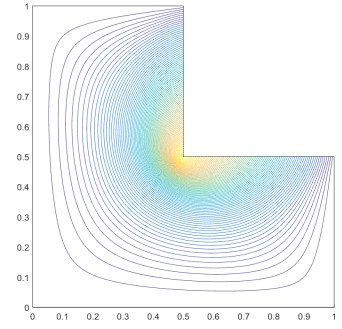(a) PWL   (b) Mean Value   (c) Maximum Entropy
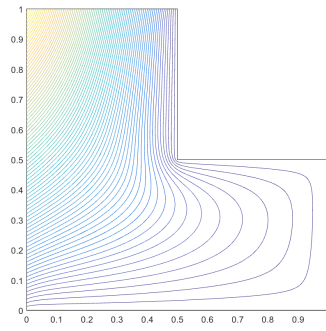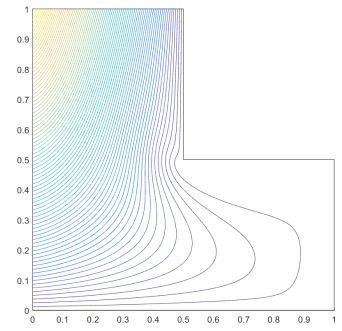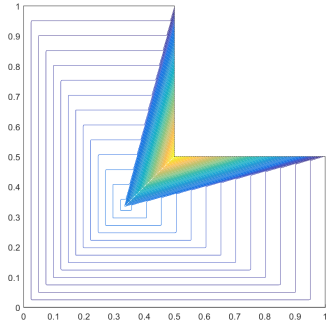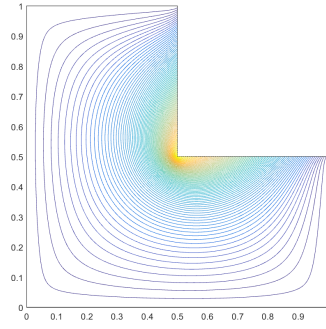
(d) PWL   (e) Mean Value   (f) Maximum Entropy

Figure 5: Contour plots of the mean value and maximum entropy functions on the L-shaped domain at vertex (0,1) (a-c) and vertex $(1/2, 1/2)$ (d-f).
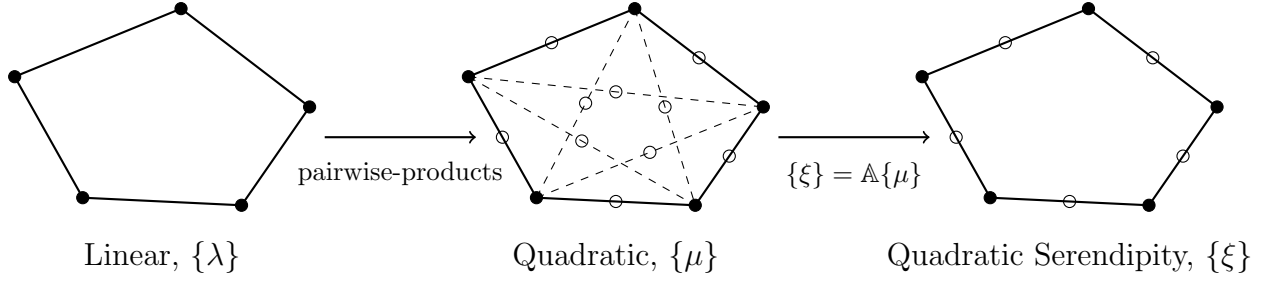
Figure 6: Overview of the process to construct the quadratic serendipity basis functions on polygons. The filled dots correspond to basis functions that maintain the Lagrange property while empty dots do not.

4. Quadratic Serendipity Polygonal Basis Functions

We have fully-defined the linearly-complete generalized barycentric functions that will be used to derive the quadratic serendipity functions. These 2D serendipity functions can exactly interpolate the $\{1, x, y, x^2, xy, y^2\}$ span of functions. Next, we describe how to convert any generalized barycentric function into the quadratic serendipity space of functions to yield quadratic precision based on the work of Rand et al. [26]. The linear maximum entropy functions were independently converted into the quadratic serendipity space by other researchers [41, 42].

Figure 6 gives a visual depiction of this conversion process. For a polygon with n vertices, we can summarize this procedure as the following:

1. For a point $\vec{r}$, compute the n linear barycentric functions $\{\lambda_i(\vec{r})\}$ (*e.g.*, Wachspress, PWL, mean value, or maximum entropy);

2. Take all non-repeating pairwise products of the linear functions to obtain $\frac{n(n+1)}{2}$ quadratic functions $\{\mu_{ab}(\vec{r})\}$;

3. Form the linear transformation matrix $\mathbb{A}$ through the use of monomial constraint equations;

4. Use the $\mathbb{A}$ matrix to reduce the $\{\mu_{ab}(\vec{r})\}$ function set to the $2n$ serendipity basis set $\{\xi_{ij}(\vec{r})\}$.

We begin by computing the $(i = 1, ..., n)$ linear barycentric functions, $\{\lambda_i(\vec{r})\}$, and their gradients, $\{\vec{\nabla}\lambda_i(\vec{r})\}$, for a point $\vec{r}$. These evaluation points will correspond to spatial quadrature points to compute the FEM integrals of Eq. (5). These linearly-complete barycentric functions can be converted immediately to barycentric-like functions with quadratic precision. Taking all non-repeating pairwise products of the linear functions yields a total of $N_Q = \frac{n(n+1)}{2}$ quadratic functions: $\mu_{ab} = \lambda_a \lambda_b$ $(1 \leq a \leq n; 1 \leq a \leq b)$. Doing this generates functions that either live on the polygon's vertices, mid-face points, or midpoints of the polygon's interior diagonals between two vertices as seen in Figure 6. The set of n vertex functions is denoted as V $(a = b)$, the set of n mid-face (mid-edge) functions is denoted as E $(|a - b| = 1)$, and the set of $\frac{n(n-3)}{2}$ interior functions is denoted as I $(|a - b| > 1)$. For the mid-edge and interior functions,

13

only 1 combination of ab is kept since $\mu_{ab} = \mu_{ba}$. We also define the abbreviated notation of $\vec{r}_{ab} = \frac{\vec{r}_a + \vec{r}_b}{2}$. Using these various notations, we can write the precision properties of the μ_{ab} functions for the constant constraint,

$$\sum_{aa \in V} \mu_{aa}(\vec{r}) + \sum_{ab \in E \cup I} 2\mu_{ab}(\vec{r}) = 1, \tag{28}$$

for the linear constraint,

$$\sum_{aa \in V} \mu_{aa}(\vec{r})\, \vec{r}_a + \sum_{ab \in E \cup I} 2\mu_{ab}(\vec{r})\, \vec{r}_{ab} = \vec{r}, \tag{29}$$

and for the quadratic constraint,

$$\sum_{aa \in V} \mu_{aa}(\vec{r})\, (\vec{r}_a \otimes \vec{r}_a) + \sum_{ab \in E \cup I} \mu_{ab}(\vec{r})\, (\vec{r}_a \otimes \vec{r}_b + \vec{r}_b \otimes \vec{r}_a) = \vec{r} \otimes \vec{r}. \tag{30}$$

In Eq. (30), $\otimes$ is the dyadic tensor product. We can immediately see that quadratic precision is ensured. However, the set of these pairwise quadratic functions grows quadratically. This means that as n grows large, the number of interpolatory functions grows as order $O(n^2)$, but still only maintains precision of the $\{1, x, y, x^2, xy, y^2\}$ space of functions. Therefore, the computational work required to utilize these quadratic barycentric functions can become prohibitive for polygons with large vertex counts.

To minimize the number of interpolatory functions but still maintain the precision of the $\{1, x, y, x^2, xy, y^2\}$ span of functions, we seek to convert the quadratic barycentric functions into the quadratic serendipity space of functions $\{\xi_{ij}\}$. This quadratic serendipity space only contains the vertex and mid-face functions (total of $2n$) and has been extensively studied for tensor-based elements in the past [31]. This means that we seek to reduce the $\{\mu_{ab}\}$ set of functions by removing the interior functions ($ab \in I$) while still maintaining quadratic precision. If we define ξ_{ii} and $\xi_{i(i+1)}$ as the serendipity functions that live at vertex i and the mid-face point between vertices i and $i+1$, respectively, then we can write the serendipity precision properties for the constant constraint,

$$\sum_{ii \in V} \xi_{ii}(\vec{r}) + \sum_{i(i+1) \in E} 2\xi_{i(i+1)}(\vec{r}) = 1, \tag{31}$$

for the linear constraint,

$$\sum_{ii \in V} \xi_{ii}(\vec{r})\, \vec{r}_{ii} + \sum_{i(i+1) \in E} 2\xi_{i(i+1)}(\vec{r})\, \vec{r}_{i(i+1)} = \vec{r}, \tag{32}$$

and for the quadratic constraint,

$$\sum_{ii \in V} \xi_{ii}(\vec{r})\, (\vec{r}_i \otimes \vec{r}_i) + \sum_{i(i+1) \in E} \xi_{i(i+1)}(\vec{r})\, (\vec{r}_i \otimes \vec{r}_{i+1} + \vec{r}_{i+1} \otimes \vec{r}_i) = \vec{r} \otimes \vec{r}. \tag{33}$$

14

To remove the interior functions, we formalize this procedure by recasting it as a linear algebra problem. We seek a matrix $\mathbb{A}$ such that the linear transformation,

$$\{\xi\} = \mathbb{A}\{\mu\}, \tag{34}$$

will satisfy the precision properties of Eqs. (31 - 33). It is easy to see that $\mathbb{A}$ has dimension $(2n \times N_Q)$. We wish for this matrix to have constant entries for any point within the polygon's interior so that it does not have to be recalculated for each interpolatory quadrature point of interest. To ease the notation, we will assign specific basis orderings for the quadratic and quadratic serendipity functions. The serendipity basis is ordered such that all vertex functions ($ii \in V$) in a counter-clockwise ordering are first, followed by a counter-clockwise ordering of the mid-face nodes ($ij \in E$) starting with the node between vertices 1 and 2. This ordering can be succinctly stated by

$$\{\xi_{ij}\} = \left\{ \left[\xi_{11}, \xi_{22}, ..., \xi_{nn}\right], \left[\xi_{12}, \xi_{23}, ..., \xi_{(n-1)n}, \xi_{n(n+1)}\right] \right\}, \tag{35}$$

where $n + 1$ is understood to be 1. The quadratic basis begins identically to the serendipity basis by first listing the vertex and mid-face functions. Then the interior functions are indexed in lexicographical order. This gives the following ordering for the quadratic functions

$$\begin{aligned}\{\mu_{ab}\} = \Big\{ &\left[\mu_{11}, \mu_{22}, ..., \mu_{nn}\right], \left[\mu_{12}, \mu_{23}, ..., \mu_{(n-1)n}, \mu_{n(n+1)}\right], \\ &\left[\mu_{13}, ..., (\text{lexicographical}), ..., \mu_{(n-2)n}\right] \Big\}.\end{aligned} \tag{36}$$

With these basis orderings, we can now denote the entries of $\mathbb{A}$ (given by c_{ab}^{ij}) as

$$\mathbb{A} = \begin{bmatrix} c_{11}^{11} & \cdots & c_{ab}^{11} & \cdots & c_{(n-2)n}^{11} \\ \cdots & \ddots & \vdots & \ddots & \vdots \\ c_{11}^{ij} & \cdots & c_{ab}^{ij} & \cdots & c_{(n-2)n}^{ij} \\ \cdots & \ddots & \vdots & \ddots & \vdots \\ c_{11}^{n(n+1)} & \cdots & c_{ab}^{n(n+1)} & \cdots & c_{(n-2)n}^{n(n+1)} \end{bmatrix}, \tag{37}$$

A sufficient set of constraints for the entries in $\mathbb{A}$ to ensure the precision properties of Eqs. (31 - 33) can be written for the constant constraint,

$$\begin{aligned} \sum_{ii \in V} c_{aa}^{ii} + \sum_{i(i+1) \in E} 2c_{aa}^{i(i+1)} &= 1, \qquad \forall aa \in V \\ \sum_{ii \in V} c_{ab}^{ii} + \sum_{i(i+1) \in E} 2c_{ab}^{i(i+1)} &= 2, \qquad \forall ab \in E \cup I \end{aligned} \tag{38}$$

for the linear constraint,

15

$$\sum_{ii \in V} c_{aa}^{ii} \vec{r}_{ii} + \sum_{i(i+1) \in E} 2c_{aa}^{i(i+1)} \vec{r}_{i(i+1)} = \vec{r}_{aa}, \qquad \forall aa \in V$$

$$\sum_{ii \in V} c_{ab}^{ii} \vec{r}_{ii} + \sum_{i(i+1) \in E} 2c_{ab}^{i(i+1)} \vec{r}_{i(i+1)} = 2\vec{r}_{ab}, \qquad \forall ab \in E \cup I \tag{39}$$

for the $a \in V$ vertex quadratic constraints,

$$\sum_{ii \in V} c_{aa}^{ii} \vec{r}_{ii} \otimes \vec{r}_{ii} + \sum_{i(i+1) \in E} c_{aa}^{i(i+1)} \left(\vec{r}_i \otimes \vec{r}_{i+1} + \vec{r}_{i+1} \otimes \vec{r}_i \right) = \vec{r}_{aa} \otimes \vec{r}_{aa}, \tag{40}$$

and for the $ab \in E \cup I$ mid-face and interior quadratic constraints,

$$\sum_{ii \in V} c_{ab}^{ii} \vec{r}_{ii} \otimes \vec{r}_{ii} + \sum_{i(i+1) \in E} c_{ab}^{i(i+1)} \left(\vec{r}_i \otimes \vec{r}_{i+1} + \vec{r}_{i+1} \otimes \vec{r}_i \right) = \left(\vec{r}_a \otimes \vec{r}_b + \vec{r}_b \otimes \vec{r}_a \right). \tag{41}$$

For $n > 3$, there are more coefficients than the six constraint equations. This means that there is flexibility in the construction of the solution to the constraint equations. Therefore, we choose a simple structure for $\mathbb{A}$ that consists of the following,

$$\mathbb{A} = [\mathbb{I} \,|\, \mathbb{A}'], \tag{42}$$

where $\mathbb{I}$ is the $(2n \times 2n)$ identity matrix, and $\mathbb{A}'$ is a full $(2n \times \frac{n(n-3)}{2})$ matrix. This means that the vertex and face midpoint serendipity functions, ξ_{ij}, are formed by taking their corresponding quadratic function and adding some linear combination of the interior functions. Therefore, we only need to determine the $\frac{n(n-3)}{2}$ columns of the $\mathbb{A}'$ matrix to complete this linear transformation.

In their work, Rand et al. proposed a methodology where only six coefficients are chosen to be non-zero and can be directly calculated through geometric expressions. However, their approach is only valid for strictly-convex polygons. This will not work for our analysis since we wish to also analyze degenerate polygons that will arise in our AMR calculations. Therefore, we will use a least squares method to calculate each of the columns of $\mathbb{A}'$. We note that the coefficients calculated with this least squares method and that of Rand will be identical only for rectangles. If we isolate the column (ab) from $\mathbb{A}'$, then we can form the following system of equations,

$$\mathbb{B}\vec{c}_{ab} = \vec{q}_{ab}, \tag{43}$$

where $\mathbb{B}$ is a matrix of dimension (6x2n), $\vec{c}_{ab}$ is a vector of length $2n$, and $\vec{q}_{ab}$ is a vector of length 6. The entries of $\mathbb{B}$ correspond to coefficients given in the left-hand-side terms of Eqs. (38), (39), and (41). The values of $\vec{q}_{ab}$ are given by the right-hand-sides of these same equations. To invert $\mathbb{B}$, we use the Moore-Penrose pseudoinverse (denoted as $\mathbb{B}^*$) [43]. For an under-determined system of equations, the pseudoinverse is given by,

$$\mathbb{B}^* = \mathbb{B}^T(\mathbb{B}\mathbb{B}^T)^{-1}. \tag{44}$$

We note that we have only tested this methodology on convex and weakly-convex polygons. Once all of the coefficients for the $\mathbb{A}'$ matrix are known, each of the quadratic serendipity functions can be computed by

$$\xi_{ij} = \lambda_i \lambda_j + \sum_{ab \in I} c_{ab}^{ij} \lambda_a \lambda_b, \tag{45}$$

where $j = i$ for the vertex functions and $j = i + 1$ for the mid-face functions. From Eq. (45), we can clearly see that the quadratic functions, μ_{ab}, act as an intermediate step and do not ever need to be explicitly computed.

We now present some example contour plots of the different quadratic serendipity functions. Figure 7 provides contour plots of the functions on the unit square. Then, Figure 8 provides the contour plots of the PWL, MV, and ME functions on the degenerate square that is formed by inserting a vertex at $(1/2, 1)$ onto the unit square.

| (a) PWL | (b) Wachspress | (c) Mean Value | (d) Maximum Entropy |

| (e) PWL | (f) Wachspress | (g) Mean Value | (h) Maximum Entropy |
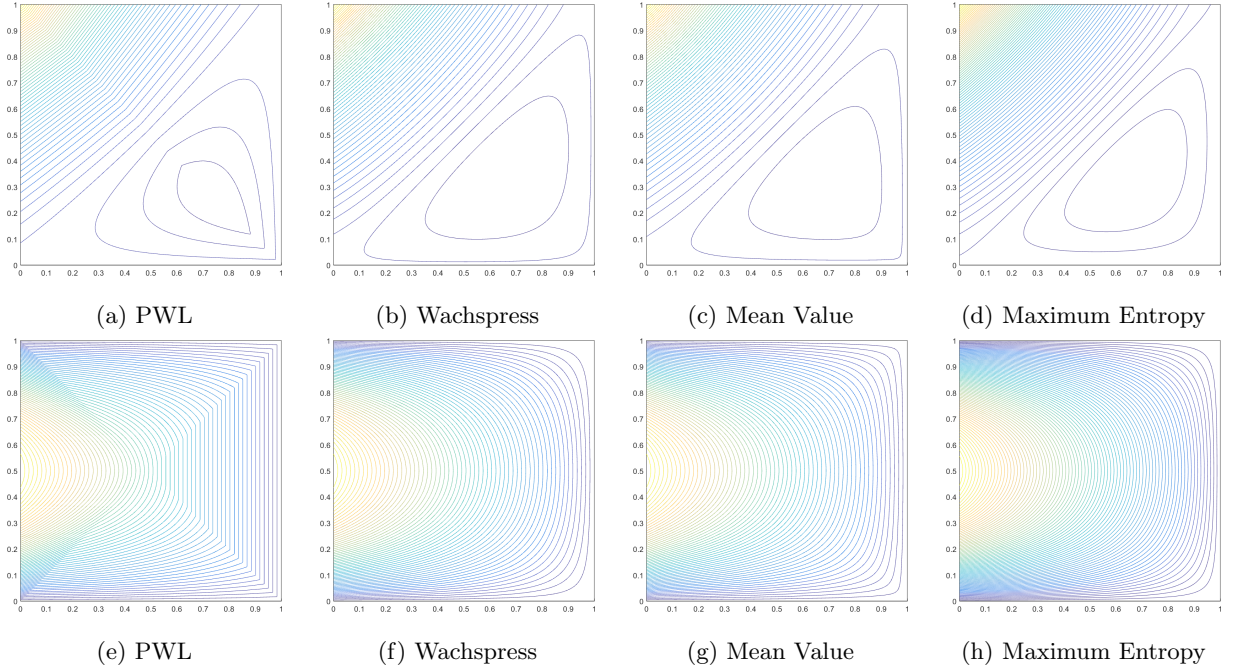
Figure 7: Contour plots of the different quadratic serendipity basis function on the unit square located at vertex (0,1) (a-d) and vertex (0,1/2) (e-h).

5. Adaptive Mesh Refinement Using Polygonal Basis Functions

Mesh adaptivity is a methodology that can accurately and efficiently capture solutions with strong and rapid spatial variations [44]. With spatial AMR, the mesh at refinement level ℓ, $\mathbb{T}_h^\ell$, is formed by selectively

17

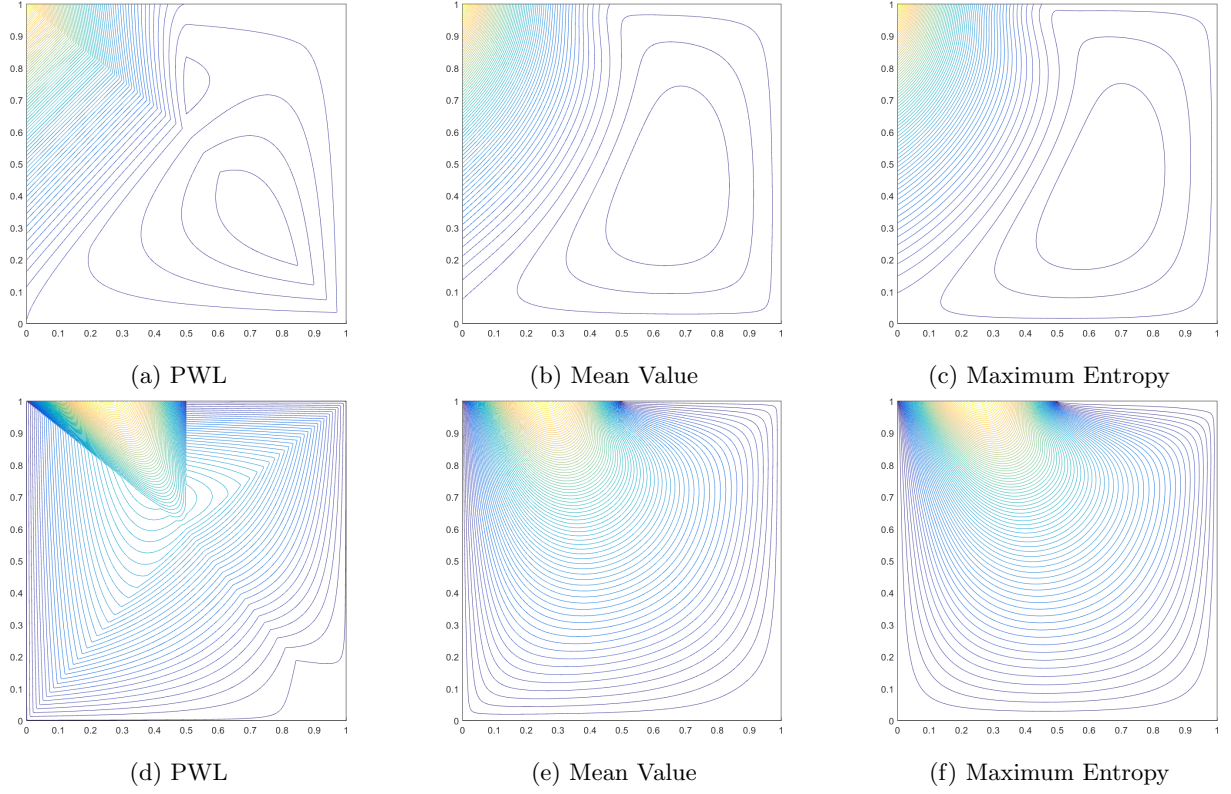|		
(a) PWL	(b) Mean Value	(c) Maximum Entropy
(d) PWL	(e) Mean Value	(f) Maximum Entropy

Figure 8: Contour plots of the different quadratic serendipity basis function on the unit degenerate pentagon located at vertex (0,1) (a-c) and mid-face node (1/4,1) (d-f).

refining mesh cells from $\mathbb{T}_h^{\ell-1}$ based on local error estimates. Using discontinuous functions, the magnitude of the solution jump across cell faces is proportional to the degree of spatial variation. Therefore, a heuristic local error indicator can monitor the interfacial jumps [45]. We define this jump-based error indicator for cell K and refinement level ℓ as

$$\eta_K^\ell = \int_{\partial K} [\![\phi^\ell]\!]^2 d\,s = \int_{\partial K} \left(\sum_m w_m [\![\psi_m^\ell]\!] \right)^2, \tag{46}$$

where $[\![\cdot]\!]$ is the jump operator along a face defined as

$$[\![\phi(\vec{r})]\!] = \phi^+(\vec{r}) - \phi^-(\vec{r}), \tag{47}$$

and $\phi^\pm(\vec{r}) = \lim_{s\to 0^\pm} \phi(\vec{r} + s\vec{n})$. With the error estimates defined for all cells $K \in \mathbb{T}_h^\ell$, the criterion to determine the cells flagged for refinement is

$$\eta_K^\ell \geq \alpha \max_{K' \in \mathbb{T}_h^\ell} \left(\eta_{K'}^\ell \right), \tag{48}$$

18

where α is a user-defined value $(0, 1)$. Equation (48) states that cells selected for refinement have error, normalized to the cell with the largest magnitude error, that is greater than or equal to α.

For this work, we will only utilize h-type refinement strategies for 2D transport problems on initial meshes containing only quadrilateral cells. Refined parent cells are subdivided into four daughter cells where each daughter cell contains four vertices: one of the corner vertices of its parent, the midpoint of the two faces containing that corner, and the parent centroid.

6. Numerical Results

6.1. Exactly-Linear Transport Solutions

We first verify that the linear polygonal finite element basis functions capture an exactly-linear solution space. We will show this by the method of manufactured solutions (MMS) [46]. We build our exact solution by investigating the 2D, one energy group transport problem with no scattering and an angle-dependent distributed source,

$$\mu \frac{\partial \psi}{\partial x} + \eta \frac{\partial \psi}{\partial y} + \sigma_t \psi = Q(x, y, \mu, \eta), \tag{49}$$

where the streaming term was separated into the corresponding two-dimensional terms. We then define an angular flux solution that is linear in both space and angle along with the corresponding scalar flux solution:

$$\psi(x, y, \mu, \eta) = ax + by + c\mu + d\eta + e$$
$$\phi(x, y) = 4\pi \left(ax + by + e \right). \tag{50}$$

Inserting the angular flux solution of Eq. (50) into Eq. (49) yields the following functional form for the distributed source:

$$Q(x, y, \mu, \eta) = a\mu + b\eta + \sigma_t \left(c\mu + d\eta \right) + \sigma_t \left(ax + by + e \right). \tag{51}$$

Our boundary conditions for all inflow boundaries are uniquely determined by the angular flux solution of Eq. (50). We test the linear functions on six different mesh types, including triangular, quadrilateral, and polygonal meshes:

1. Orthogonal Cartesian mesh formed by the intersection of 11 equally-spaced vertices in both the x and y dimensions. This forms a 10x10 array of quadrilateral mesh cells.

2. Ordered-triangular mesh formed by the bisection of the previous orthogonal Cartesian mesh (forming 200 triangles all of the same size/shape).

3. Polygonal Shestakov grid formed by the randomization of vertices based on a skewness parameter [47, 48]. With a certain range of this skewness parameter, highly distorted meshes can be generated.

19

345   4. Sinusoidal polygonal grid that is generated by the transformation of a uniform orthogonal grid based

346   on a sinusoid functional [49].

347   5. Kershaw's quadrilateral z-mesh [50]. This mesh is formed by taking an orthogonal quadrilateral grid

348   and displacing certain interior vertices only in the y dimension.

349   6. A polygonal variant of the quadrilateral z-mesh.

350 The polygonal Shestakov grid, sinusoidal polygonal grid, and the polygonal z-mesh grid are formed by the

351 conversion of an underlying quadrilateral grid into a corresponding polygonal grid by computing a bounded

352 Voronoi diagram.

353   For this example, we use $\sigma_t = a = c = d = e = 1.0$ and $b = 1.5$. The solution has a tilt in space $(a \neq b)$

354 so that it would not align with the triangular mesh. An S_8 level-symmetric (LS) quadrature set [51] is used

355 for all combinations of basis functions and mesh types. Figure 9 provides contour plots of the scalar flux

356 solution for the Wachspress functions on the different meshes. We can see that an exactly-linear solution is

357 captured as shown by the unbroken nature of the contour lines. The PWL, MV, and ME functions provide

358 identical results and are not shown for brevity.

359 *6.2. Exactly-Quadratic Transport Solutions*

360   We next demonstrate that the quadratic serendipity basis functions capture the appropriate functional

361 space, using two different MMS solutions. The first problem interpolates the $\{1, x, y, x^2, xy, y^2\}$ span of

362 functions, which we denote with the following exactly-quadratic spatial solution, $\{\psi_q, \phi_q\}$:

$$\psi_q(x, y, \mu, \eta) = a + bx + cy + dxy + ex^2 + fy^2 + g\mu + h\eta + i\mu\eta + j\mu^2 + k\eta^2$$
$$\phi_q(x, y) = 4\pi \left(a + \frac{j+k}{3} + bx + cy + dxy + ex^2 + fy^2 \right). \tag{52}$$

363 We clearly see that any combination of positive or negative (non-zero) values for the coefficients $a - k$ will

364 span the quadratic serendipity space, if the angular quadrature set can exactly integrate the $\{\mu, \eta, \mu^2, \mu\eta, \eta^2\}$

365 functions. The second problem contains terms up to x^2y^2. This higher-order functional form has a solution,

366 $\{\psi_{x2y2}, \phi_{x2y2}\}$, given by the following

$$\psi_{x2y2}(x, y) = x (L_x - x) y (L_y - y)$$
$$\phi_{x2y2}(x, y) = 4\pi x (L_x - x) y (L_y - y) \tag{53}$$

367 where L_x and L_y are the dimensions of the domain in x and y. For this problem, we set $L_x = L_y = 1$ and

368 use the S_8 LS quadrature set.

369   From the methodology presented in Section 4, we expect the quadratic serendipity functions to exactly

370 capture the ϕ_q solution but not the ϕ_{x2y2} solution. For both problems, we use the same meshes utilized for

371 the exactly-linear problem in Section 6.1. Tables 2 and 3 give the L_2-norm of the error for the quadratic
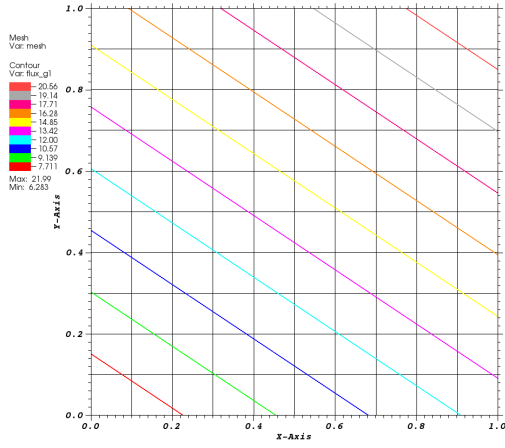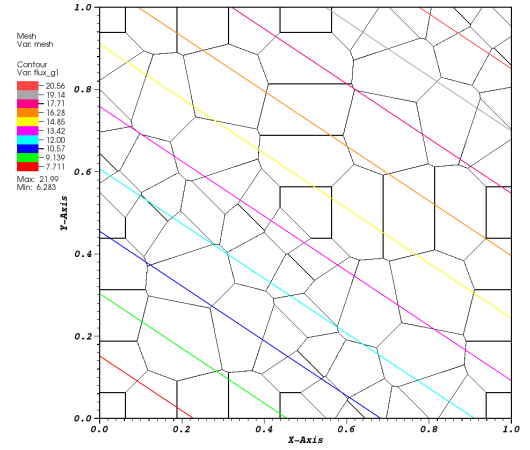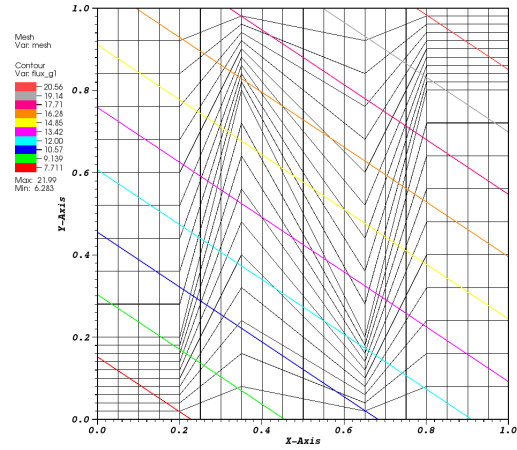
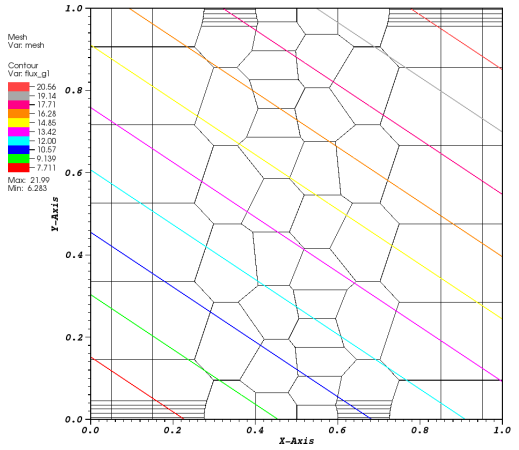(a) Cartesian
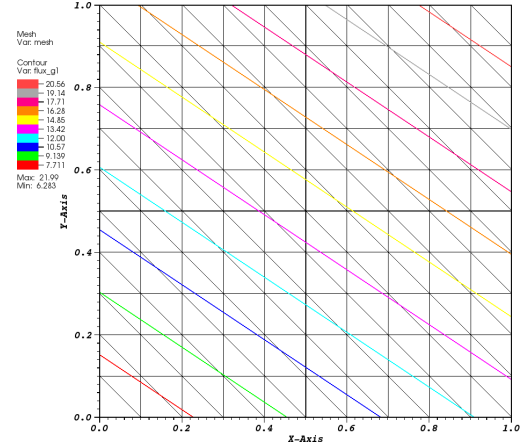
(b) Triangular
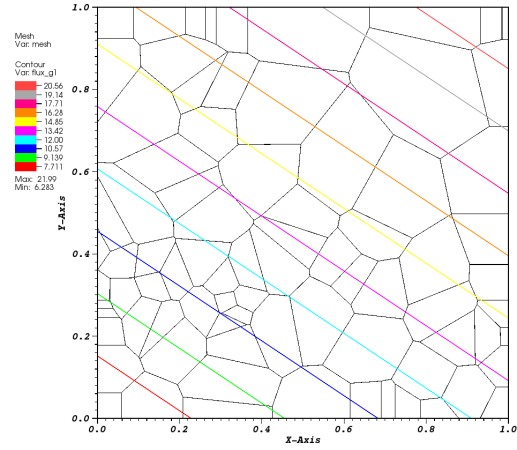
(c) Shestakov Polygons

(d) Sinusoid Polygons

(e) Z-Quadrilaterals

(f) Z-Polygons

Figure 9: Contour plots of the exactly-linear solution with the Wachspress basis functions. The PWL, MV, and ME functions yield identical results.

21

and $x^2 y^2$ solutions, respectively. We can clearly see from the machine precision results, that the quadratic serendipity functions exactly capture the ϕ_q solution. We note that the ME functions have higher norms since a tolerance value of 10^{-14} was used for the Newton iterations to compute the linear ME functions. Likewise, we can see from Table 3 that the functions clearly do not capture the ϕ_{x2y2} solution.

Table 2: L_2-norm of the error in the quadratic solution spanning $\{1, x, y, x^2, xy, y^2\}$ for the different quadratic serendipity basis functions on different mesh types.

Mesh Type	Basis Functions			
	PWL	Wachspress	Mean Value	Max. Entropy
Cartesian	1.31e-15	8.53E-16	1.70e-15	2.73e-14
Triangular	2.03e-15	9.14E-16	1.41e-15	7.38e-14
Shes. Poly	1.59e-15	1.14e-15	1.51e-15	3.95e-13
Sine Poly	1.45e-15	1.20e-15	1.48e-15	1.05e-13
Z-Poly	2.08e-15	1.08e-15	2.44e-15	1.77e-14
Z-Quad	2.79e-15	1.13e-15	1.89e-15	3.99e-13

Table 3: L_2-norm of the error in the quadratic solution containing the $x^2 y^2$ term for the different quadratic serendipity basis functions on different mesh types.

Mesh Type	Basis Functions			
	PWL	Wachspress	Mean Value	Max. Entropy
Cartesian	2.72e-05	3.50e-06	8.29e-06	3.86e-05
Triangular	5.13e-05	5.13e-05	5.13e-05	5.13e-05
Shes. Quad	3.37e-04	3.97e-04	2.81e-04	3.91e-04
Sine Poly	1.22e-04	3.75e-05	7.62e-05	1.39e-04
Z-Poly	3.07e-05	2.93e-05	2.60e-05	3.46e-05
Z-Quad	8.73e-05	2.98e-05	5.08e-05	1.17e-04

6.3. Convergence Rate Analysis Using MMS

After verifying the interpolation space of the linear and quadratic serendipity functions, we now demonstrate that the functions capture the appropriate theoretical convergence rates, by again using MMS. We demonstrate this on polygonal meshes generated by Voronoi diagrams in Section 6.3.1 and by polygonal meshes generated by spatial AMR in Section 6.3.2.

6.3.1. MMS on Voronoi Polygonal Meshes

For the first MMS example, we will analyze convergence rates on polygonal meshes generated from Voronoi diagrams. We use the PolyMesher mesh generator [52] to generate all of our polygonal grids. The functional form of the MMS solution is a sinusoid, $\{\psi_s, \phi_s\}$, having the following parameterized form,

$$
\begin{aligned}
\psi_s(x,y) &= \sin(\nu\frac{\pi x}{L_x})\sin(\nu\frac{\pi y}{L_y}), \\
\phi_s(x,y) &= 4\pi \sin(\nu\frac{\pi x}{L_x})\sin(\nu\frac{\pi y}{L_y}),
\end{aligned}
\tag{54}
$$

where ν is a frequency parameter ($\nu = 1, 2, 3, ...$) that we have set to 3 for this problem. The domain is $[0,1]^2$, which makes $L_x = L_y = 1$. The domain is filled with a pure absorber that has a total cross section of $\sigma_t = 1$. A S_8 LS quadrature is used.

We first provide examples of the polygonal meshes that are employed for this problem along with the corresponding MV isoline solutions in Figure 10. Upon convergence, PolyMesher provides grids with mesh cells composed of approximately regular polygons. From the isolines, it is apparent that the quadratic functions better capture the sinusoid solution than the linear functions. This result is obvious since the curvature of the quadratic functions can better reproduce the smoothly varying nature of the sinusoid solution. Then, Figure 11 provides the convergence rates for all of the functions in terms of the global number of degrees of freedom. The expected $p+1$ convergence rates are observed with slopes of $-(p+1)/2$ per Eq. (10).

6.3.2. MMS using Adaptive Mesh Refinement

For the second MMS example, we will analyze convergence rates on polygonal meshes generated from spatial AMR. The functional form of the MMS solution that will be analyzed is a localized Gaussian. This Gaussian solution space, $\{\psi_g, \phi_g\}$, that has its local maximum centered at (x_0, y_0) has the parameterized form,

$$
\begin{aligned}
\psi_g(x,y) &= C_M x(L_x - x)y(L_y - y)\exp(-\frac{(x-x_0)^2 + (y-y_0)^2}{\gamma}), \\
\phi_g(x,y) &= 4\pi C_M x(L_x - x)y(L_y - y)\exp(-\frac{(x-x_0)^2 + (y-y_0)^2}{\gamma}),
\end{aligned}
\tag{55}
$$

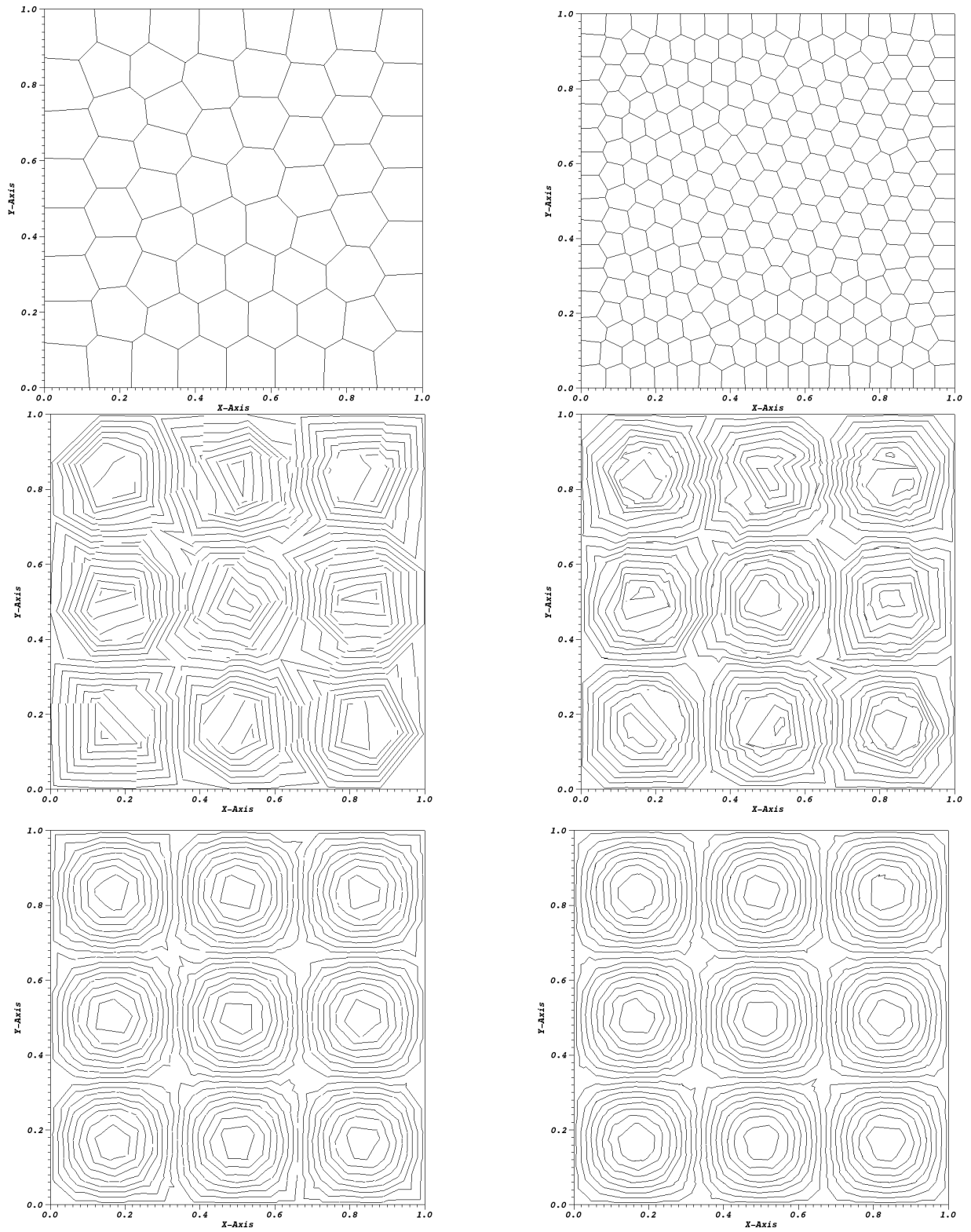where the constants in the equations are:

23

Figure 10: Examples contour plots of the sinusoid solution on polygonal meshes (top) using the linear mean value functions (middle) and the quadratic mean value functions (bottom).
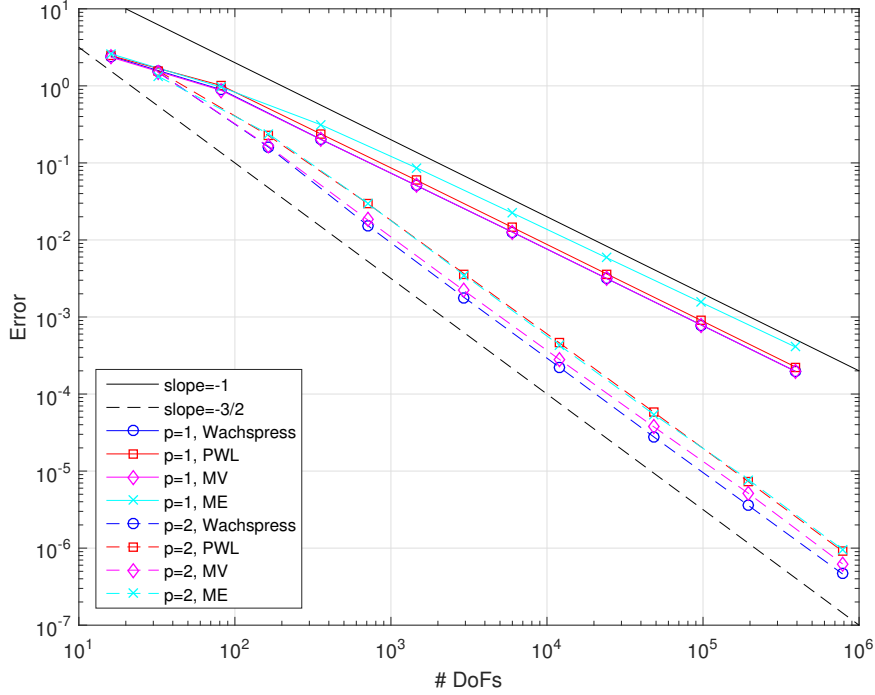
24

Figure 11: Convergence rates for the 2D Sinusoid MMS problem.

$$C_M = \frac{100}{L_x^2 L_y^2} \qquad \gamma = \frac{L_x L_y}{100}. \tag{56}$$

We again choose the dimensionality of our problem to be $[0,1]^2$ which makes $L_x = L_y = 1$. We set the local maximum to be $x_0 = y_0 = 0.75$. The domain consists of a pure absorber with $\sigma_t = 1$, and the quadrature is again the S_8 LS set.

Because we are using AMR for this Gaussian problem, we will only analyze the PWL, MV, and ME functions since the Wachspress functions cannot handle degenerate polygons. For both the linear and quadratic AMR runsets, the refinement criterion, α, was set to 0.1. Figure 12 gives the convergence history of the MMS error in the L_2-norm. We plot the different AMR cases with their corresponding uniform refinement case ($\alpha = 0$) for comparison. We can clearly see that we obtain the proper convergence rates for all cases, and the AMR sets provide better solution accuracy with fewer spatial degrees of freedom. Finally, we provide an example of a pair of AMR meshes and solutions in Figure 13 using the ME basis functions. We can see a clear difference in how the linear and quadratic basis functions will refine, with the quadratic functions causing much smoother refinement to occur.

6.4. Convergence Rate Analysis Bounded by the Solution Regularity

Our next numerical example involves studying the convergence rates of transport solutions in a purely-absorbing medium. Specifically, we seek to analyze the effects of mesh alignment along the solution discon-
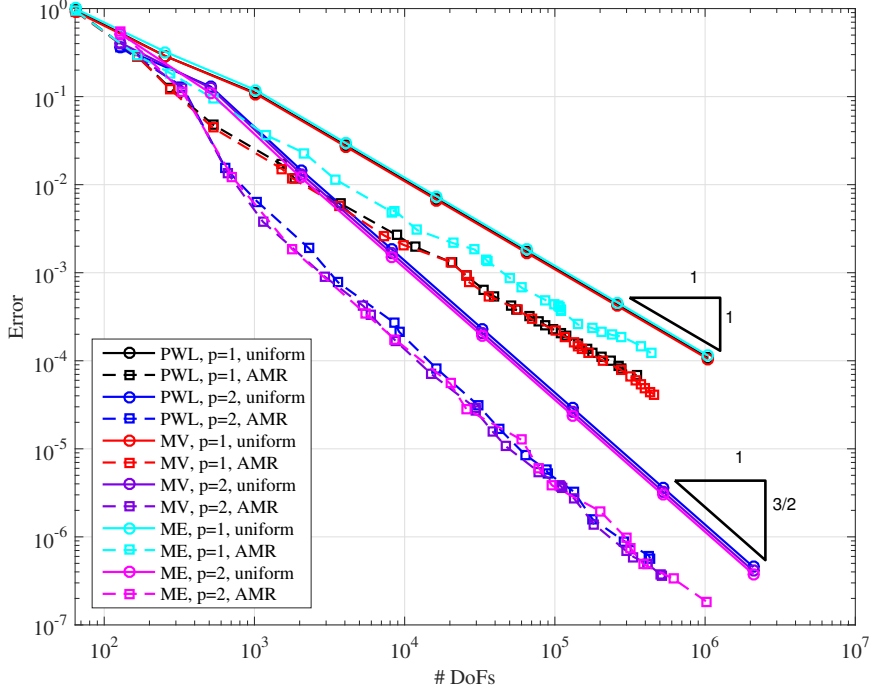
Figure 12: Convergence rates for the 2D Gaussian MMS problem.

tinuities for problems that will be constrained by the solution regularity. From Section 2, we noted that if the mesh does not align with the discontinuities of the transport solution, then the solution convergence is restricted by the regularity, r. However, if the mesh aligns with all of the solution discontinuities, then the maximum $p+1$ convergence rates can be observed.

For this example, we analyze a transport problem without scattering ($\sigma_s = 0$) and no volumetric distributed source ($q = 0$). We use a domain consisting of the unit square, $[0,1]^2$, where an incident angular flux is imparted on the left face at a downward $45°$ angle. This means that there is a solution discontinuity along the line from the point $(0,1)$ to point $(1,0)$ and the upper-right portion of the domain contains no particles. Therefore, this transport solution has a regularity of $r = 1/2$.

Two different mesh types are analyzed for this example. The first mesh type is a polygonal mesh that is again generated with the PolyMesher software. The second mesh type is a split-polygonal mesh where the mesh elements that fall on the line $(0,1)$ to $(1,0)$ are bisected. This bisection aligns with the discontinuity of the transport solution. Therefore, we expect convergence rates of $1/2$ on the polygonal mesh and $p+1$ on the split-polygonal mesh.

We first give an example solution of this problem on the two mesh types using the linear Wachspress functions in Figure 14. The solution does not "see" the transport discontinuity with the split-polygonal mesh, but the polygonal mesh contains numerical dispersion across the transport discontinuity from the
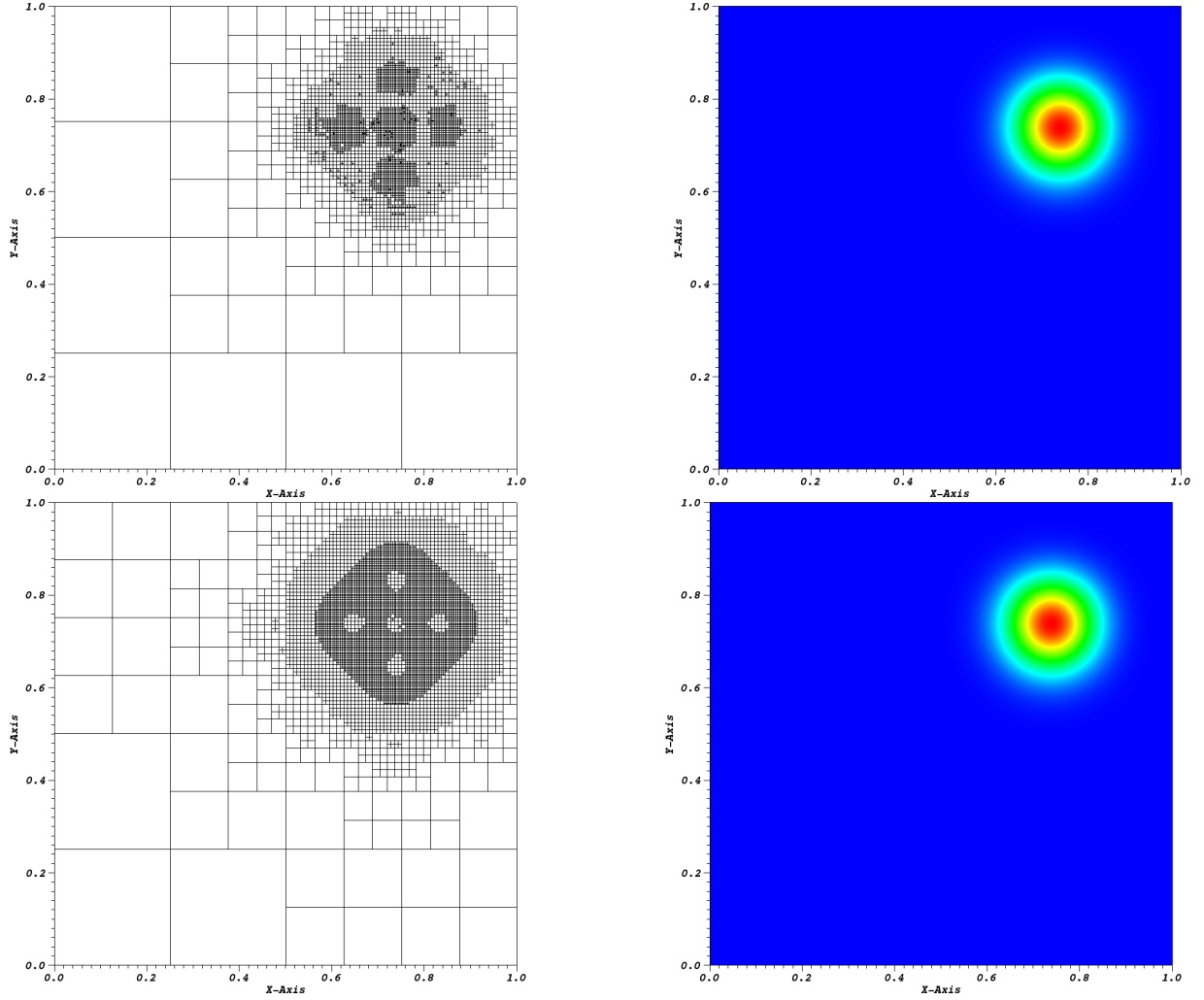
26

Figure 13: AMR meshes and solutions for the Gaussian MMS problem using the maximum entropy functions: (top) linear basis functions at cycle 15 and (bottom) quadratic serendipity basis functions at cycle 08.

non-aligned boundary outflow. The convergence rates in the L_2-norm are given in Figures 15 and 16 for the split-polygonal and polygonal meshes, respectively. With the discontinuity-aligned, split-polygonal mesh, we can clearly see that convergence rates of $p + 1$ are obtained. With the polygonal mesh, we see that convergence rates of about $1/2$ are obtained for optically thin meshes. However, for thicker domains in the pre-asymptotic range, higher convergence rates (without quite obtaining $p + 1$ values) are observed.

6.5. Thick Diffusion Limit Problem

So far, we have only presented numerical examples without scattering. Adams showed that the choice of spatial basis functions for the DGFEM transport equation is important in the thick diffusive limit [13]. Therefore, we conclude our numerical examples by demonstrating the applicability of the linear and quadratic

27
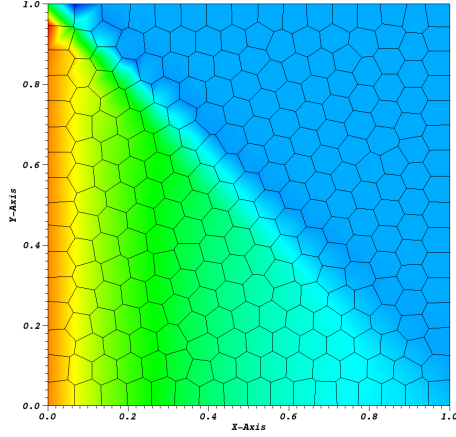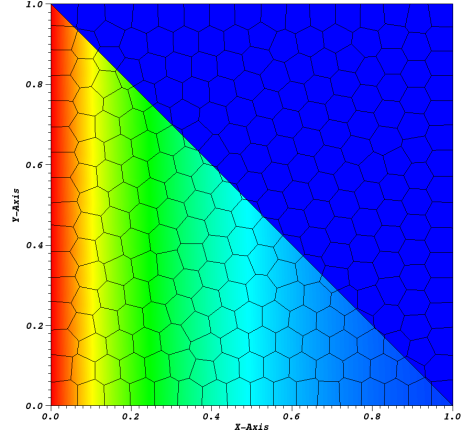
(a) Polygonal Mesh

(b) Split-Polygonal Mesh

Figure 14: Example solution of the purely-absorbing medium case with left-face incidence and $\sigma_t = 1$ using the linear Wachspress basis functions.

polygonal functions for this limit. Our problem consists of a homogeneous square domain $[0, 1]^2$ with isotropic scattering, an isotropic distributed source, and vacuum boundary conditions. The cross sections are scaled to reflect the properties of the thick diffusion limit. If we introduce a scaling parameter, ϵ, then the total cross section, absorption cross section, and distributed source scale as $1/\epsilon$, ϵ, and ϵ, respectively. Using these scaled parameters yields transport and diffusion equations of

$$\vec{\Omega} \cdot \vec{\nabla}\psi + \frac{1}{\epsilon}\psi = \left(\frac{1}{\epsilon} - \epsilon\right)\frac{\phi}{4\pi} + \frac{\epsilon}{4\pi}, \tag{57}$$

and

$$-\frac{\epsilon}{3}\nabla^2\phi + \epsilon\phi = \epsilon, \tag{58}$$

respectively. We have kept the scaling terms in Eq. (58) for completeness, even though the diffusion solution is invariant to the value of ϵ. The transport equation is discretized according to Eq. (5), while the diffusion equation is discretized using a continuous finite element (CFEM) form. The diffusion equations use a homogeneous Dirichlet boundary condition, and the S_8 quadrature is used for the transport equations.

With the chosen boundary conditions, the transport and diffusion solutions will converge only as ϵ gets small. Figure 17 provides the transport and diffusion solutions of the linear and quadratic maximum entropy functions. We can see that as ϵ goes from 10^{-1} to 10^{-5}, the transport solutions qualitatively converge to their corresponding diffusion solutions. Figure 18 then provides the convergence rates of the discretized transport, ϕ_T, and diffusion, ϕ_D, solutions under the L_2-norm ($\|\phi_T - \phi_D\|_{L_2}$). We can see that the transport solutions for the linear and quadratic Wachspress, MV, and ME functions converge to the diffusion solution at a rate
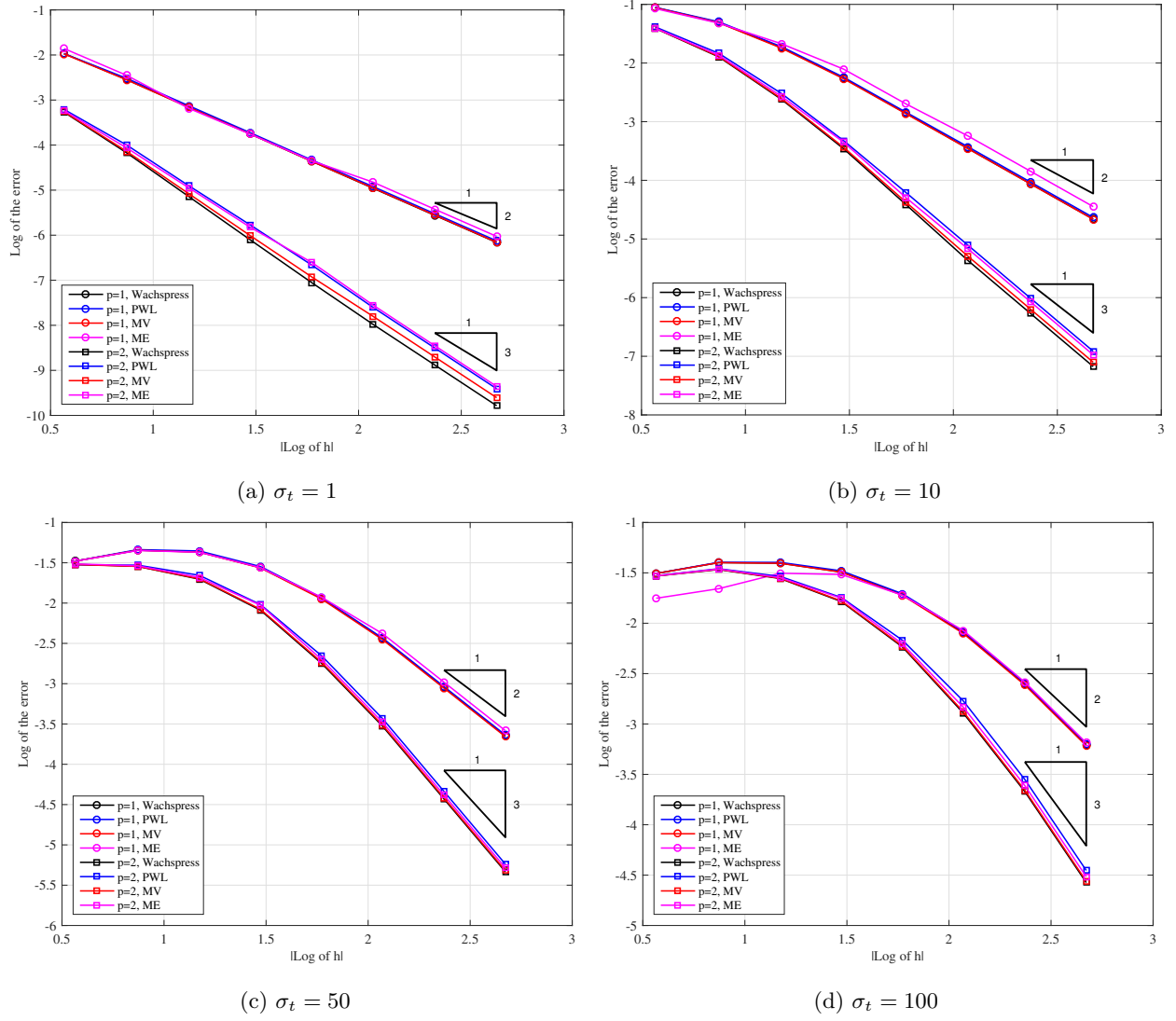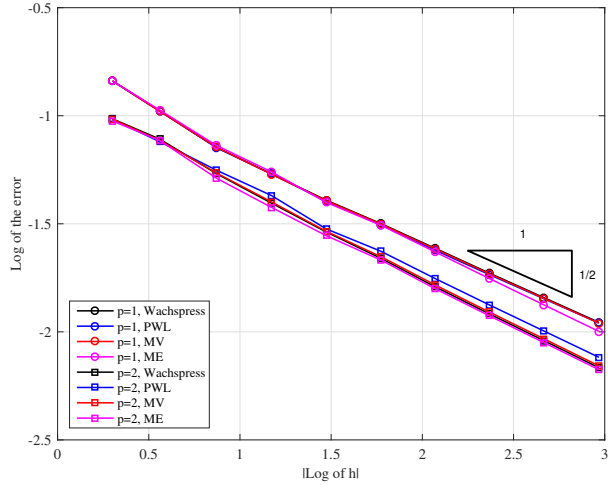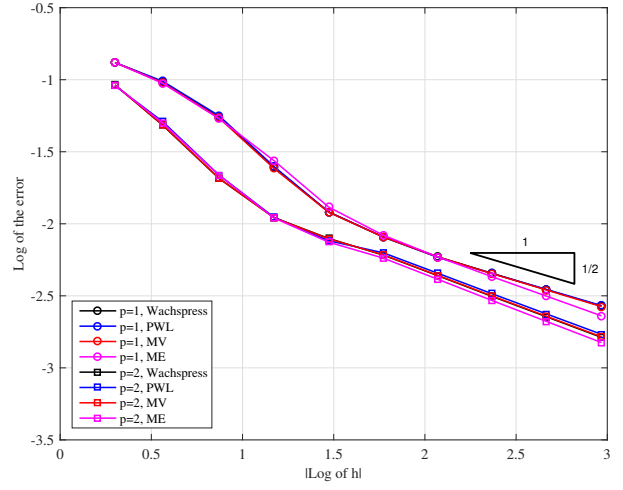
28

Figure 15: Convergence rates for the pure absorber problem with left-face incidence on split-polygonal meshes with different values of σ_t.
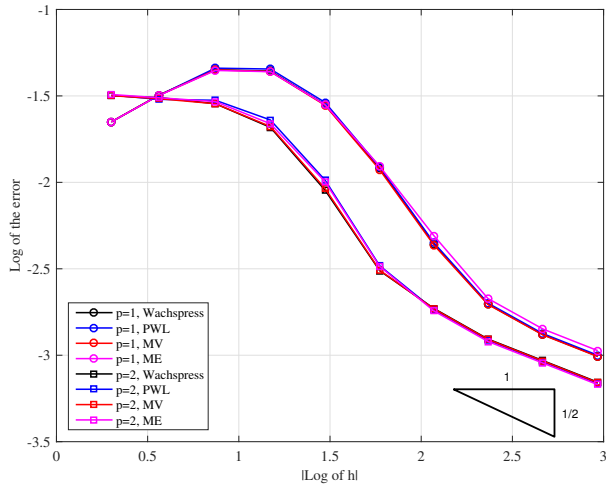
of ϵ. However, the linear and quadratic PWL functions plateau because their gradients do not live in the same functional space. Both of these behaviors are expected from [13].
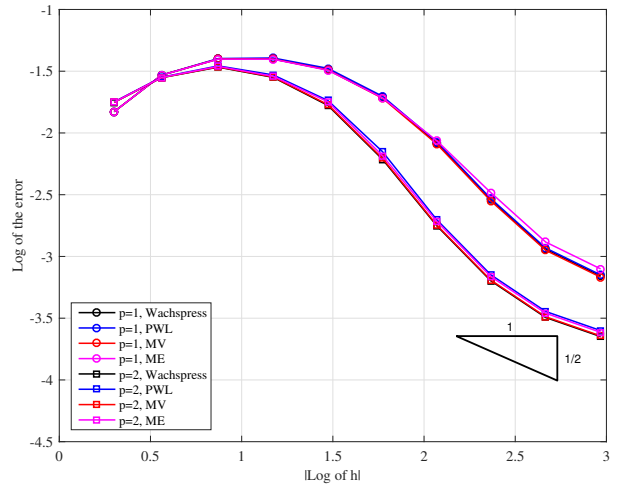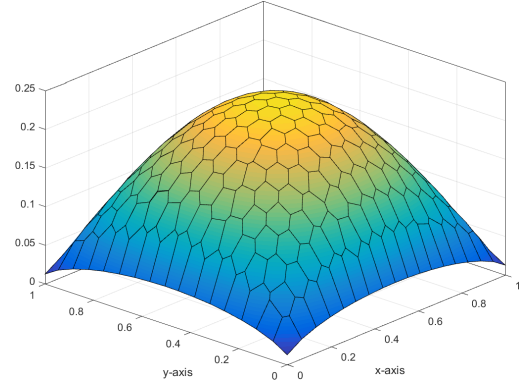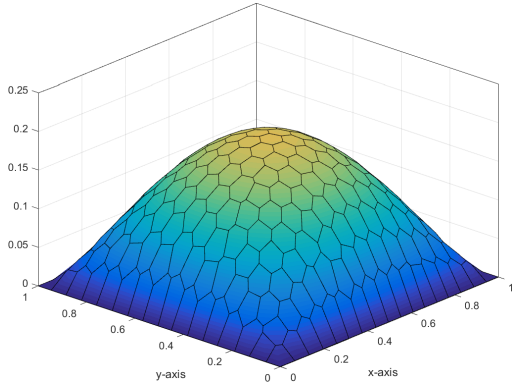
Figure 16: Convergence rates for the pure absorber problem with left-face incidence on polygonal meshes with different values of σ_t.
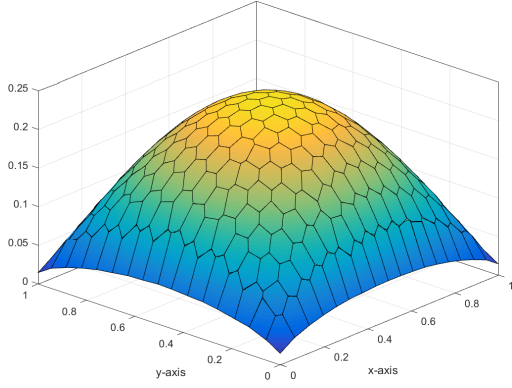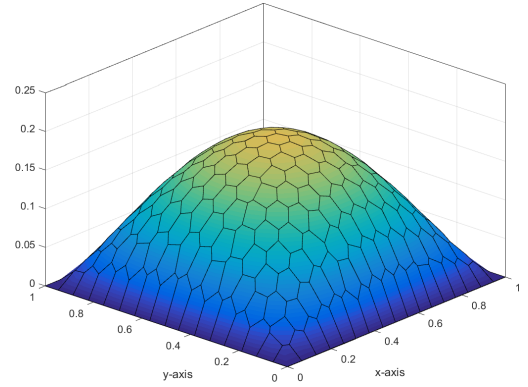
(a) Linear, $\epsilon = 10^{-1}$

(b) Quadratic, $\epsilon = 10^{-1}$

(c) Linear, $\epsilon = 10^{-5}$

(d) Quadratic, $\epsilon = 10^{-5}$
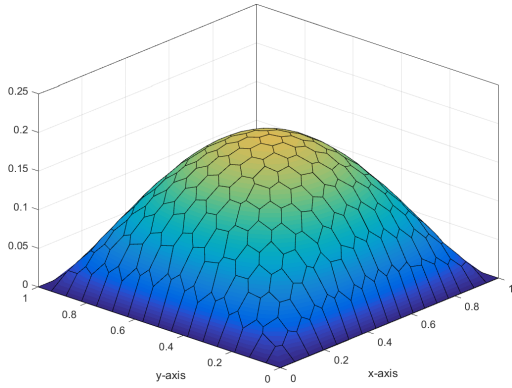
(e) Linear, Diffusion
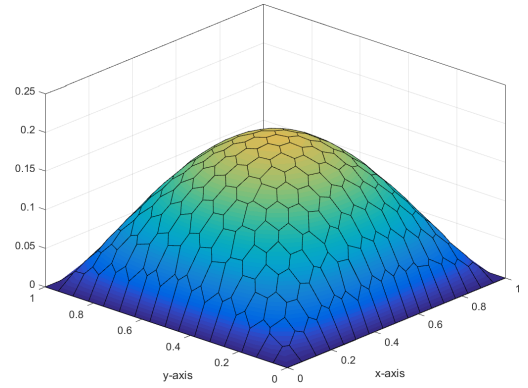
(f) Quadratic, Diffusion

Figure 17: Solutions of the thick diffusion limit problem using the maximum entropy basis functions. The top and middle rows show the linear and quadratic transport solutions for $\epsilon = 10^{-1}$ and $\epsilon = 10^{-5}$, respectively. The bottom row shows the corresponding diffusion solutions.

Figure 18: Convergence rates for the diffusion limit problem on a polygonal mesh.

7. Conclusions

A higher-order discontinuous Galerkin finite element discretization using the quadratic serendipity space of functions has been applied to S_N transport equation on arbitrary polyg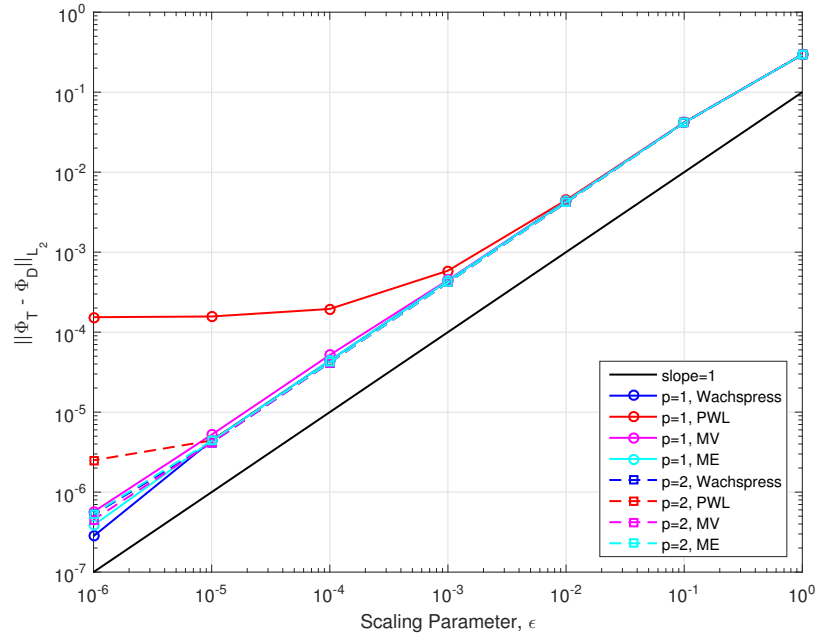onal grids. The quadratic serendipity functional space is formed by combinations of pairwise products of linear polygonal barycentric functions, of which we utilized piecewise linear functions, the Wachspress functions, mean value functions, and maximum entropy functions in this work. The quadratic serendipity functions preserve the thick diffusion limit of the S_N transport equation as predicted by the theory of [13]. Third-order convergence was numerically verified for transport problems on polygonal grids that were not bound by solution regularity. Spatial adaptive mesh refinement was also used to solve the transport equation on polygonal grids, where degenerate polygons were used in place of hanging nodes. We verified that third-order convergence was retained while achieving greater solution accuracy with fewer unknowns than grids that were uniformly refined.

This paper shows that the S_N transport equation can be effectively discretized on arbitrary polygonal meshes using a higher-order DGFEM spatial discretization. The basis functions presented here can be employed with forms of the transport equation other than the first-order S_N form; for instance, the self-adjoint angular flux equation, the least-squares equation, and the even-parity equations, as well as the P_N method in angle.

References

[1] James J. Duderstadt and William R. Martin. *Transport theory*. John Wiley & Sons, 1979.

[2] W.H. Reed and T.R. Hill. Triangular mesh methods for the neutron transport equation. *Los Alamos Report LA-UR-73-479*, 1973.

[3] P. Lesaint and Pierre-Arnaud Raviart. On a finite element method for solving the neutron transport equation. *Mathematical Aspects of Finite Elements in Partial Differential Equations*, (33):89–123, 1974.

[4] M.L. Adams and E.W. Larsen. Fast iterative methods for discrete-ordinates particle transport calculations. *Progress in Nuclear Energy*, 40(1):3–159, 2002.

[5] William D. Hawkins, Timmie Smith, Michael P. Adams, Lawrence Rauchwerger, Nancy M. Amato, and Marvin L. Adams. Efficient massively parallel transport sweeps. In *Transactions of the American Nuclear Society*, volume 107, pages 477–481, 2012.

[6] Thomas M. Evans, Alissa S. Stafford, Rachel N. Slaybaugh, and Kevin T. Clarno. Denovo: A new three-dimensional parallel discrete ordinates code in SCALE. *Nuclear Technology*, 171(2):171–200, 2010.

[7] R.E. Alcouffe, R.S. Baker, J.A. Dahl, and S.A. Turner. PARTISN code abstract. In *PHYSOR 2000 International Topical Meeting, Advances in Reactor Physics and Mathematics and Computation into the Next Millenium*, 2000.

[8] Todd A. Wareing, John M. McGhee, and Jim E. Morel. ATTILA: A three-dimensional, unstructured tetrahedral mesh discrete ordinates transport code. In *Transactions of the American Nuclear Society*, volume 75, 1996.

[9] A.M. Watson, R.E. Grove, and M.T. Shearer. Effective software design for a deterministic transport system. In *Interational Conference on Advances in Mathematics, Computational Methods & Reactor Physics (M&C 2009)*, 2009.

[10] Hiromi G. Stone and Marvin L. Adams. New spatial discretization methods for transport on unstructured grids. In *Proc. ANS Topical Meeting Mathematics and Computation, Supercomputing, Reactor Physics and Biological Applications*, 2005.

[11] Teresa S. Bailey. *The piecewise linear discontinuous finite element method applied to the RZ and XYZ transport equations.* PhD thesis, Texas A&M University, 2008.

[12] Bruno Turcksin and Jean C Ragusa. Discontinuous diffusion synthetic acceleration for s_n transport on 2d arbitrary polygonal meshes. *Journal of Computational Physics*, 274:356–369, 2014.

[13] Marvin L. Adams. Discontinuous finite element transport solutions in thick diffusive problems. *Nuclear Science & Engineering*, 137(3):298–333, 2001.

[14] Peter G. Maginot, Jean C. Ragusa, and Jim E. Morel. Discontinuous finite element discretizations for the S_N neutron transport equation in problems with spatially varying cross sections. *Annals of Nuclear Energy*, 73:506–526, 2014.

[15] Peter G. Maginot, Jean C. Ragusa, and Jim E. Morel. Lumping techniques for DFEM S_N transport in slab geometry. *Nuclear Science & Engineering*, 179(2), 2015.

[16] J.P. Hennart and E. Del Valle. A generatlized nodal finite element formalism for discrete ordinates equations in slab geometry: Part II: Theory in the discontinuous moment case. *Transport Theory and Statistical Physics*, 24:479–504, 1995.

[17] J.P. Hennart and E. Del Valle. A generatlized nodal finite element formalism for discrete ordinates equations in slab geometry: Part III: Numerical results. *Transport Theory and Statistical Physics*, 24:505–533, 1995.

[18] W.H. Reed, T.R. Hill, F.W. Brinkley, and K.D. Lathrop. TRIPLET: A two-dimensional, multigroup, triangular mesh, planar geometry, explicit transport code. Technical report, Los Alamos Scientific Lab., N. Mex.(USA), 1973.

[19] Yaqi Wang and Jean C. Ragusa. On the convergence of DGFEM applied to the discrete ordinates transport equation for structured and unstructured triangular meshes. *Nuclear Science & Engineering*, 163(1):56–72, 2009.

[20] Yaqi Wang and Jean C. Ragusa. A high-order discontinuous Galerkin method for the S_N transport equations on 2d unstructured triangular meshes. *Annals of Nuclear Energy*, 36(7):931–939, 2009.

[21] Yousry Y. Azmy. The weighted diamond-difference form of nodal transport methods. *Nuclear Science & Engineering*, 98(1):29–40, 1988.

[22] Yousry Y. Azmy. Arbitrarily high order characteristic methods for solving the neutron transport equation. *Annals of Nuclear Energy*, 19(10-12):593–606, 1992.

[23] E.L. Wachspress. High-order curved finite elements. *International Journal for Numerical Methods in Engineering*, 17(5):735–745, 1981.

[24] STAR-CCM+. http://www.cd-adapco.com. 2015.

[25] Mien Yip, Jon Mohle, and J.E. Bolander. Automated modeling of three-dimensional structural components using irregular lattices. *Computer-Aided Civil and Infrastructure Engineering*, 20(6):393–407, 2005.

[26] Alexander Rand, Andrew Gillette, and Chandrajit Bajaj. Quadratic serendipity finite elements on polygons using generalized barycentric coordinates. *Mathematics of Computation*, 83(290):2691–2716, 2014.

[27] J.E. Morel and J.M. McGhee. A self-adjoint angular flux equation. *Nuclear Science & Engineering*, 132(3):312–325, 1999.

[28] E.E. Lewis. Finite element approximation to the even-parity transport equation. In *Advances in Nuclear Science and Technology*, pages 155–225. Springer, 1981.

[29] Paul Houston, Christoph Schwab, and Endre Süli. Stabilized hp-finite element methods for first-order hyperbolic problems. *SIAM Journal on Numerical Analysis*, 37(5):1618–1643, 2000.

[30] Paul Houston, Christoph Schwab, and Endre Süli. Discontinuous hp-finite element methods for advection-diffusion-reaction problems. *SIAM Journal on Numerical Analysis*, 39(6):2133–2163, 2002.

[31] Douglas N. Arnold and Gerard Awanou. The serendipity family of finite elements. *Foundations of Computational Mathematics*, 11(3):337–344, 2011.

[32] N. Sukumar and E.A. Malsch. Recent advances in the construction of polygonal finite element interpolants. *Archives of Computational Methods in Engineering*, 13(1):129–163, 2006.

[33] Hiromi G. Stone and Marvin L. Adams. A piecewise linear finite element basis with application to particle transport. In

Proc. ANS Topical Meeting Nuclear Mathematical and Computational Sciences Meeting, 2003.

[34] E.L. Wachspress. A rational finite element basis. *Mathematics in Science and Engineering*, 1975.

[35] Michael S. Floater. Mean value coordinates. *Computer Aided Geometric Design*, 20(1):19–27, 2003.

[36] N. Sukumar. Construction of polygonal interpolants: a maximum entropy approach. *International Journal for Numerical Methods in Engineering*, 61(12):2159–2181, 2004.

[37] Claude E. Shannon. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379–423, 1948.

[38] Edwin T. Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620–630, 1957.

[39] Kai Hormann and Natarajan Sukumar. Maximum entropy coordinates for arbitrary polytopes. In *Computer Graphics Forum*, volume 27, pages 1513–1520. Wiley Online Library, 2008.

[40] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[41] David González, Elías Cueto, and Manuel Doblaré. A higher order method based on local maximum entropy approximation. *International Gournal for Numerical Methods in Engineering*, 83(6):741–764, 2010.

[42] N. Sukumar. Quadratic maximum-entropy serendipity shape functions for arbitrary planar polygons. *Computer Methods in Applied Mechanics and Engineering*, 263:27–41, 2013.

[43] Roger Penrose. A generalized inverse for matrices. In *Mathematical proceedings of the Cambridge philosophical society*, volume 51, pages 406–413. Cambridge Univ Press, 1955.

[44] Wolfgang Bangerth and Rolf Rannacher. *Adaptive finite element methods for differential equations*. Springer Basel AG, 2013.

[45] Yaqi Wang and Jean C. Ragusa. Standard and goal-oriented adaptive mesh refinement applied to radiation transport on 2D unstructured triangular meshes. *Journal of Computational Physics*, 230(3):763–788, 2011.

[46] Kambiz Salari and Patrick Knupp. Code verification by the method of manufactured solutions. Technical report, Sandia National Labs., Albuquerque, NM (US); Sandia National Labs., Livermore, CA (US), 2000.

[47] A.I. Shestakov, J.A. Harte, and D.S. Kershaw. Solution of the diffusion equation by finite elements in Lagrangian hydrodynamic codes. *Journal of Computational Physics*, 76(2):385–413, 1988.

[48] A.I. Shestakov, D.S. Kershaw, and G.B. Zimmerman. Test problems in radiative transfer calculations. *Nuclear Science & Engineering*, 105(1):88–104, 1990.

[49] Jean C. Ragusa. Discontinuous finite element solution of the radiation diffusion equation on arbitrary polygonal meshes and locally adapted quadrilateral grids. *Journal of Computational Physics*, 280:195–213, 2015.

[50] David S. Kershaw. Differencing of the diffusion equation in Lagrangian hydrodynamic codes. *Journal of Computational Physics*, 39(2):375–395, 1981.

[51] E.E. Lewis and W.F. Miller. *Computational methods of neutron transport*. John Wiley and Sons, Inc., New York, NY, 1984.

[52] Cameron Talischi, Glaucio H. Paulino, Anderson Pereira, and Ivan Menezes. PolyMesher: a general-purpose mesh generator for polygonal elements written in Matlab. *Structural and Multidisciplinary Optimization*, 45(3):309–328, 2012.

A. Limits of the Barycentric Functions on the Boundary

Each of the four basis functions presented in this paper satisfies the definitions of the general barycentric functions given in Section 3. These functions can be written for a polygonal element K with n vertices using the general form:

$$\lambda_i(\vec{r}) = \frac{w_i(\vec{r})}{\sum\limits_{j=1}^{n} w_j(\vec{r})}. \tag{59}$$

The differences between the functions lie with the definition of the weight functions, w. Unfortunately, for the Wachspress, MV, and ME functions, there is a sub-set of the element boundary, ∂K, where some of the weight functions are undefined (divide-by-zero numerical evaluations). However, these functions do have valid limits, which can be directly evaluated to compute the surface integrals in Eq. (5). For this analysis, we will evaluate the limits of the barycentric functions on face e_j connected by vertices j and $j+1$. The limits of the functions λ_j and λ_{j+1} for position $\vec{r}^* \in e_j$ are given by

$$\lim_{\vec{r} \to \vec{r}^* \in e_j} \lambda_j(\vec{r}) = \frac{w_j(\vec{r})}{w_j(\vec{r}) + w_{j+1}(\vec{r})} \tag{60}$$

and

$$\lim_{\vec{r} \to \vec{r}^* \in e_j} \lambda_{j+1}(\vec{r}) = \frac{w_{j+1}(\vec{r})}{w_j(\vec{r}) + w_{j+1}(\vec{r})}, \tag{61}$$

respectively, where the denominators contain the two terms because all the other weight functions are zero along e_j. The affine property then states the following:

$$w_j(\vec{r})\,(\vec{r}_j - \vec{r}) + w_{j+1}(\vec{r})\,(\vec{r}_{j+1} - \vec{r}) = \vec{0}. \tag{62}$$

If we take the norm of Eq. (62) and rearrange terms, then we obtain

$$\frac{w_{j+1}}{w_j} = \frac{|\vec{r}_j - \vec{r}|}{|\vec{r}_{j+1} - \vec{r}|}. \tag{63}$$

Inserting Eq. (63) into Eqs. (60 - 61) obtains the limits of the barycentric functions on face e_j:

$$\lim_{\vec{r} \to \vec{r}^* \in e_j} \lambda_i(\vec{r}) = \begin{cases} \dfrac{|\vec{r}_{j+1} - \vec{r}|}{|\vec{r}_{j+1} - \vec{r}_j|}, & i = j \\[2ex] \dfrac{|\vec{r}_j - \vec{r}|}{|\vec{r}_{j+1} - \vec{r}_j|}, & i = j+1 \\[2ex] 0, & \text{otherwise} \end{cases} \tag{64}$$

From Eq. (64), we can see that the Lagrange property and the piecewise boundary linearity property are satisfied. Using the limits of λ_j and λ_{j+1} on face e_j, we can explicitly write the surface integral for all linear functions as

$$\Big\langle \vec{n}\,b, b \Big\rangle_{e_j} = \frac{L_j \vec{n}_j}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \tag{65}$$

where $b = [b_j, b_{j+1}]$, $\vec{n}_j$ is the outward unit normal of face e_j, and L_j is the edge length of face e_j. It is assumed that $\vec{n}_j$ is constant along e_j.

36