ICES REPORT 17-10

May 2017

High-order polygonal finite elements using ultraweak formulations

by

Ali Vaziri Astaneh, Federico Fuentes, Jaime Mora, and Leszek Demkowicz

The Institute for Computational Engineering and Sciences
The University of Texas at Austin
Austin, Texas 78712

Reference: Ali Vaziri Astaneh, Federico Fuentes, Jaime Mora, and Leszek Demkowicz, "High-order polygonal finite elements using ultraweak formulations," ICES REPORT 17-10, The Institute for Computational Engineering and Sciences, The University of Texas at Austin, May 2017.

High-order polygonal finite elements using ultraweak formulations

Ali Vaziri Astaneh, Federico Fuentes, Jaime Mora, and Leszek Demkowicz

The Institute for Computational Engineering and Sciences (ICES), The University of Texas at Austin, 201 E 24th St, Austin, TX 78712, USA

Abstract

This work represents the first endeavor in using ultraweak formulations to implement high-order polygonal finite element methods via the DPG methodology. Ultraweak variational formulations are nonstandard in that all the weight of the derivatives lies in the test space, while the trial space can be chosen as copies of L^2-discretizations that have no need to be continuous along adjacent elements. Additionally, the test spaces are broken along the mesh. This allows to define a conforming finite element method by defining most spaces by restriction of a bounding triangle or box to the polygonal element. The only variables that require nontrivial compatibility across elements are the so-called interface or skeleton variables, which can be defined directly at the element boundaries that consist of only straight line segments. Unlike other polygonal high-order methods, this one does not require ad hoc stabilization terms thanks to the crafted stability resulting from the use of the DPG methodology. A proof of convergence of the form h^p is provided and corroborated through several numerical examples of Poisson's equation, including the use of highly distorted concave elements and the application-driven study of discontinuous material properties along an arbitrary interface. Lastly, the numerical method includes a natural a posteriori error estimator, so adaptivity schemes with potential use in crack propagation are investigated in the context of general polygons, and compared to standard adaptivity schemes (with hanging nodes).

1 Introduction

Numerical solutions of boundary value problems with meshes of general polytopes were first proposed by Wachspress [73], who introduced rational barycentric coordinates that formed a finite element basis over convex polygons, leading to a conforming finite element method (FEM) with new types of elements. Over the last two decades, there has been growing collection of numerical methods using general polytopes which extend well beyond the original ideas of Wachspress. Among the reasons for this group of methods to thrive is a handful of advantages that polytopes offer over traditionally shaped elements (simplices, hexahedra, etc.) such as the ability of matching complex interfaces (see for example [60, 22]); the flexibility to mesh arbitrary geometries in a simple way (a single algorithm to mesh) and their possible role of transition elements [69]; the possibility to avoid the limitations of parametric elements for highly distorted or ill-shaped physical elements (see examples of very unconventional element shapes on [24]); the potential of conformingly handling multiple hanging nodes in local h-refinements [70]; and the possibility to allow greater deformations, so there is less tendency to mesh-locking than triangles or quadrilaterals in incompressible media [25].

All of the features mentioned above give polytopal finite element methods a wide range of applicability, especially where conventional element methods do not fare well. For example, in the deformation of materials with heterogeneous microstructure [46], in the mechanics of complex materials like elastomers and

biomaterials [25, 29], in problems where interface fitting is required [22], as well as in modeling fractured media [10]. Promising results also have been obtained in crack propagation modeling [67] and in topology optimization with unstructured polygonal meshes [71, 45], since these are less biased than structured or unstructured meshes of simplices.

Before proceeding with the ideas surrounding our method, we first give a brief overview of the major polytopal element methods in the literature. Many methods still utilize different types of generalized barycentric coordinates, which have proliferated since Wachspress originally introduced them, and these include extensions to nonconvex polytopes. Other choices of shape functions are also possible [11, 12, 13]. All these methods fall under the rubric of polygonal FEM (PFEM), which are usually H^1-conforming Galerkin FEM [69], but there are extensions to mixed methods too [25]. They have the advantage of allowing very flexible adaptivity schemes, and may admit hanging nodes [70]. Most of these methods are limited by first order h-convergence, but for convex polytopes some families of higher-order shape functions have been proposed too [65, 48]. Another difficulty is the choice of quadrature schemes used to integrate, since the barycentric coordinates are in general rational polynomials [59, 26]. For more details of the historical development of the field, we refer to a thorough review [57].

Mimetic finite difference (MFD) methods are based on another discretization technique, which designs discrete differential operators such that fundamental vector calculus identities and physical laws can be reproduced in a discrete context [55, 16, 15]. The ideas of MFD led to the development of the Virtual Element Method (VEM) [6], which has enjoyed great acceptance. In this method, appropriate spaces are tailored for each polytopal element, such that their functions have continuous and piecewise polynomial traces over the boundaries, and that the integrals over the cells can be computed exactly (i.e. up to machine precision) with quadrature points on the boundary only [57]. The power of VEM lies partly in eliminating the need of explicitly constructing the shape functions in the element, and yet resulting in a FEM-like variational setting [9]. It is also a high-order method [5], and recent work has resulted in the construction of $\boldsymbol{H}(\mathrm{div})$- and $\boldsymbol{H}(\mathrm{curl})$-conforming spaces [8]. VEM has been used for different physical problems like linear elasticity, plate bending and general second-order elliptic PDEs [7, 17, 9]. The downsides are that a stability operator needs to be added to guarantee convergence [57], and that the solution at interior points of the elements cannot be accessed directly, but instead needs to be approximated [9].

Another high-order numerical method for general polytopes is the interior penalty hp discontinuous Galerkin FEM (IP DGFEM) [18]. This is a nonconforming high-order method which uses restrictions of standard FE spaces associated to a bounding box of each element. Due its nonconformity, the method has a thorough, but nonstandard error analysis (which depends on the equation), and it also has the need of adding extra terms to ensure stability. Lastly, there are yet more methods in the literature, such as hybrid mimetic mixed methods [39], PFEM-VEM [57] and the weak Galerkin (WG) method [60, 61].

The purpose of this article is to add a completely new family of polygonal finite element methods to the current list. It stands apart in that it arises from very different ideas to the existing methods, and as such, it carries its own advantages and disadvantages which can make it competitive to other methods. It is technically a conforming method, yet it allows arbitrary high-order polynomials in its discretization due to the variational structure. Indeed, the key to this method is the use of "broken" ultraweak variational formulations, which have many unique properties. Most remarkable are that its test space is broken (i.e. it may be discontinuous at element interfaces), and that a large part of its trial space is in L^2, so that continuity across elements is not required. In fact, the only communication between elements happens through so-called

skeleton (or interface) variables that only live in the boundaries of the elements. In 2D, this implies that the variables live in line segments making up the edges of the elements, so defining these spaces is not complicated at all. In 3D more difficulties arise, so for the purpose of this article we will restrict ourselves to general polygons in 2D.

Ultraweak formulations have been a subject of intense research in the last decade, especially because of their tight connection with discontinuous Petrov-Galerkin (DPG) methods [33]. Indeed, we will use the DPG methodology to solve the variational formulation, in large part because there is no other systematic alternative to solve these nonstandard variational formulations having different trial and test spaces. But this is not a hindrance at all. To the contrary, it is very beneficial for the numerical method itself and provides a systematic avenue to tackle very large families of equations, which other methods have difficulties with. Indeed, the DPG methodology, which always produces a positive-definite FE stiffness matrix, is essentially crafted to produce stability, and requires no additional stabilization terms to accomplish this goal. It is also posed in extremely general terms, thus making it variationally versatile, and allowing it to solve nonstandard formulations, such as ultraweak. This is why using this methodology it is possible to solve a wide variety of equations, especially those involving numerical stability issues [28, 37, 21, 62], and even the same equations but with very different variational formulations [53]. Some applications where it has been used are wave propagation [74, 49, 34, 64], transmission problems [51, 44], electromagnetism [19], elasticity [53, 14, 42, 41], fluid flow [66, 20, 40, 54] and optical fibers via Schrödinger's equation [35].

The advantages the our method will have are the ease with which it generalizes to other equations; the solid mathematical background that accompanies the DPG methodology in terms of proving stability and convergence of the resulting numerical methods; that it is a high-order method; that it allows for discontinuous material properties while retaining stability; and that it carries a completely natural a posteriori error estimator (coming from the DPG methodology) for arbitrary polynomial order p and for any equation being solved. This last point has major implications, as it allows to easily implement adaptive refinement strategies, which are particularly convenient in the context of general polygons, due to their mere flexibility (hanging nodes do not pose any resistance) and their applicability in practical problems related to fracture [67, 56]. The disadvantages are possibly the ease of implementation for an unexperienced programmer, and the computational cost.

The overview of the article is as follows. In Section 2 we describe polygonal finite element methods using ultraweak formulations for a model problem (Poisson's equation), along with how to solve it via the DPG methodology and appropriate discretizations. The proof of convergence is also included. In Section 3 several illustrative examples are presented. High-order convergence for different p is verified for both convex and highly distorted concave elements. Then, a physically relevant problem involving discontinuous material properties along an arbitrary interface is solved. Finally, an adaptivity strategy is described, successfully implemented, and compared to more traditional adaptive strategies making use of hanging nodes. Our concluding remarks are presented in Section 4.

2 Numerical method

Typical finite element methods map elements in the actual physical space to a known fixed master element space corresponding to the same element type. For example, in 2D a general quadrilateral in $\mathbb{R}^2$ is mapped to a master quadrilateral (typically $(0,1)^2$ or $(-1,1)^2$). This requires to have a master element for each

element type, which is possible if the number element types is limited (e.g. to quadrilaterals and triangles in 2D, or to hexahedra, tetrahedra, triangular prisms and pyramids in 3D), but is usually nonviable when dealing with general polytopes. Thus, as with any polytopal finite element method, the idea is to circumvent any master elements and shift the focus directly to the physical space.

The main issue becomes that of satisfying inter-element continuity of the basis functions, which is required when discretizing Sobolev spaces such as H^1. This is partly resolved by using generalized barycentric coordinates, but these techniques are usually limited to first order methods (in terms of convergence), and it becomes difficult to discretize other Sobolev spaces, such as $\boldsymbol{H}(\mathrm{curl})$ and $\boldsymbol{H}(\mathrm{div})$, including the lowest order cases [23]. Indeed, even with the "traditional" pyramid element, having high-order discretizations for different spaces is challenging to achieve [63, 43, 47, 1], and so is the case for 2D non-affine quadrilaterals [2]. To overcome this, VEMs concentrate on the boundaries while interior penalty discontinuous Galerkin methods remove the continuity requirements altogether. However, both methods need to carefully add (equation-dependent) stabilization or penalty terms, and they must account for these in the error analysis, which becomes nonstandard when compared to the classical theory of convergence [27].

Meanwhile, as will be seen, the discontinuous Petrov-Galerkin (DPG) methodology, enjoys a "variational" versatility which allows it to distance itself from traditional variational formulations, like primal and mixed formulations. In particular, it is compatible with ultraweak variational formulations, where the trial variables do not need to be differentiated (thus lying in copies of L^2) and the test variables have all the weight of the derivatives. Nevertheless, as will be observed later, the practicality of the method relies on being able to break the test spaces (but not the trial spaces), which transforms the variational formulation into a "broken" variational formulation with additional skeleton (or interface) unknowns, and which shares the same stability properties as the original formulation. This implies that with the resulting DPG method based on the broken ultraweak formulation, all trial variables taking values in a domain Ω (which lie in L^2) and all test variables (which belong to a mesh-broken space) need not be continuous across elements. This gives very high flexibility with respect to their discretization. The continuity requirements are met by the skeleton variables which live only in the boundaries of the elements in the mesh. Technically speaking the resulting method is still a conforming finite element method, and the usual error analysis can be applied.

For polygons in 2D the boundaries are simply line segments, so it is easy to define high-order discretizations along the boundaries. This is less trivial for polyhedra in 3D, and for this reason, as warned previously, the computations and analysis will be restricted to 2D throughout this work. Before proceeding further we introduce a model problem and its associated ultraweak formulations.

2.1 Model problem and ultraweak variational formulations

As a model problem, consider Poisson's equation coming from the steady-state heat equation in a domain $\Omega \subseteq \mathbb{R}^2$, where u is the temperature, $\boldsymbol{q}$ is the heat flow, $k > 0$ is the (possibly heterogeneous) thermal conductivity, and r is the internal heat source,

$$-\operatorname{div}(k\nabla u) = r\,, \qquad \Leftrightarrow \qquad \begin{cases} \operatorname{div}\boldsymbol{q} = r\,, \\ \boldsymbol{q} + k\nabla u = 0\,. \end{cases} \tag{2.1}$$

Note the equation can be written directly as a second order system (left) or as a first order system (right). For simplicity, assume temperature boundary conditions along all of $\partial\Omega$, so that $u = g$ at $\partial\Omega$, where g is a known function.

To solve the equation using a finite element method, it is necessary to write it in variational form, and in this respect, there are many possibilities. For now assume vanishing temperature boundary conditions so that $g = 0$. The classical approach stems directly from the second order equation by multiplying by a test function and integrating by parts once, leading to the primal formulation where the solution u is sought in the trial space $\mathcal{U}^{\mathcal{P}}$ and must satisfy that

$$\begin{aligned} b^{\mathcal{P}}(u,v) &= \ell^{\mathcal{P}}(v) \qquad \forall v \in \mathcal{V}^{\mathcal{P}} = \mathcal{U}^{\mathcal{P}} = H^1_0(\Omega)\,, \\ b^{\mathcal{P}}(u,v) &= (k\nabla u, \nabla v)_\Omega\,, \qquad\qquad \ell^{\mathcal{P}}(v) = (r,v)_\Omega\,, \end{aligned} \tag{2.2}$$

with $(u,v)_K = \int_K u \cdot v \,\mathrm{d}\, K$ for $K \subseteq \Omega$. Notice in this case $\mathcal{V}^{\mathcal{P}} = \mathcal{U}^{\mathcal{P}}$, so both spaces can be discretized in the same way, leading to the Galerkin method. The same property holds for standard mixed formulations which stem from the first order system. The ultraweak formulation is also derived from the first order system, but the difference is that all equations are integrated by parts to pass the derivatives to the test functions. The resulting ultraweak formulation seeks $(u, \boldsymbol{q}) = \mathfrak{u}_0 \in \mathcal{U}_0 = L^2(\Omega) \times \boldsymbol{L}^2(\Omega)$ satisfying

$$\begin{aligned} b_0(\mathfrak{u}_0, \mathfrak{v}) &= \ell(\mathfrak{v}_0) \qquad \forall (v,\boldsymbol{\tau}) = \mathfrak{v}_0 \in \mathcal{V}_0 = H^1_0(\Omega) \times \boldsymbol{H}(\mathrm{div},\Omega)\,, \\ b_0\big((u,\boldsymbol{q}),(v,\boldsymbol{\tau})\big) &= -(\boldsymbol{q},\nabla v)_\Omega + (\tfrac{1}{k}\boldsymbol{q},\boldsymbol{\tau})_\Omega - (u,\mathrm{div}\,\boldsymbol{\tau})_\Omega, \qquad \ell\big((v,\boldsymbol{\tau})\big) = (r,v)_\Omega\,, \end{aligned} \tag{2.3}$$

where $\boldsymbol{L}^2(\Omega) = (L^2(\Omega))^2$. Clearly the trial and test spaces in this case are completely different, $\mathcal{U}_0 \neq \mathcal{V}_0$. Thus, to solve this system it is necessary to drift away from the traditional Galerkin method. As we will see, a discretization via minimum residual finite element methods is a viable option. In the meantime, it is worth remarking that the primal and ultraweak formulations are mutually well-posed in the infinite-dimensional setting [53, 32, 19], meaning the existence of a unique solution in the trial space satisfying a stability estimate is guaranteed. That is, the ultraweak formulation is well-posed (since the primal formulation is well-posed in view of the Lax-Milgram theorem and Poincaré's inequality).

The ultraweak formulation has copies of $L^2(\Omega)$ as a trial space, which means its discretization need not satisfy any inter-element continuity requirements, and this is very convenient in the case of general polygons. However, all the difficulties are passed to the test space for which inter-element continuity requirements seem to be essential. Fortunately, it is possible to remove these requirements in the test space as well, but at a cost. In fact, the practicality of DPG methods rely on using broken (or discontinuous) test spaces. This will result in a slightly modified formulation called the *broken* ultraweak formulation. To derive it, consider a mesh (i.e. an open partition), $\mathcal{T}$, of Ω comprised of (disjoint) elements $K \in \mathcal{T}$, and define the broken spaces and piecewise integration,

$$\begin{aligned} H^1(\mathcal{T}) &= \{v \in L^2(\Omega) \mid v|_K \in H^1(K)\,, \forall K \in \mathcal{T}\}\,, \\ \boldsymbol{H}(\mathrm{div},\mathcal{T}) &= \{\boldsymbol{\tau} \in \boldsymbol{L}^2(\Omega) \mid \boldsymbol{\tau}|_K \in \boldsymbol{H}(\mathrm{div},K)\,, \forall K \in \mathcal{T}\}\,, \\ (u,v)_{\mathcal{T}} &= \sum_{K\in\mathcal{T}} (u|_K, v|_K)_K\,. \end{aligned} \tag{2.4}$$

Then, *element-wise*, multiply by broken test functions $(v,\boldsymbol{\tau}) = \mathfrak{v} \in \mathcal{V} = H^1(\mathcal{T}) \times \boldsymbol{H}(\mathrm{div},\mathcal{T})$, integrate by parts, and sum across all elements. The result is very similar to the ultraweak formulation, but has new terms in the boundaries of the elements involving $u|_{\partial K}$ and $\boldsymbol{q}|_{\partial K}\cdot\hat{\mathbf{n}}_K$, where $\hat{\mathbf{n}}_K$ is the outward normal to the element K. These terms vanish if the test space is not broken (i.e. $\mathcal{V}_0$). Unfortunately, if we want $u \in L^2(\Omega)$ and $\boldsymbol{q} \in \boldsymbol{L}^2(\Omega)$, then the traces $u|_{\partial K}$ and $\boldsymbol{q}|_{\partial K}\cdot\hat{\mathbf{n}}_K$ technically do not exist and to incorporate them it is

necessary to add new skeleton (or interface) variables in the spaces

$$\begin{aligned} H_0^{1/2}(\partial\mathcal{T}) &= \big\{\hat{u} \mid \exists u \in H_0^1(\Omega),\ \hat{u} = \textstyle\prod_{K\in\mathcal{T}} (u|_K)\big|_{\partial K}\big\}\,, \\ H^{-1/2}(\partial\mathcal{T}) &= \big\{\hat{q}_{\hat{\mathbf{n}}} \mid \exists \boldsymbol{q} \in \boldsymbol{H}(\mathrm{div},\Omega),\ \hat{q}_{\hat{\mathbf{n}}} = \textstyle\prod_{K\in\mathcal{T}} (\boldsymbol{q}|_K)\big|_{\partial K}\cdot\hat{\mathbf{n}}_K\big\}\,, \\ \langle \hat{u}, \hat{v}\rangle_{\partial\mathcal{T}} &= \sum_{K\in\mathcal{T}} \langle (\hat{u})_K, (\hat{v})_K\rangle_{\partial K}\,, \end{aligned} \tag{2.5}$$

where the duality $\langle\cdot,\cdot\rangle_{\partial K}$ can be thought of as a boundary integral (it is a boundary integral for smooth enough inputs). Therefore, the resulting broken ultraweak variational formulation seeks

$$\begin{aligned} (\mathfrak{u}_0, \hat{\mathfrak{u}}) = \mathfrak{u} \in \mathcal{U} = \mathcal{U}_0 \times \hat{\mathcal{U}}\,, \\ (u, \boldsymbol{q}) = \mathfrak{u}_0 \in \mathcal{U}_0 = L^2(\Omega)\times \boldsymbol{L}^2(\Omega)\,, \qquad (\hat{u}, \hat{q}_{\hat{\mathbf{n}}}) = \hat{\mathfrak{u}} \in \hat{\mathcal{U}} = H_0^{1/2}(\partial\mathcal{T}) \times H^{-1/2}(\partial\mathcal{T})\,, \end{aligned} \tag{2.6}$$

such that

$$\begin{aligned} &b(\mathfrak{u},\mathfrak{v}) = \ell(\mathfrak{v}) \qquad \forall (v,\boldsymbol{\tau}) = \mathfrak{v} \in \mathcal{V} = H^1(\mathcal{T}) \times \boldsymbol{H}(\mathrm{div},\mathcal{T})\,, \\ &b\big((\mathfrak{u}_0,\hat{\mathfrak{u}}),\mathfrak{v}\big) = b_0(\mathfrak{u}_0,\mathfrak{v}) + \hat{b}(\hat{\mathfrak{u}},\mathfrak{v})\,, \qquad \ell\big((v,\boldsymbol{\tau})\big) = (r,v)_{\mathcal{T}}\,, \\ &b_0\big((u,\boldsymbol{q}),(v,\boldsymbol{\tau})\big) = -(\boldsymbol{q},\nabla v)_{\mathcal{T}} + (\tfrac{1}{k}\boldsymbol{q},\boldsymbol{\tau})_{\mathcal{T}} - (u,\mathrm{div}\,\boldsymbol{\tau})_{\mathcal{T}}\,, \\ &\hat{b}\big((\hat{u},\hat{q}_{\hat{\mathbf{n}}}),(v,\boldsymbol{\tau})\big) = \langle \hat{q}_{\hat{\mathbf{n}}}, v_{\partial\mathcal{T}}\rangle_{\partial\mathcal{T}} + \langle \hat{u}, \boldsymbol{\tau}_{\partial\mathcal{T}}\rangle_{\partial\mathcal{T}}\,, \end{aligned} \tag{2.7}$$

where $v_{\partial\mathcal{T}} = \prod_{K\in\mathcal{T}} (v|_K)\big|_{\partial K}$ and $\boldsymbol{\tau}_{\partial\mathcal{T}} = \prod_{K\in\mathcal{T}} (\boldsymbol{\tau}|_K)\big|_{\partial K}\cdot\hat{\mathbf{n}}_K$. This formulation can also be proved to be well-posed with stability properties independent of the choice of the mesh [19, 53]. With nontrivial boundary conditions, $g \neq 0$, simply consider $\ell(\mathfrak{v}) = (r,v)_{\mathcal{T}} - \langle \widetilde{g}_{\partial\mathcal{T}}, \boldsymbol{\tau}_{\partial\mathcal{T}}\rangle_{\partial\mathcal{T}}$ instead, where $\widetilde{g} \in H^1(\Omega)$ is an extension of $g \in H^{1/2}(\partial\Omega) = \{h = \widetilde{h}|_{\partial\Omega} \mid \widetilde{h} \in H^1(\Omega)\}$, and add $\widetilde{g}$ to the solution u of (2.7) to obtain the final temperature.

Despite looking intricate, the broken ultraweak variational formulation has the advantage of removing much of the inter-element compatibility conditions, since some of its trial variables are in L^2 and its test variables are discontinuous along the elements. The only inter-element compatibility is due to the skeleton variables which reside solely in the boundaries of the elements. In 2D, as we mentioned before, this is extremely convenient since the boundaries are simply 1D line segments.

2.2 Discretization and the DPG methodology

The question is: how do we discretize ultraweak formulations into a linear system that we can solve computationally? For more standard formulations, the usual approach is to use the Galerkin method, which relies on having the same test and trial spaces, and therefore leads to a square linear system. Indeed, consider the primal formulation in (2.2), with $\{\mathfrak{u}_j^{\mathcal{P}}\}_{j=1}^N$ being a basis for the discrete subspaces $\mathcal{U}_h^{\mathcal{P}} = \mathcal{V}_h^{\mathcal{P}} \subseteq \mathcal{U}^{\mathcal{P}} = \mathcal{V}^{\mathcal{P}}$. Then, the discrete solution $u_h = \sum_{j=1}^N (\mathsf{u}_h)_j \mathfrak{u}_j^{\mathcal{P}} \in \mathcal{U}_h^{\mathcal{P}}$ for $\mathsf{u}_h \in \mathbb{R}^N$, satisfies

$$\mathsf{B}^{\mathcal{P}}\mathsf{u}_h = \mathsf{l}^{\mathcal{P}}\,, \tag{2.8}$$

where $\mathsf{B}_{ij}^{\mathcal{P}} = b^{\mathcal{P}}(\mathfrak{u}_j^{\mathcal{P}}, \mathfrak{v}_i^{\mathcal{P}})$ and $\mathsf{l}_i^{\mathcal{P}} = \ell^{\mathcal{P}}(\mathfrak{v}_i^{\mathcal{P}})$ with $\mathfrak{v}_i^{\mathcal{P}} = \mathfrak{u}_i^{\mathcal{P}}$, so that $\mathsf{B}^{\mathcal{P}} \in \mathbb{R}^{N\times N}$ and $\mathsf{l}^{\mathcal{P}} \in \mathbb{R}^N$. The computational practicality of this method relies in the fact that the basis functions, $\mathfrak{u}_j^{\mathcal{P}}$, are chosen with a very small support not exceeding a few neighboring elements, and this results in a desirable sparse structure of $\mathsf{B}^{\mathcal{P}}$.

In general, when the trial and test spaces are different, $\mathcal{U} \neq \mathcal{V}$, this approach is technically still possible but requires of finding bases $\{\mathfrak{u}_j\}_{j=1}^N$ and $\{\mathfrak{v}_i\}_{i=1}^N$ for $\mathcal{U}_h \subseteq \mathcal{U}$ and $\mathcal{V}_h \subseteq \mathcal{V}$ respectively. However, two issues immediately arise. First, the canonical polynomial-based discrete basis of $\mathcal{V}_h \subseteq \mathcal{V}$ typically will not be of

size N (the same size of the basis for $\mathcal{U}_h$). Second, even if a nonstandard basis for $\mathcal{V}_h$ of the right size is found, the resulting numerical method could well be unstable, meaning that the inf-sup inequality,

$$\inf_{\delta\mathfrak{u}_h\in\mathcal{U}_h}\sup_{\mathfrak{v}_h\in\mathcal{V}_h\setminus\{0\}}\frac{b(\delta\mathfrak{u}_h,\mathfrak{v}_h)}{\|\delta\mathfrak{u}_h\|_{\mathcal{U}}\|\mathfrak{v}_h\|_{\mathcal{V}}}=\gamma_h>0\,, \tag{2.9}$$

might *not* hold. In fact, depending on the equation and mesh size, even the Galerkin method will be unstable. Minimum residual finite element methods overcome these two difficulties by design.

Let $\mathcal{U}'$ and $\mathcal{V}'$ be the dual spaces to $\mathcal{U}$ and $\mathcal{V}$ respectively, and define $\mathcal{B}:\mathcal{U}\to\mathcal{V}'$ and its adjoint $\mathcal{B}':\mathcal{V}\to\mathcal{U}'$ through duality pairings as $\langle\mathcal{B}\mathfrak{u},\mathfrak{v}\rangle=b(\mathfrak{u},\mathfrak{v})=\langle\mathfrak{u},\mathcal{B}'\mathfrak{v}\rangle$. Then, for a discrete trial space $\mathcal{U}_h\subseteq\mathcal{U}$, minimum residual methods seek the minimizer of the residual [33, 53],

$$\mathfrak{u}_h^{\text{opt}}=\underset{\delta\mathfrak{u}_h\in\mathcal{U}_h}{\arg\min}\|\mathcal{B}\delta\mathfrak{u}_h-\ell\|_{\mathcal{V}'},\quad\Leftrightarrow\quad b(\mathfrak{u}_h^{\text{opt}},\mathfrak{v}^{\text{opt}})=\ell(\mathfrak{v}^{\text{opt}})\quad\forall\mathfrak{v}^{\text{opt}}\in\mathcal{V}^{\text{opt}}=\mathcal{R}_{\mathcal{V}}^{-1}\mathcal{B}\mathcal{U}_h\,, \tag{2.10}$$

where $\mathcal{R}_{\mathcal{V}}:\mathcal{V}\to\mathcal{V}'$ is the Riesz map which is defined by duality as $\langle\mathcal{R}_{\mathcal{V}}\mathfrak{v},\delta\mathfrak{v}\rangle=(\mathfrak{v},\delta\mathfrak{v})_{\mathcal{V}}$, with $(\cdot,\cdot)_{\mathcal{V}}$ being the inner product of the Hilbert space $\mathcal{V}$. Here, $\mathcal{V}^{\text{opt}}=\mathcal{R}_{\mathcal{V}}^{-1}\mathcal{B}\mathcal{U}_h$ is called the optimal test space, because this exact choice of discrete test space automatically results in the best inf-sup stable discrete method satisfying (2.9) [33]. Given an element of the basis for $\mathcal{U}_h$, $\mathfrak{u}_i\in\{\mathfrak{u}_j\}_{j=1}^N$, the corresponding optimal test function is $\mathfrak{v}_i^{\text{opt}}=\mathcal{R}_{\mathcal{V}}^{-1}\mathcal{B}\mathfrak{u}_i$, and with these choices the resulting matrix $\mathsf{B}_{ij}^{\text{opt}}=b(\mathfrak{u}_j,\mathfrak{v}_i^{\text{opt}})$ is called the optimal stiffness matrix, which is always symmetric positive-definite. Unfortunately, computing $\mathcal{R}_{\mathcal{V}}^{-1}$ is impossible because $\mathcal{V}$ is infinite-dimensional. Thus, minimum residual methods simply make a choice of an *enriched* test space $\mathcal{V}_r\subseteq\mathcal{V}$ (with $M=\dim(\mathcal{V}_r)\geq\dim(\mathcal{U}_h)=N$) over which the operator is inverted. The advantage is that this enriched space may be discretized with a standard canonical polynomial-based basis, $\{\mathfrak{v}_i\}_{i=1}^M$, and ultimately the resulting *near*-optimal space is $\mathcal{V}_h=\mathcal{V}^{\text{n-opt}}=\mathcal{R}_{\mathcal{V}_r}^{-1}\mathcal{B}\mathcal{U}_h$ and its corresponding *near*-optimal basis is $\mathfrak{v}_i^{\text{n-opt}}=\mathcal{R}_{\mathcal{V}_r}^{-1}\mathcal{B}\mathfrak{u}_i$ for every $\mathfrak{u}_i\in\{\mathfrak{u}_j\}_{j=1}^N$. The resulting discrete method can be shown to be equivalent to the linear system,

$$\mathsf{B}^{\text{n-opt}}\mathsf{u}_h=\mathsf{B}^{\mathsf{T}}\mathsf{G}^{-1}\mathsf{B}\mathsf{u}_h=\mathsf{B}^{\mathsf{T}}\mathsf{G}^{-1}\mathsf{l}=\mathsf{l}^{\text{n-opt}}\,, \tag{2.11}$$

where $\mathfrak{u}_h=\sum_{j=1}^N(\mathsf{u}_h)_j\mathfrak{u}_j\in\mathcal{U}_h$ is the discrete solution; the Gram matrix $\mathsf{G}_{ij}=(\mathfrak{v}_i,\mathfrak{v}_j)_{\mathcal{V}}$ is a discretization of $\mathcal{R}_{\mathcal{V}_r}$; $\mathsf{B}_{ij}=b(\mathfrak{u}_j,\mathfrak{v}_i)$ and $\mathsf{l}_i=\ell(\mathfrak{v}_i)$ are called the enriched stiffness matrix and load; and $\mathsf{B}_{ij}^{\text{n-opt}}=b(\mathfrak{u}_j,\mathfrak{v}_i^{\text{n-opt}})$ and $\mathsf{l}_i^{\text{n-opt}}=\ell(\mathfrak{v}_i^{\text{n-opt}})$ are the near-optimal stiffness matrix and load. Clearly the enriched stiffness matrix is rectangular and tall, $\mathsf{B}\in\mathbb{R}^{M\times N}$ with $M\geq N$, while the near-optimal stiffness matrix is square and symmetric positive-definite, $\mathsf{B}^{\text{n-opt}}\in\mathbb{R}^{N\times N}$. Computationally, the Gram matrix ($\mathsf{G}\in\mathbb{R}^{M\times M}$), enriched stiffness matrix ($\mathsf{B}\in\mathbb{R}^{M\times N}$) and enriched load ($\mathsf{l}\in\mathbb{R}^M$) are computed first, and using them, the near-optimal stiffness matrix ($\mathsf{B}^{\text{n-opt}}=\mathsf{B}^{\mathsf{T}}\mathsf{G}^{-1}\mathsf{B}\in\mathbb{R}^{N\times N}$) and near-optimal load ($\mathsf{l}^{\text{n-opt}}=\mathsf{B}^{\mathsf{T}}\mathsf{G}^{-1}\mathsf{l}\in\mathbb{R}^N$) are computed next, after which the basis coefficients for the discrete solution ($\mathsf{u}_h\in\mathbb{R}^N$) are solved for. All this derivation holds for an arbitrary linear variational formulation, so in particular it can be used for the ultraweak formulations in (2.3) and (2.7). The numerical method is near-optimal in that it is designed to approximate the optimal method (with B^{opt}), so technically speaking it is not necessarily stable, but in practice it typically is or can be made more stable (if it is not stable simply enrich $\mathcal{V}_r$ even more so that $M\gg N$). In fact, the stability of the near-optimal method can often be proved by constructing a Fortin operator, $\Pi_F:\mathcal{V}\to\mathcal{V}_r$ [50, 19].

However, there are major differences from applying this numerical method to the ultraweak formulation in (2.3) and the broken ultraweak formulation in (2.7). Namely, for the standard ultraweak formulation the enriched stiffness matrix, B, and the Gram matrix, G, are assembled globally first (fortunately both matrices are sparse), and then the near-optimal stiffness matrix, $\mathsf{B}^{\text{n-opt}}$ is computed at a global level. This is very

expensive, especially the inversion of G. Thus, despite many advantages, the method is not very practical in this sense. However, when using broken test spaces, as in the broken ultraweak formulation, the matrix G has a disjoint diagonal block structure, with each block representing an element of the mesh. Hence, the Gram matrix can be inverted locally, and this has major implications, since the local near-optimal stiffness matrix $(\mathsf{B}^{\text{n-opt}})_K$ can be computed directly at each element $K \in \mathcal{T}$, and $\mathsf{B}^{\text{n-opt}}$ can be assembled as in any other finite element method. Therefore, using formulations with broken test spaces localizes much of the computations, which allows for parallelization of the assembly, but the local computations themselves become more expensive in view of the additional skeleton variables. Indeed, the broken ultraweak formulation in (2.7) has an enriched stiffness matrix with the structure,

$$\{\mathfrak{v}_i\}_{i=1}^M \left\{ \begin{bmatrix} \overbrace{\begin{matrix} | \\ \mathsf{B}_0 \\ | \end{matrix}}^{\{(\mathfrak{u}_0)_j\}_{j=1}^{N_0}} & \overbrace{\begin{matrix} | \\ \hat{\mathsf{B}} \\ | \end{matrix}}^{\{\hat{\mathfrak{u}}_j\}_{j=1}^{\hat{N}}} \end{bmatrix} \right. = \mathsf{B} = \begin{bmatrix} \mathsf{B}_{uv} & \mathsf{B}_{\boldsymbol{q}v} & \mathsf{B}_{\hat{u}v} & \mathsf{B}_{\hat{q}_{\hat{\mathfrak{n}}}v} \\ \mathsf{B}_{u\boldsymbol{\tau}} & \mathsf{B}_{\boldsymbol{q}\boldsymbol{\tau}} & \mathsf{B}_{\hat{u}\boldsymbol{\tau}} & \mathsf{B}_{\hat{q}_{\hat{\mathfrak{n}}}\boldsymbol{\tau}} \end{bmatrix} \tag{2.12}$$

where $(\mathsf{B}_0)_{ij} = b_0((\mathfrak{u}_0)_j, \mathfrak{v}_i)$ and $\hat{\mathsf{B}}_{ij} = \hat{b}(\hat{\mathfrak{u}}_j, \mathfrak{v}_i)$, with the $\mathcal{U}_h$-basis $\{\mathfrak{u}_j\}_{j=1}^N = \{((\mathfrak{u}_0)_j, 0)\}_{j=1}^{N_0} \cup \{(0, \hat{\mathfrak{u}}_j)\}_{j=1}^{\hat{N}}$ so that $N = N_0 + \hat{N}$, and similarly with the other sub-blocks. The application of minimum residual methods to variational formulations with broken test spaces is referred to in the literature as the *DPG methodology*. The methodology itself can be applied generally to broken variational formulations other than the ultraweak, even for the same equations (broken primal formulation, broken mixed formulation, etc.) [53, 19], and each application results in a different DPG method (just like the Galerkin methodology can be applied to different formulations as long as $\mathcal{U}_h = \mathcal{V}_h$). Nonetheless, due to the lack of inter-element compatibility restrictions on the $\mathcal{U}_0$-part of the trial space (which lies in copies of L^2), the ultraweak formulation is the most attractive to implement a DPG method for general polygons.

It is worth mentioning that the DPG methodology carries a natural arbitrary-order residual-based a posteriori error estimator. The expression for the residual is,

$$\|\mathcal{B}\mathfrak{u}_h - \ell\|_{\mathcal{V}'}^2 \approx \|\mathcal{R}_{\mathcal{V}_r}^{-1}(\mathcal{B}\mathfrak{u}_h - \ell)\|_{\mathcal{V}}^2 = (\mathsf{B}\mathsf{u}_h - \mathsf{l})^{\mathsf{T}}\mathsf{G}^{-1}(\mathsf{B}\mathsf{u}_h - \mathsf{l})\,, \tag{2.13}$$

where $\mathfrak{u}_h$ (and u_h) is the solution computed from the DPG method. Since the test spaces are broken, it is possible to make these computations locally, meaning that this serves as an a posteriori error estimator that may drive different adaptivity strategies. Adaptivity in its own right is a very interesting subject to study in the context of general polygons, as they provide great flexibility for the implementation of such strategies. More information is included in Section 3.4.

A final comment on minimum residual methods, including all DPG methods, is that the choice of test norm (or inner product) for $\mathcal{V}$, which appears in the computation of G, does change things. Generally speaking, the standard norms are usually chosen as norms. For example, for the broken ultraweak formulation in (2.7), the standard norm is,

$$\|(v, \boldsymbol{\tau})\|_{\mathcal{V}}^2 = \|v\|_{H^1(\mathcal{T})}^2 + \|\boldsymbol{\tau}\|_{H(\operatorname{div},\mathcal{T})}^2 = (v, v)_{\mathcal{T}} + (\nabla v, \nabla v)_{\mathcal{T}} + (\boldsymbol{\tau}, \boldsymbol{\tau})_{\mathcal{T}} + (\operatorname{div}\boldsymbol{\tau}, \operatorname{div}\boldsymbol{\tau})_{\mathcal{T}}\,. \tag{2.14}$$

However, other choices of norm making $\mathcal{V}$ a Hilbert space are possible, and these will affect the final results of the method. In the special case of broken ultraweak formulations, another choice with interesting properties is the adjoint graph norm, which can be derived from the ultraweak formulation itself (see (2.3)) as,

$$\|(v, \boldsymbol{\tau})\|_{\mathcal{V}}^2 = \|\tfrac{1}{k}\boldsymbol{\tau} - \nabla v\|_{\boldsymbol{L}^2(\mathcal{T})}^2 + \|-\operatorname{div}\boldsymbol{\tau}\|_{L^2(\mathcal{T})}^2 + \varepsilon^2\big(\|v\|_{L^2(\mathcal{T})}^2 + \|\boldsymbol{\tau}\|_{\boldsymbol{L}^2(\mathcal{T})}^2\big)\,, \tag{2.15}$$

where $\|\cdot\|^2_{L^2(\mathcal{T})} = (\cdot,\cdot)_{\mathcal{T}}$ and the same with $\|\cdot\|^2_{\boldsymbol{L}^2(\mathcal{T})}$. The term with the ε^2 is said to make the norm localizable, since without it, it would not even be a norm for arbitrary broken functions $v \in H^1(\mathcal{T})$ (although it would be a norm for $v \in H^1_0(\Omega)$). Here, the value of $\varepsilon > 0$ can be arbitrary, but the case when ε is small is of particular interest [49]. The corresponding inner products for the (real-valued) Hilbert space $\mathcal{V}$ can be derived from the polarization identity, $(\mathfrak{v}_1, \mathfrak{v}_2)_{\mathcal{V}} = \frac{1}{4}\big(\|\mathfrak{v}_1 + \mathfrak{v}_2\|^2_{\mathcal{V}} - \|\mathfrak{v}_1 - \mathfrak{v}_2\|^2_{\mathcal{V}}\big)$.

2.3 Choice of trial and test spaces

The choice of trial and test spaces is important to determine the convergence of the method. As we mentioned previously, whenever there are strict inter-element compatibility requirements, the possibilities seem to be severely limited. Particularly, the problem seems to be extremely complicated for general polygons which we want to discretize with high polynomial orders (to eventually have better convergence behavior, among other reasons). Fortunately, the $\mathcal{U}_0$ trial space component of the broken ultraweak formulation (see (2.6)) consists of copies of L^2, so their discretization may be discontinuous across the elements. Even better, the test spaces are broken, so their discretization ought to be discontinuous across elements too. This freedom to be discontinuous allows to create bases locally, disregarding the neighboring elements. In particular, bases may be defined by restriction to the polygon of interest, as we will see next.

Our procedure is similar to that in [18] where a bounding box was utilized, but we decided to use a bounding triangle instead. This is computed cheaply by first determining the centroid of the polygon and its furthest vertex from the centroid, so that the centroid becomes the center of a bounding circle passing through the furthest vertex. Then, the bounding triangle is computed as the equilateral triangle having the previous circle inscribed and such that one of its edge-midpoints is the furthest vertex. This is shown in Figure 2.1. Lastly, the "usual" high-order polynomial shape functions for the triangle are used and then restricted to the polygon. We use the term "usual" liberally, but to clarify, we include further details below.

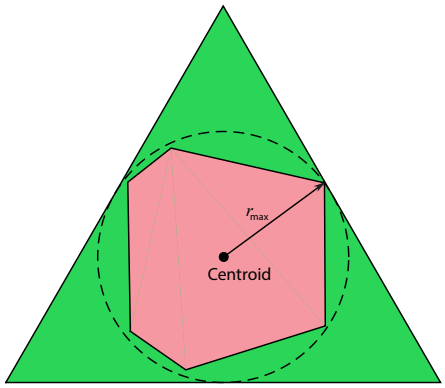

Figure 2.1: Bounding triangle of polygonal element. The triangle is such that the bounding circle centered at the polygon centroid is inscribed and such that the furthest vertex (which also determines the radius of the circle) is one of the edge midpoints.

First of all, note that there are several spaces at the infinite-dimensional level which we want to discretize using this technique. Namely, the test space components, $H^1(\mathcal{T})$ and $\boldsymbol{H}(\mathrm{div}, \mathcal{T})$, and the $\mathcal{U}_0$ trial space component, which may be represented by $L^2(\Omega)$. Due to our technique, this reduces to finding local discretizations of $H^1(T_K)$, $\boldsymbol{H}(\mathrm{div}, T_K)$ and $L^2(T_K)$, where T_K is the bounding triangle of the polygonal

element $K \in \mathcal{T}$. These three spaces actually form a differential exact sequence, and it is convenient that their respective discretizations do too. This is satisfied by the classical Nédélec sequence of the first type [38, 43],

$$\begin{array}{ccccc} H^1(T_K) & \xrightarrow{\text{curl}} & \boldsymbol{H}(\text{div}, T_K) & \xrightarrow{\nabla\cdot} & L^2(T_K) \\ \cup\!| & & \cup\!| & & \cup\!| \\ \mathcal{P}^p(T_K) & \xrightarrow{\text{curl}} & \mathcal{RT}^p(T_K) & \xrightarrow{\nabla\cdot} & \mathcal{P}^{p-1}(T_K)\,, \end{array} \tag{2.16}$$

where $\mathcal{P}^p(T_K)$ are the polynomials in $x = (x_1, x_2)$ of total order less than or equal $p \in \mathbb{N}$, the 2D Raviart-Thomas space is $\mathcal{RT}^p(T_K) = (\mathcal{P}^{p-1}(T_K))^2 + x\mathcal{P}^{p-1}(T_K)$ (a rotation of the 2D Nédélec space), and the 1D-to-2D curl operator is defined as $\text{curl}(u) = \left(\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right)\nabla u$ for any $u \in H^1(T_K)$. Notice that the parameter p represents the order of the discrete *sequence*, but does not necessarily coincide with the order of the polynomials of a particular discretization. For example if $p = 3$, the discretization of $L^2(T_K)$ are the polynomials of at most total order $p-1 = 2$. This sequence has many nice properties, and precisely because of these, we preferred to use a bounding triangle over a bounding box. In particular, the spaces are invariant under affine transformations (the spaces remain the same even if the bounding triangle is arbitrarily rotated about the polygon centroid); the overall drop of polynomial order across the sequence is one (from $\mathcal{P}^p(T_K)$ to $\mathcal{P}^{p-1}(T_K)$); the approximation properties are good (see Appendix A); and they are the smallest such spaces with all these properties [3, §3.4]. Having said this, a similar procedure can be carried out for a bounding box, Q_K of $K \in \mathcal{T}$, where the spaces become

$$\begin{array}{ccccc} H^1(Q_K) & \xrightarrow{\text{curl}} & \boldsymbol{H}(\text{div}, Q_K) & \xrightarrow{\nabla\cdot} & L^2(Q_K) \\ \cup\!| & & \cup\!| & & \cup\!| \\ \mathcal{Q}^{p,p}(Q_K) & \xrightarrow{\text{curl}} & \mathcal{Q}^{p,p-1}(Q_K)\times\mathcal{Q}^{p-1,p}(Q_K) & \xrightarrow{\nabla\cdot} & \mathcal{Q}^{p-1,p-1}(Q_K)\,, \end{array} \tag{2.17}$$

with $\mathcal{Q}^{p,q}(Q_K) = \mathcal{P}^p(x_1) \otimes \mathcal{P}^q(x_2)$. In either case, the final spaces for the polygon $K \subseteq T_K$ (or $K \subseteq Q_K$) are defined by restricting the domain to $K \in \mathcal{T}$, so denote them by $\mathcal{P}^p(K)$ and $\mathcal{RT}^p(K)$ instead (and similarly when using a bounding box).

The only remaining spaces to specify are those of the skeleton variables lying in the $\hat{\mathfrak{U}}$ trial space component (see (2.6)). These can also be deduced using the same philosophy of exact sequences, but utilizing the traces instead. Indeed, the spaces $H_0^{1/2}(\partial\mathcal{T})$ and $H^{-1/2}(\partial\mathcal{T})$ are merely $\mathcal{T}$-tuples of compatible traces of $H^1(K)$ and normal-traces of $\boldsymbol{H}(\text{div}, K)$ respectively. Supposing there were two elements of different type (a triangle and a quadrilateral) sharing an edge, the discrete spaces should be compatible across that edge. Note this is indeed the case when considering the H^1-discretizations of triangles and quadrilaterals: even though the discretizations themselves are different ($\mathcal{P}^p$ and $\mathcal{Q}^{p,p}$), their restrictions to edges are exactly the same, $\mathcal{P}^p(e)$, where e represents an edge parametrized linearly by t_e. The same occurs with the $\boldsymbol{H}(\text{div})$-discretizations, which have $\mathcal{P}^{p-1}(e)$ as normal-trace along the edges. Additionally, the H^1-discretizations should be compatible at vertices. This is consistent with 1D discretizations of H^1 and L^2, which also form an exact sequence, but instead occurring along the boundary of each element and being edge-parametrized along all edges. This pattern should hold for arbitrary polygons as well. For this, let $\mathcal{E}(K)$ be the set of edges of a polygon $K \in \mathcal{T}$, and define the local discretizations,

$$\begin{aligned} \mathcal{P}^{p-1}(\partial K) &= \{\hat{w}_K \mid \hat{w}_K|_e \in \mathcal{P}^{p-1}(e),\ \forall e \in \mathcal{E}(K)\} \subseteq H^{-1/2}(\partial K)\,, \\ \mathcal{P}^p_C(\partial K) &= \mathcal{P}^p(\partial K) \cap C(\partial K) \subseteq H^{1/2}(\partial K)\,, \end{aligned} \tag{2.18}$$

where $C(\partial K)$ are the continuous functions in ∂K (the intersection ensures that values of neighboring edges coincide at a common vertex), and the local trace spaces are $H^{1/2}(\partial K) = \{\hat{u}_K = u|_{\partial K} \mid u \in H^1(K)\}$ and $H^{-1/2}(\partial K) = \{(\hat{q}_{\hat{\mathbf{n}}})_K = \boldsymbol{q}|_{\partial K}\cdot\hat{\mathbf{n}}_K \mid \boldsymbol{q} \in H(\text{div}, K)\}$.

Finally, we have enough information to actually globally define the discrete trial space. For a value of $p \in \mathbb{N}$, it is

$$\begin{aligned}\mathcal{U}_h = \big\{(u, \boldsymbol{q}, \hat{u}, \hat{q}_{\hat{\mathbf{n}}}) \in \mathcal{U} \mid u|_K \in \mathcal{P}^{p-1}(K),\, \boldsymbol{q}|_K \in \big(\mathcal{P}^{p-1}(K)\big)^2,\\ \hat{u}_K \in \mathcal{P}^p_C(\partial K),\, (\hat{q}_{\hat{\mathbf{n}}})_K \in \mathcal{P}^{p-1}(\partial K),\, \forall K \in \mathcal{T}\big\}\,.\end{aligned} \tag{2.19}$$

Notice that the condition $(u, \boldsymbol{q}, \hat{u}, \hat{q}_{\hat{\mathbf{n}}}) \in \mathcal{U}$ (so $(\hat{u}, \hat{q}_{\hat{\mathbf{n}}}) \in \hat{\mathcal{U}}$) implies that $\hat{u}$ vanishes at the boundaries, that $\hat{u}_{K_1}|_e = \hat{u}_{K_2}|_e$, and that $(\hat{q}_{\hat{\mathbf{n}}})_{K_1}|_e = -(\hat{q}_{\hat{\mathbf{n}}})_{K_2}|_e$, where e is a common edge between the elements K_1 and K_2. No such compatibility implications exist for $(u, \boldsymbol{q}) \in \mathcal{U}_0$. For the enriched test space, the discretizations are chosen from a sequence of order $p + \Delta p$, and we say the space is p-enriched, so that

$$\mathcal{V}_r = \big\{(v, \boldsymbol{\tau}) \mid v|_K \in \mathcal{P}^{p+\Delta p_K}(K),\, \boldsymbol{\tau}|_K \in \mathcal{RT}^{p+\Delta p_K}(K),\, \forall K \in \mathcal{T}\big\}\,. \tag{2.20}$$

The notation Δp_K indicates that this value may change depending on the element. In fact, recall that for minimum residual methods to work, $M = \dim(\mathcal{V}_r) \geq \dim(\mathcal{U}_h) = N$, and this restriction on the dimensionality should hold locally as well. Thus, Δp_K has to be chosen such that this condition holds. In polygonal element methods, this is important, because when a polygon has many sides, the size of the local trial space may be quite large, and as a result a large value of Δp_K must be chosen for that particular element. Indeed, consider an interior n-sided polygonal element K (so that $\partial K \cap \partial\Omega = \varnothing$). Its local trial and test space dimensions would be

$$\begin{aligned}\dim(\mathcal{U}_h(K)) &= \overbrace{\tfrac{1}{2}p(p+1)}^{u|_K} + \overbrace{p(p+1)}^{\boldsymbol{q}|_K} + \overbrace{n + n(p-1)}^{\hat{u}_K} + \overbrace{np}^{(\hat{q}_{\hat{\mathbf{n}}})_K}\,,\\ \dim(\mathcal{V}_r(K)) &= \underbrace{\tfrac{1}{2}(p+\Delta p_K+1)(p+\Delta p_K+2)}_{v|_K} + \underbrace{(p+\Delta p_K)(p+\Delta p_K+2)}_{\boldsymbol{\tau}|_K}\,.\end{aligned} \tag{2.21}$$

Thus, for $p = 2$ and $n = 3$ (a triangle), $\dim(\mathcal{U}_h(K)) = 21$, so that a value of $\Delta p_K = 1$ is sufficient ($\dim(\mathcal{V}_r(K)) = 25$); but if $p = 2$ and $n = 8$ (an octagon), $\dim(\mathcal{U}_h(K)) = 41$, a value of at least $\Delta p_K = 3$ (so that $\dim(\mathcal{V}_r(K)) = 56$) is required. Having said this, sometimes for simplicity a valid value of Δp is chosen uniformly throughout the mesh. For illustrative purposes, some representative functions of the components $\mathcal{U}_h(K)$ and $\mathcal{V}_r(K)$ are shown in Figure 2.2 for the different energy spaces and multiple values of p.

A final comment is that it would seem that the method is very expensive due to the large number of variables in the trial space $\mathcal{U}_h$, but this may be deceiving. In fact, all of the $\mathcal{U}_0$ trial space component can be statically condensed locally for ultraweak formulations, meaning that this part of the near-optimal stiffness matrix, $\mathsf{B}^{\text{n-opt}}$, can be effectively removed via Schur complements, and the only remaining connectivity will be that coming from the skeleton variables in $\hat{\mathcal{U}}$. So computationally speaking, solving with this variational formulation is not as costly as one might initially imagine.

2.4 Convergence

The method described up to this point is technically speaking, a conforming finite element method, since the subspaces used to discretize the ultraweak variational formulation are rigorously subspaces of the infinite dimensional trial and test spaces. This has the convenient implication that "standard" convergence theory can be applied. However, this is an understatement in that the skeleton variables themselves are pretty far from being standard. The details are left to Appendix A, but the main result is stated here along with the key assumptions.

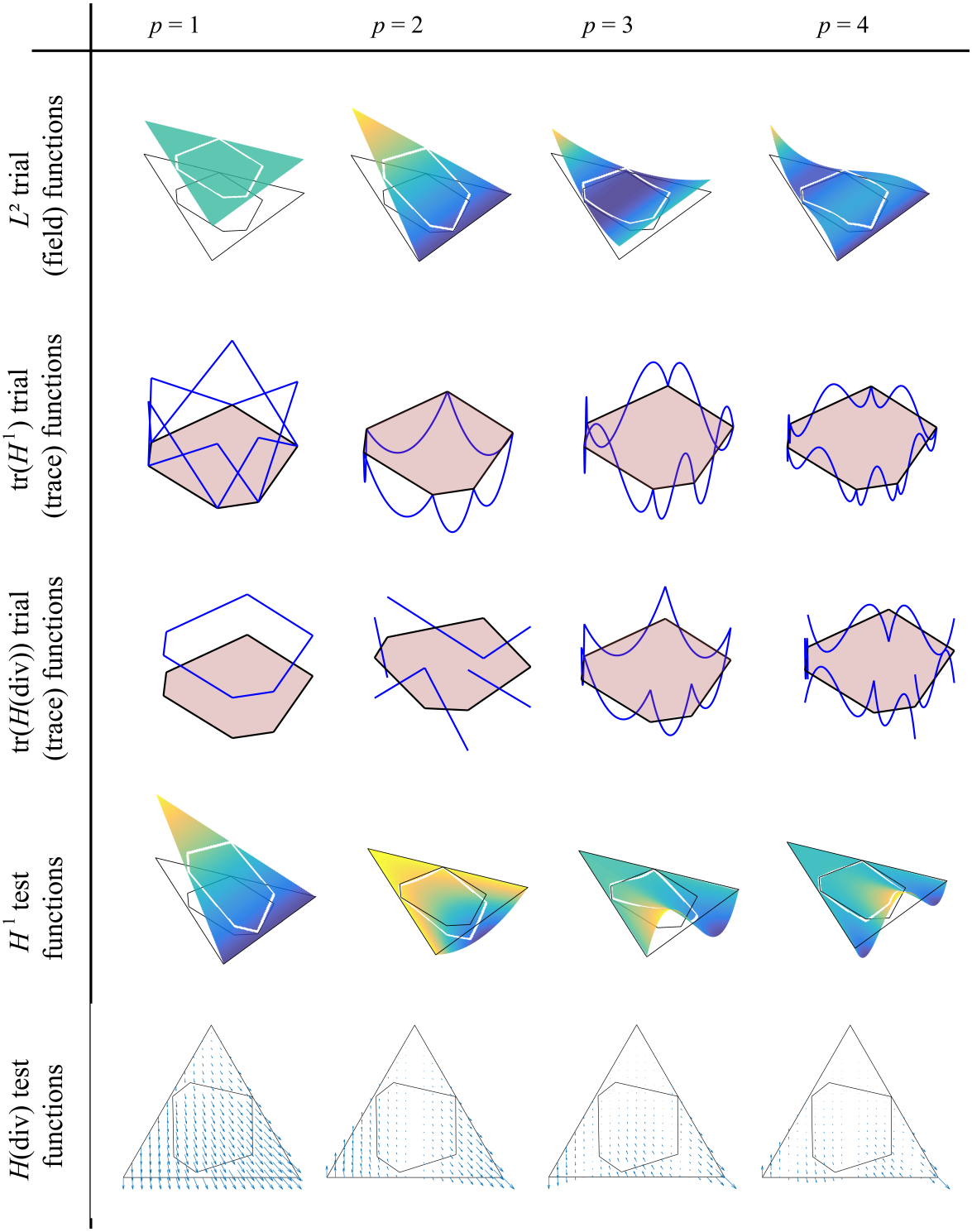

Figure 2.2: Some of the shape functions on a polygonal element used either as trial or test variables in the ultraweak DPG method. They are classified by the energy space ($H^1(K)$, $\boldsymbol{H}(\mathrm{div}, K)$, $L^2(K)$ and their respective traces) and for different values of the parameter $p \in \mathbb{N}$ which denotes the order of the differential *sequence*.

Definition. A collection of subsets of $\mathbb{R}^2$, $\mathcal{T}_\mathcal{K}$, is said to have the finite overlap condition if

$$\mathrm{ov}(\mathcal{T}_\mathcal{K}) = \sup_{x\in\mathbb{R}^2} \mathrm{ov}(x) < \infty\,, \qquad \mathrm{ov}(x) = |\{\mathcal{K}\in\mathcal{T}_\mathcal{K} \mid x\in\mathcal{K}\}|\,. \tag{2.22}$$

If there is a family of such collections given by a parameter $\mathfrak{h}$, $\mathcal{T}_{\mathcal{K},\mathfrak{h}}$, the finite overlap condition is said to be robust in $\mathfrak{h}$ if there exists an integer $M_{\mathrm{ov}} > 0$ independent of $\mathfrak{h}$ such that $\mathrm{ov}(\mathcal{T}_{\mathcal{K},\mathfrak{h}}) \le M_{\mathrm{ov}}$ for any $\mathfrak{h}$.

Definition. A triangulation $\mathcal{T}(K) = \{\mathcal{T}_i(K)\}_{i\in I_K}$ (with I_K finite) of a polygonal element K is said to be edge-compatible if for each edge of K, only one $\mathcal{T}_i(K)$ shares that edge. The triangulation is additionally said to be shape-regular if all $\mathcal{T}_i(K)$ satisfy some uniform shape-regularity condition (e.g. they satisfy a minimum angle condition or the ratio of their diameters to their incircle radii remains bounded).

Theorem 2.1. *Let $p\in\mathbb{N}$ be a polynomial order and $\mathcal{T}_\mathfrak{h}$ be a family of polygonal meshes discretizing the domain Ω, such that there exist robust ($\mathcal{T}_\mathfrak{h}$-independent) shape-regular edge-compatible triangulations of each $K\in\mathcal{T}_\mathfrak{h}$. Assume that the associated collections of bounding triangles (see Section (2.3)), $\mathcal{T}_{T,\mathfrak{h}} = \{T_K\}_{K\in\mathcal{T}_\mathfrak{h}}$, where T_K is the bounding triangle of a polygonal element K, satisfy a robust finite overlap condition. Also, assume the existence of a linear and continuous Fortin operator, $\Pi_F : \mathcal{V}\to\mathcal{V}_r$, satisfying the orthogonality condition, $b(\mathfrak{u}_h, \mathfrak{v}-\Pi_F\mathfrak{v}) = 0$, for all $\mathfrak{u}_h\in\mathcal{U}_h$ and $\mathfrak{v}\in\mathcal{V}$, and with a continuity bound, $M_F > 0$ (so $\Pi_F\mathfrak{v} \le M_F\mathfrak{v}$ for all $\mathfrak{v}\in\mathcal{V}$), independent of $\mathcal{T}_\mathfrak{h}$, where $b : \mathcal{U}\times\mathcal{V}\to\mathbb{R}$, $\ell : \mathcal{V}\to\mathbb{R}$, $\mathcal{U}_h$ and $\mathcal{V}_r$ are given in (2.6), (2.7), (2.19) and (2.20). Then, the problem of finding $\mathfrak{u}_h\in\mathcal{U}_h$ such that*

$$b(\mathfrak{u}_h, \mathfrak{v}_h) = \ell(\mathfrak{v}_h)\,, \qquad \forall\mathfrak{v}_h\in\mathcal{V}_h = \mathcal{R}_{\mathcal{V}_r}^{-1}\mathcal{B}\mathcal{U}_h\,,$$

has a unique solution. When compared to the unique solution of the infinite dimensional problem, $\mathfrak{u}\in\mathcal{U}$ (so $b(\mathfrak{u},\mathfrak{v}) = \ell(\mathfrak{v})$ for all $\mathfrak{v}\in\mathcal{V}$), and assuming it is regular enough, $\mathfrak{u}\in\mathcal{U}^s\subseteq\mathcal{U}$, for an $s > \frac{1}{2}$, the following h-convergence estimate holds,

$$\|\mathfrak{u}-\mathfrak{u}_h\|_\mathcal{U} \le Ch^{\min\{s,p\}}\|\mathfrak{u}\|_{\mathcal{U}^s}\,,$$

where $h = \sup_{K\in\mathcal{T}_\mathfrak{h}}\mathrm{diam}(K)$ and $C = C(s,p,\Omega) > 0$ is a constant independent of $\mathcal{T}_\mathfrak{h}$ (and so of h as well). For more details about $s > \frac{1}{2}$ and $\mathcal{U}^s$ see Appendix A. Moreover, in the p-asymptotic limit $C = \widetilde{C}(\ln p)p^{-s}$ where $\widetilde{C} = \widetilde{C}(s,\Omega)$ is independent of p.

3 Numerical results

The implementation of the present code was made in Matlab, but the shape functions are called through an interface to Fortran. The shape functions used were those described in [43]. The numerical integration was carried out by splitting the polygons into triangles (through Delaunay triangulation in most cases), and then using Gaussian quadrature on each triangle (the Gaussian quadrature points and weights were carefully mapped back from a square), so that polynomials integrands of a certain order were computed up to machine precision.

Several examples with different features involving polygonal elements were solved using the numerical method. In all cases, Poisson's equation representing the nondimensionalized steady-state heat equation was solved in the domain $\Omega = (0,1)^2$. Unless otherwise stated, bounding triangles were utilized (as opposed to bounding boxes) and the (nondimensional) conductivity was taken as $k = 1$. Also, a default uniform value of $\Delta p = 1$ was used, but was increased if deemed necessary (see Section 2.3). The first example is simply

an implementation of a polygonal mesh of convex elements with nontrivial number of sides (greater than 4). The second example delves with highly distorted concave elements in the mesh. The third example is inspired by problems in geoscience, where arbitrary faults separating different material properties occur. To model this, we placed a uniform grid and superposed an arbitrary fault at an angle which in turn cuts the mesh into polygons (some pentagons and triangles), and divides the domain in terms of material properties as well. In all these three examples, "uniform" refinements were analyzed for different values of $p \in \mathbb{N}$, in the sense that the largest element diameter was roughly cut in half with each refinement. Lastly, the fourth example describes a polygonal adaptivity scheme and makes a careful comparison with more standard adaptive methods (involving standard element shapes). This is particularly important, since adaptive polygonal algorithms have applications in topology optimization [71, 45] and crack propagation [67, 56], and our numerical DPG method *always* has an arbitrary p a posteriori error estimator to drive such adaptive applications.

In the computations that follow, only the relative error in the $\mathcal{U}_0$ trial space component was reported, since rigorously computing the norms in the $\hat{\mathcal{U}}$ trial space component was simply not viable. The $\mathcal{U}_0$ relative error was defined as

$$\text{Relative error} = \frac{\|\mathfrak{u}_0 - (\mathfrak{u}_0)_h\|_{\mathcal{U}_0}}{\|\mathfrak{u}_0\|_{\mathcal{U}_0}}\,, \quad \text{with} \quad \|(u,\boldsymbol{q})\|^2_{\mathcal{U}_0} = \|u\|^2_{L^2(\Omega)} + \|\boldsymbol{q}\|^2_{\boldsymbol{L}^2(\Omega)} = (u,u)_\Omega + (\boldsymbol{q},\boldsymbol{q})_\Omega\,, \tag{3.1}$$

where $\mathfrak{u}_0$ was the exact solution and $(\mathfrak{u}_0)_h$ was the computed solution from the numerical method.

3.1 Mesh of convex polygons

This case consists of meshes of convex polygonal elements. The polygonal mesh generator PolyMesher [72] was used to generate the meshes. In Figure 3.1 an initial mesh and three refinements are displayed, where the elements are colored according to their number of sides, ranging from 4 (quadrilaterals) to 7 (heptagons). Independently of the type of polygon, the construction of shape functions is invariant and follows the procedure described in Section 2. The manufactured solution,

$$u(x,y) = \sin(\pi x)\sin(\pi y)\,, \tag{3.2}$$

for $(x,y) \in \Omega = (0,1)^2$ was used to determine the forcing (the internal heat source r in (2.1)) and boundary conditions of u at $\partial\Omega$ (in this case vanishing boundary conditions).

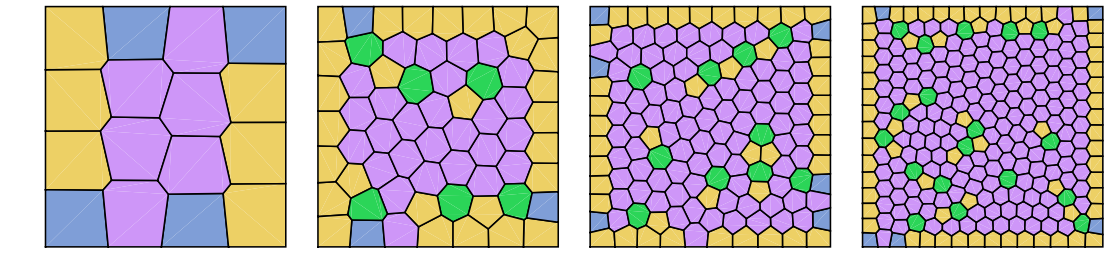

Figure 3.1: Four refinement levels of a mesh of arbitrary convex polygonal elements. The elements are colored according to their number of sides.

For the trial spaces corresponding to values of $p = 1, 2$ the uniform value of $\Delta p = 2$ was used across the mesh for the enriched test space, while for $p = 3, 4$, an associated value of $\Delta p = 3$ was utilized. The

numerical results are plotted and presented in Figure 3.2 when $p = 4$, including the skeleton-trace of the temperature (which is a skeleton variable), the temperature and the heat flow. Additionally, the relative error, calculated through (3.1), is shown in Figure 3.3, where the expected h-convergence rates were observed for all values of p utilized (the behavior is of the form h^p as established by Theorem 2.1). Note that the number degrees of freedom, N_{dof}, is proportional to h^2, and this is why the slope indicators in Figure 3.3 display a 2 on one of the sides, while the other label corresponds to the h-convergence rate, $\widetilde{p}$ (so that $\frac{\widetilde{p}}{2}$ is the N_{dof}-convergence rate).

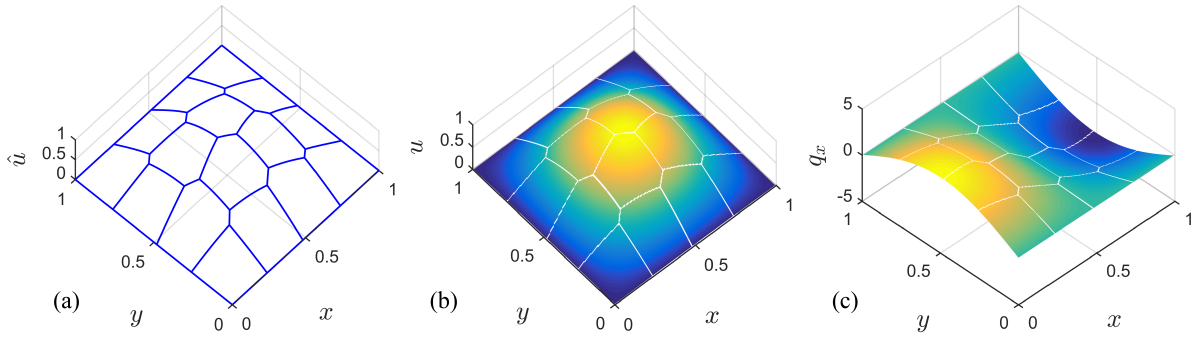

Figure 3.2: Numerical results arising from the manufactured problem in (3.2) on the coarse mesh from Figure 3.1(a) using $p = 4$ and $\Delta p = 3$: (a) skeleton-trace of the temperature, (b) temperature, (c) first component of the heat flow.

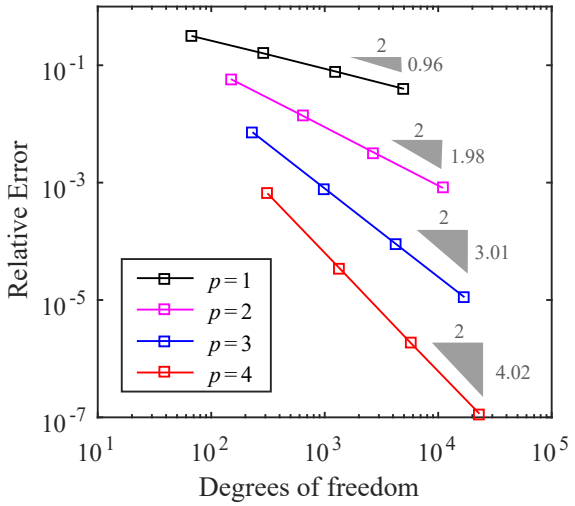

Figure 3.3: Relative error as a function of degrees of freedom. The h-convergence behavior is displayed at different p for the meshes of arbitrary convex polygons in Figure 3.1.

3.2 Distorted mesh

To show the robustness of the method with respect to the element shapes, we considered a mesh with highly distorted quadrilaterals, including concave elements. The pattern was then scaled and tessellated to produce refinements, as shown in Figure 3.4. This example is important, because it represents a challenge (or even a

failure) of other methods due to the likely degeneration of either the parametric mapping or the barycentric coordinates in the highly distorted elements. However, in our current method, these elements are treated just like any other, and this shows its flexibility to deal with nonstandard shapes. The value of $\Delta p = 2$ was uniformly used for the enriched test space, and for all values of p. The same problem as in Section 3.1 was solved (see (3.2) for manufactured solution), and just like in the previous example, the solution values and h-convergence rates for $1 \leq p \leq 4$ had the expected behavior (see Figures 3.5 and 3.6).

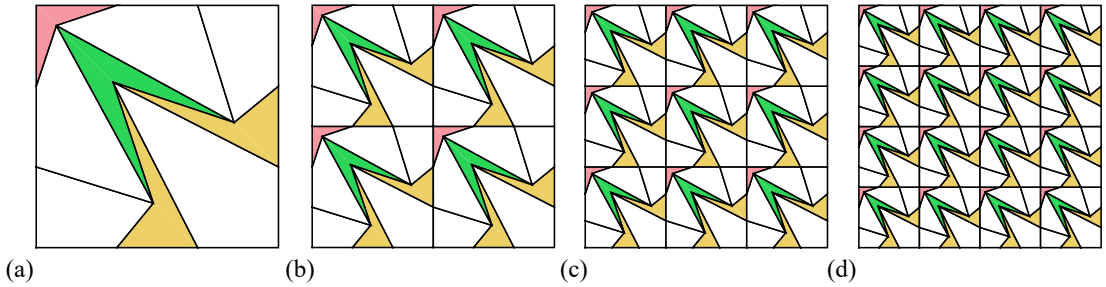

Figure 3.4: Four refinement levels showing a tessellation of a mesh with highly distorted quadrilaterals. The concave elements are colored.

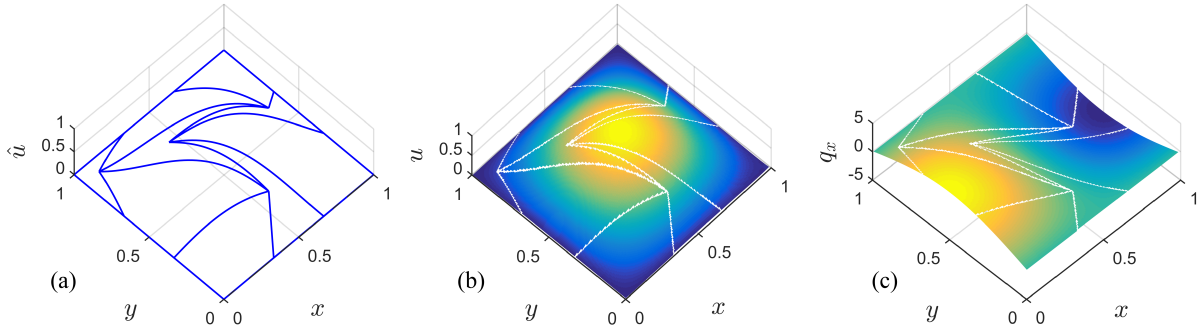

Figure 3.5: Numerical results arising from the manufactured problem in (3.2) on the coarse mesh from Figure 3.4(a) using $p = 4$ and $\Delta p = 2$: (a) skeleton-trace of the temperature, (b) temperature, (c) first component of the heat flow.

3.3 Interface problem

For the third sample problem, there was a different motivation. The inspiration comes from geoscience applications where an arbitrary fault abruptly separates the material properties within a domain. In this case, suppose the domain Ω is composed of two materials with different heat conductivities, which share an interface (for simplicity a straight line at an arbitrary angle dividing completely the square). The heat conductivities are assumed to be uniform at each side of the interface, taking values k_1 and k_2, as depicted on Figure 3.7.

If we have an initial mesh consisting of a uniform grid of quadrilaterals, and if the interface is not horizontal or vertical, a typical refinement to face this problem would require the use of triangles and smaller quadrilaterals surrounding the fault, in order to keep using traditional elements (with more arbitrary faults, the mesh might become unstructured even if it was initially structured before the fault appeared). On the

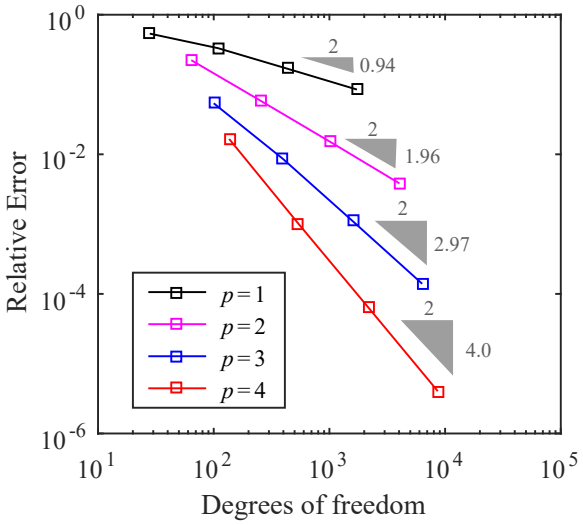

Figure 3.6: Relative error as a function of degrees of freedom. The h-convergence behavior is displayed at different p for the meshes of highly distorted quadrilaterals in Figure 3.4.

other hand, we are able to generate a mesh with a simple algorithm that simply cuts the original mesh, and we obtain triangles, right trapezoids and pentagons. To obtain the refinements, first the underlying regular mesh is uniformly refined, then the elements are cut by the interface line. There is one caveat which, as we will see, is only evident for high values of p or small values of h. When extremely small triangles, when compared to their neighbors, are formed, the system becomes ill conditioned. Thus, it is necessary to deal with them either by node relocation along the interface or by collapsing the nodes of the small triangle into a single node at the interface. We chose to implement the latter approach whenever the area of the small triangle was less than 1% of the area of the underlying grid element. The meshes obtained may be appreciated on Figure 3.8.

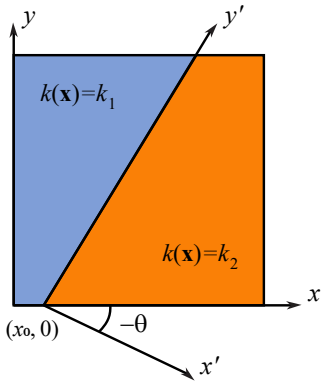

Figure 3.7: Material properties and rotated coordinates in the interface problem.

This problem used a manufactured solution that guaranteed continuity of the temperature and the heat flux across the interface, taking into account the finite jump in the conductivity coefficient. By means of a translated and rotated system of coordinates, and following the notation in Figure 3.7 the exact solution is

given by,

$$u(x',y') = \begin{cases} k_2 \sin(\pi x')\sin(\pi y'), & \text{for } x' \leq 0\,, \\ k_1 \sin(\pi x')\sin(\pi y'), & \text{for } x' > 0\,, \end{cases} \tag{3.3}$$

where the coordinates x' and y' come from a translation and rotation of the reference system defined by the following transformation,

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x - x_0 \\ y \end{pmatrix}\,. \tag{3.4}$$

The values of conductivity and the geometric data used for the numerical computation were $k_1 = 1$, $k_2 = 5$, $x_0 = 0.12$ and $\theta = \tan^{-1}(1/0.65)$. Figure 3.9 shows the appearance of the computed ultraweak solution using $p = 4$ and $\Delta p = 3$. Once again, the expected convergence rates were verified, as can be observed on Figure 3.10. As a side note, without collapsing any nodes in the meshes, the same data points were observed for $1 \leq p \leq 3$, but the last data point of $p = 4$ did behave unexpectedly, so collapsing the nodes is still recommended in general.

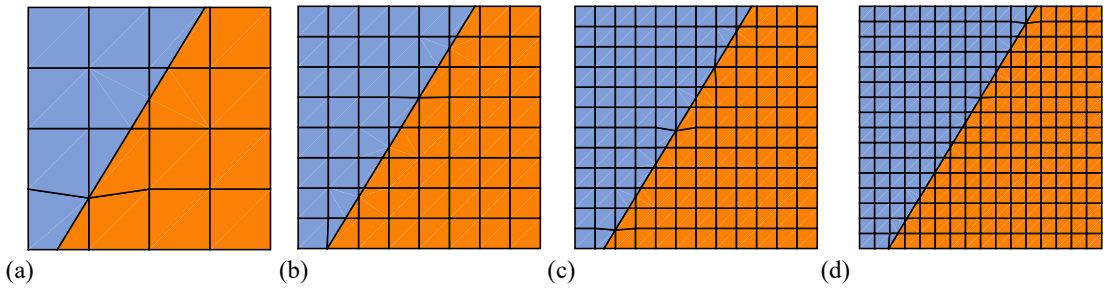

Figure 3.8: Four refinement levels of a mesh with an interface between two materials. Notice some nodes have been collapsed to a node in the interface. This is due to the undesired very small triangles that produce ill conditioning, so they have to be eliminated.

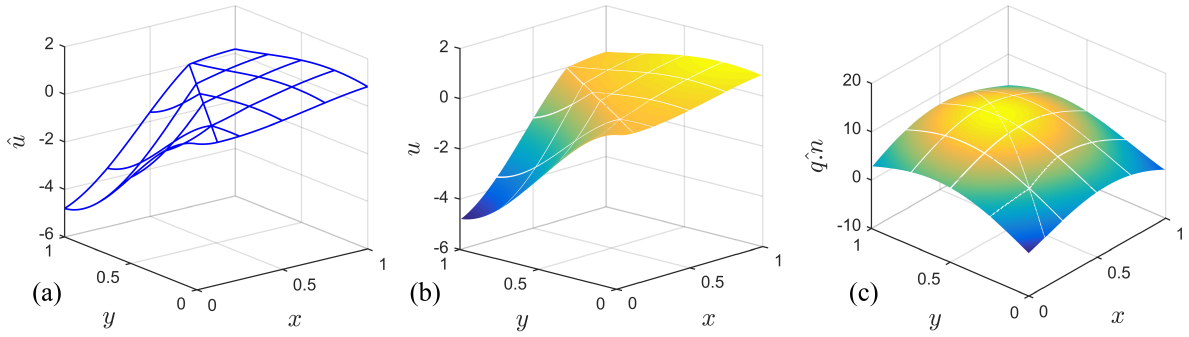

Figure 3.9: Numerical results arising from the manufactured problem in (3.3) and (3.4) on the coarse mesh from Figure 3.8(a) using $p = 4$ and $\Delta p = 3$: (a) skeleton-trace of the temperature, (b) temperature, (c) first component of the heat flow.

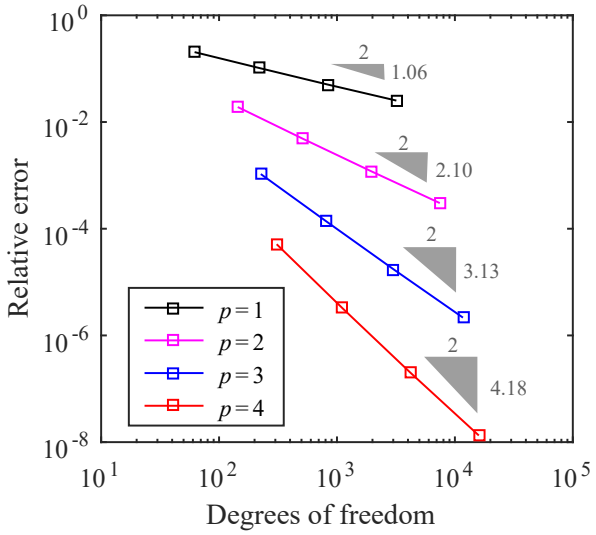

Figure 3.10: Relative error as a function of degrees of freedom. The h-convergence behavior is displayed at different p for the meshes with an interface in Figure 3.8.

3.4 Adaptivity

The last example aims to present a polygonal adaptive strategy. This is of interest as it has direct applications in fracture dynamics [67, 56] and topology optimization [71, 45]. Implementing such a strategy is possible, because the DPG methodology carries a natural arbitrary p a posteriori error estimator (see Section (2.2)). Indeed, if η_K is the a posteriori error estimator (representing the square of the residual as in (2.13)) for $K \in \mathcal{T}$, and $\eta_{\max} = \max_{K\in\mathcal{T}} \eta_K$, then the criterion used to mark an element for refinement is if $\eta_K \geq 0.25\eta_{\max}$ [36].

Once the elements are marked, the actual adaptive strategy must be developed. When using traditional elements, typically hanging nodes arise in the mesh and one must be careful in dealing with them, as only one one "level" of refinement is possible per element (an edge cannot have more than one hanging node). Additionally, sometimes dead-lock scenarios arise (where it is logically impossible to continue refining) and these must be avoided [30]. In short, it may be challenging to implement adaptive strategies.

We will describe a strategy to refine successively. The main advantage with polygonal methods, is that hanging nodes are a completely natural scenario, as they merely represent that a polygon has an extra edge coplanar with another edge. Thus, they do not require an extra level of difficulty in terms of implementation. Here, a practical polygon adaptive strategy was devised and it is illustrated on Figure 3.11: part (a) shows the initial mesh, out of which an element of interest is picked and split into quadrilaterals as depicted in (b); next, any of the resulting elements can be subsequently refined into finer quadrilaterals (c); if a neighbor element needs to be refined too, despite the new collinear vertices generated on the common edge the refinement is applied as if it was the original coarse polygonal element (d).

The manufactured solution for this problem is the sum of two Gaussian surfaces, given by the following function:

$$u(x,y) = \frac{1}{2\pi\sigma^2}\left[e^{-\frac{1}{2}\left(\frac{x-\mu_1}{\sigma}\right)^2} e^{-\frac{1}{2}\left(\frac{y-\mu_1}{\sigma}\right)^2} + e^{-\frac{1}{2}\left(\frac{x-\mu_2}{\sigma}\right)^2} e^{-\frac{1}{2}\left(\frac{y-\mu_2}{\sigma}\right)^2}\right], \tag{3.5}$$

where the standard deviation was $\sigma = \sqrt{10^{-3}}$ and the two means were $\mu_1 = 0.25$ and $\mu_2 = 0.75$. Since in this

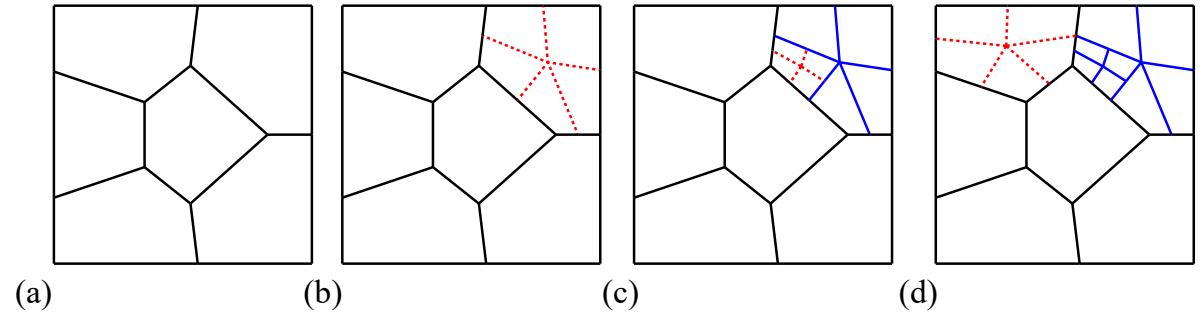

Figure 3.11: A practical local refinement strategy for convex polygons: (a) initial coarse polygonal mesh; (b) on the element of interest line segments are projected from the centroid to every edge midpoint; (c) for a n-gon n quadrilaterals are generated, and the same approach is used to refine one or more of them; (d) when finishing on one element, the strategy can be re-applied to any other coarse element with no involvement of the new collinear vertices on the shared edge.

particular problem we wanted to contrast the method to other approaches for treating hanging nodes, three different initial meshes and refinement strategies were implemented. In Figure 3.12 those three situations are illustrated: (a) an initial mesh of quadrilaterals is refined and the hanging nodes are handled through the conventional approach of no more than one level of hanging nodes being allowed on any single edge (a quadtree-type mesh); (b) an initial mesh of quadrilaterals is refined and the fine meshes are treated as polygonal meshes (in this case the bounding element is a quadrilateral); (c) an initial coarse mesh of polygons is h-refined and the finer meshes contain quadrilaterals following the technique shown in Figure 3.11. Although the proposed refinement technique generates more edges (each new sub-segment becomes an edge), therefore more degrees of freedom, in the end that is compensated by having less elements than a mesh that incorporates hanging nodes, which can be seen by comparing meshes (b) and (c) to (a).

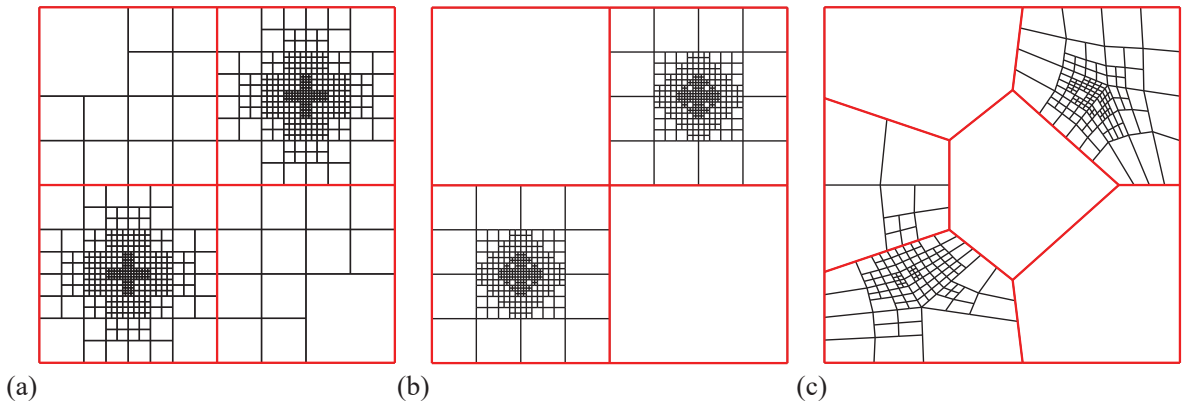

Figure 3.12: Three h-adaptively refined meshes (the red line represents the initial mesh) to solve the problem associated to (3.5): (a) quadrilaterals and hanging nodes with the one-level approach, (b) quadrilateral elements with no hanging nodes mechanism, (c) general polygonal mesh with no hanging nodes mechanism

Plots of the numerical solution using the mesh in Figure 3.12(c) using sixth order trial shape functions, $\Delta p = 2$, and 8 iterations of h-adaptive refinement are provided in Figure 3.13. The error convergence curves for the three meshes are displayed in Figure 3.14. Notice the similarity of the three resulting curves, allowing

to appreciate how the proposed method matches the one-level approach avoiding the use of a mechanism to deal with hanging nodes.

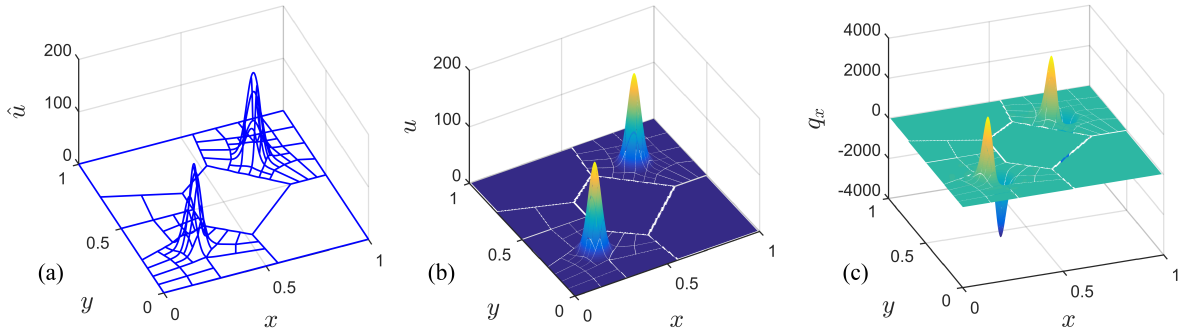

Figure 3.13: Numerical results arising from the manufactured problem in (3.5) on the meshes from Figure 3.12(c) using $p = 6$ and $\Delta p = 2$: (a) skeleton-trace of the temperature, (b) temperature, (c) first component of the heat flow.

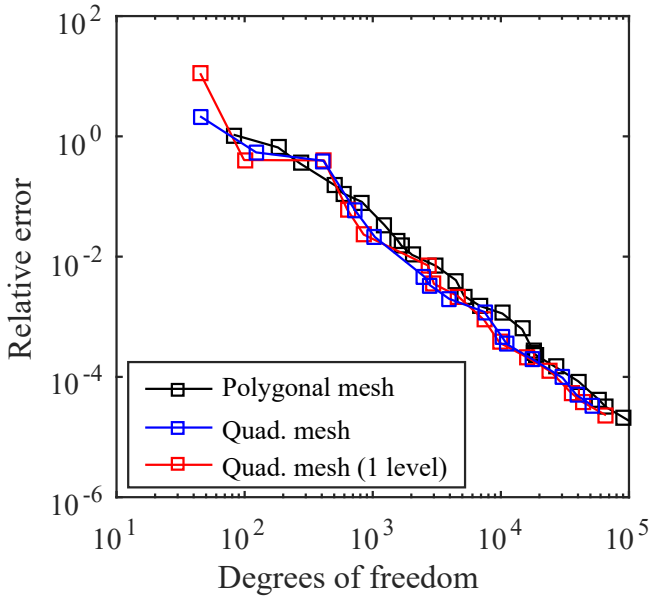

Figure 3.14: Relative error as a function of degrees of freedom. The h-convergence behavior is displayed for a single p, but for different refinement strategies.

4 Conclusions

An ultraweak formulation was successfully implemented for general polygonal elements using the DPG methodology. The method is naturally high-order, carries its own residual-based a posteriori error estimator, and has no need of ad hoc stabilization terms, despite being posed very generally (in terms of being able to discretize a large family of equations). Technically speaking, it is a conforming finite element method. As with many other polytopal methods, the spaces are defined directly in the physical space, along with the

integration schemes. Due to the advantages of ultraweak formulations in terms of avoiding inter-element compatibility conditions, it is relatively straightforward to define many of the shape functions, by simply defining them through restrictions from a bounding element (triangle or quadrilateral) to the polygonal element itself. Moreover, under reasonable assumptions, a rigorous proof posing the convergence of the numerical method is included.

Different illustrative examples corroborated the expected results. In the first example, convex polygons with multiple sides (more than 4) were considered, while in the second example, highly distorted concave polygonal elements were examined. In both cases convergence rates of the form h^p were observed for different p, as predicted by the theory. The third example is relevant in the field of geosciences, were faults separating completely different material properties are observed. With this in mind an arbitrary interface cutting a uniform grid at an angle was considered, thus creating pentagons and triangles, and different material properties were assigned at each side. Once again, displaying its robustness in resolving heterogeneous material properties, the method converged as expected with one caveat: for higher values of p, extremely small elements adjacent to significantly larger elements may produce error in conditioning, so in these rare cases it is advisable to either collapse or relocate some nodes accordingly.

The last example explored a polygonal adaptivity scheme driven by the method's arbitrary p a posteriori error estimator. This is important as these algorithms have applications in crack propagation and topology optimization. Moreover, as with many other polytopal methods, the ease to place and implement hanging nodes is a major advantage over methods with standard element shapes. A comparison with standard element methods showed that the computational cost is similar, yet polygonal adaptive methods can handle multiple levels of hanging nodes which appear naturally in problems of interest. Thus, it would be interesting to explore more detailed applications of adaptive polygonal methods in the near future.

Finally, the extension to 3D problems will be investigated in the future. In principle, the current numerical method can extend naturally to polyhedral elements, as long as all the faces are triangular, but the case of arbitrary faces is much more challenging and might lead to analyzing nonconforming numerical methods.

Acknowledgments. This work was partially supported with grants by ONR (N00014-15-1-2496), NSF (DMS-1418822), and AFOSR (FA9550-12-1-0484).

A Convergence

A.1 Stability and Fortin operators

Since the numerical method is technically a conforming finite element method, the "standard" theory of convergence can be applied. Before doing so, the issue of numerical stability, in the sense of (2.9), must be addressed. The DPG methodology is basically crafted to "almost" satisfy this condition, and intuitively, the larger the enriched test space $\mathcal{V}_r \subseteq \mathcal{V}$, the more certainty there is that the condition is satisfied. This translates to increasing Δp_K for all $K \in \mathcal{T}$ in (2.20), so that $\mathcal{V}_r$ becomes larger, but has the downside of increasing the local cost of the numerical method. In practice, the numerical stability is observed, even with very modest values of Δp.

However, to have a rigorous result, it is necessary to establish (2.9) theoretically. To do so, it is helpful to consider a linear and continuous Fortin operator, $\Pi_F : \mathcal{V} \to \mathcal{V}_r$, satisfying the orthogonality condition,

$b(\mathfrak{u}_h, \mathfrak{v} - \Pi_F \mathfrak{v}) = 0$, for all $\mathfrak{u}_h \in \mathcal{U}_h$ and $\mathfrak{v} \in \mathcal{V}$. If it exists, it follows that [50],

$$\|\mathfrak{u} - \mathfrak{u}_h\|_{\mathcal{U}} \leq \frac{\|b\|\, M_F}{\gamma} \inf_{\delta\mathfrak{u}_h \in U_h} \|\mathfrak{u} - \delta\mathfrak{u}_h\|_{\mathcal{U}}\,, \tag{A.1}$$

where $M_F \geq \|\Pi_F\| = \sup_{\mathfrak{v}\in\mathcal{V}} \frac{\|\Pi_F \mathfrak{v}\|_{\mathcal{V}}}{\|\mathfrak{v}\|_{\mathcal{V}}}$, $\|b\| = \sup_{(\mathfrak{u},\mathfrak{v})\in\mathcal{U}\times\mathcal{V}} \frac{|b(\mathfrak{u},\mathfrak{v})|}{\|\mathfrak{u}\|_{\mathcal{U}}\|\mathfrak{v}\|_{\mathcal{V}}}$ and $\gamma = \inf_{\mathfrak{u}\in\mathcal{U}} \sup_{\mathfrak{v}\in\mathcal{V}} \frac{|b(\mathfrak{u},\mathfrak{v})|}{\|\mathfrak{u}\|_{\mathcal{U}}\|\mathfrak{v}\|_{\mathcal{V}}}$, where the infima and suprema are tacitly assumed to be taken over nonzero elements. Note that when $\mathcal{V}$ is a broken test space, as in this case, the Fortin operator can be separately constructed locally at each element $K \in \mathcal{T}$. Constructions of such Fortin operators do exist for triangles [19], but have not yet been constructed for other shapes. Nevertheless, numerical results show it is reasonable to expect them to exist, and this will be assumed from now on. In any case, note that Fortin operators merely yield a conservative estimate, but in practice the results are better (i.e. instead of M_F, there is a "not very large" constant, $\mathcal{O}(1)$-$\mathcal{O}(10)$, multiplying $\frac{\|b\|}{\gamma}$ in (A.1)).

A.2 Fractional spaces

For a given $s \geq 0$ and any Lipschitz domain $A \subseteq \mathbb{R}^2$, the fractional Sobolev spaces, $H^{1+s}(A)$, $\boldsymbol{H}^s(\mathrm{div}, A)$ and $H^s(A)$, are slightly smoother subspaces of $H^1(A)$, $\boldsymbol{H}(\mathrm{div}, A) = \boldsymbol{H}^0(\mathrm{div}, A)$ and $L^2(A) = H^0(A)$ respectively. As an obvious placeholder, let $\mathcal{W}(A) = \mathcal{W}^0(A)$ be one of these spaces, and for $s \geq 0$, let $\mathcal{W}^s(A)$ be its fractional slightly smoother counterpart. As one might expect, $\|w\|_{\mathcal{W}(A)} \leq \|w\|_{\mathcal{W}^{s_1}(A)} \leq \|w\|_{\mathcal{W}^{s_2}(A)}$ for $0 \leq s_1 \leq s_2$ and all $w \in \mathcal{W}^{s_2}(A) \subseteq \mathcal{W}^{s_1}(A) \subseteq \mathcal{W}(A)$. For more details on these spaces and their norms, see [58].

Using interpolation theory (see [58, Appendix B]) applied to the universal extension operators of Sobolev spaces of differential forms defined in [52] (which is even more general than the universal extension operator defined by Stein in [68]), it is possible to establish the existence of a continuous extension operator,

$$E : \mathcal{W}^s(\Omega) \to \mathcal{W}^s(\mathbb{R}^2)\,, \qquad \|Ew\|_{\mathcal{W}^s(\mathbb{R}^2)} \leq C_E \|w\|_{\mathcal{W}^s(\Omega)}\,, \tag{A.2}$$

where $s \geq 0$, Ω is the domain where the equations are being solved, and $C_E = C_E(s, \Omega) > 0$.

The fractional skeleton spaces are better defined directly through fractional trace operators of Lipschitz elements $K \in \mathcal{T}$ (see [58]) as $H^{1/2+s}(\partial K) = \mathrm{tr}_{H^{1+s}(K)}(H^{1+s}(K))$ and $H^{-1/2+s}(\partial K) = \mathrm{tr}_{\boldsymbol{H}^s(\mathrm{div},K)}(\boldsymbol{H}^s(\mathrm{div}, K))$ (see (2.18) for the explicit trace operators in the $s = 0$ case). Again, using placeholders these are written as $\mathcal{W}^s(\partial K) = \{\hat{w}_K = \mathrm{tr}_{\mathcal{W}^s(K)} w \mid w \in \mathcal{W}^s(K)\}$, so that their minimum energy extension norm is

$$\|\hat{w}_K\|_{\mathcal{W}^s(\partial K)} = \inf_{w \in \mathrm{tr}^{-1}_{\mathcal{W}^s(K)}\{\hat{w}_K\}} \|w\|_{\mathcal{W}^s(K)}\,. \tag{A.3}$$

At a global level of the mesh, define the global trace operators as

$$\mathrm{tr}_{\mathcal{W}^s(\mathcal{T})} : \mathcal{W}^s(\Omega) \to \prod_{K\in\mathcal{T}} \mathcal{W}^s(\partial K)\,, \qquad \mathrm{tr}_{\mathcal{W}^s(\mathcal{T})} w = \prod_{K\in\mathcal{T}} \mathrm{tr}_{\mathcal{W}^s(K)}(w|_K)\,. \tag{A.4}$$

Note that $H^{1+s}_0(\Omega) = \overline{C^\infty_0(\Omega)}^{\|\cdot\|_{H^{1+s}(\Omega)}}$, so that the global fractional skeleton spaces are (see (2.5) for the $s = 0$ case),

$$H^{1/2+s}_0(\partial\mathcal{T}) = \mathrm{tr}_{H^{1+s}(\mathcal{T})}\big(H^{1+s}_0(\Omega)\big)\,, \qquad H^{-1/2+s}(\partial\mathcal{T}) = \mathrm{tr}_{\boldsymbol{H}^s(\mathrm{div},\mathcal{T})}\big(\boldsymbol{H}^s(\mathrm{div}, \Omega)\big)\,. \tag{A.5}$$

Analogous to (2.6), the fractional trial subspace for $s \geq 0$ is

$$\mathcal{U}^s = H^s(\Omega) \times (H^s(\Omega))^2 \times H^{1/2+s}_0(\partial\mathcal{T}) \times H^{-1/2+s}(\partial\mathcal{T}) \subseteq \mathcal{U}\,, \tag{A.6}$$

and it is easy to see $\|\mathfrak{u}\|_{\mathcal{U}} \leq \|\mathfrak{u}\|_{\mathcal{U}^{s_1}} \leq \|\mathfrak{u}\|_{\mathcal{U}^{s_2}}$ for $0 \leq s_1 \leq s_2$ and all $\mathfrak{u} \in \mathcal{U}^{s_2} \subseteq \mathcal{U}^{s_1} \subseteq \mathcal{U}$.

A.3 Approximation properties

Next, for a bounded Lipschitz domain $A \subseteq \mathbb{R}^2$ and polynomial order $p \in \mathbb{N}$, consider commuting exact sequence discretizations for $H^1(A)$, $\boldsymbol{H}(\mathrm{div}, A)$ and $L^2(A)$, such that

$$\begin{array}{ccccc} H^1(A) & & \boldsymbol{H}(\mathrm{div}, A) & & L^2(A) \\ \cup\mathrm{l} & & \cup\mathrm{l} & & \cup\mathrm{l} \\ H^1_{hp}(A) & \xrightarrow{\mathrm{curl}} & \boldsymbol{H}_{hp}(\mathrm{div}, A) & \xrightarrow{\nabla\cdot} & L^2_{hp}(A) \\ \cup\mathrm{l} & & \cup\mathrm{l} & & \cup\mathrm{l} \\ \mathcal{P}^p(A) & & (\mathcal{P}^{p-1}(A))^2 & & \mathcal{P}^{p-1}(A)\,. \end{array} \tag{A.7}$$

More abstractly, the discretizations are written as $\mathcal{W}_{hp}(A) \subseteq \mathcal{W}(A)$. Then, given the polynomials contained in the discretizations $\mathcal{W}_{hp}(A)$, it is well known that for $p \geq s > 0$, there exists a constant $C_h = C_h(A, s) > 0$, such that for all $w \in \mathcal{W}^s(A)$,

$$\inf_{\delta w \in \mathcal{W}_{hp}(A)} \|w - \delta w\|_{\mathcal{W}^s(A)} \leq C_h \|w\|_{\mathcal{W}^s(A)}\,. \tag{A.8}$$

For each $K \in \mathcal{T}$, the local trace spaces are supposed to be $\mathcal{W}_{hp}(\partial K) = \mathrm{tr}_{\mathcal{W}^0(K)}(\mathcal{W}_{hp}(K))$ for some $\mathcal{W}_{hp}(K)$.

We would like these approximation properties to hold for our choices of discrete trial spaces, $\mathcal{U}_h$ in (2.19), and this is indeed the case. The first two components of $\mathcal{U}_h$ when restricted to $K \in \mathcal{T}$ (representing $L^2_{hp}(K)$ and $(L^2_{hp}(K))^2$) are restrictions to K of $\mathcal{P}^{p-1}(T_K)$ and $(\mathcal{P}^{p-1}(T_K))^2$ respectively, so they do trivially contain $\mathcal{P}^{p-1}(K)$ and $(\mathcal{P}^{p-1}(K))^2$ respectively. This means (A.8) holds for those two spaces, but as we will see soon, it suffices (and is preferable) to have this result for the bounding triangle T_K (which is obviously true). For the third and fourth components of $\mathcal{U}_h$, representing the skeleton variables, locally at each $K \in \mathcal{T}$ it suffices to show that $\mathcal{P}^p_C(\partial K) = \mathrm{tr}_{H^1(K)}(H^1_{hp}(K))$ and $\mathcal{P}^{p-1}(\partial K) = \mathrm{tr}_{\boldsymbol{H}(\mathrm{div},K)}(\boldsymbol{H}_{hp}(\mathrm{div}, K))$ for some $H^1_{hp}(K)$ and $\boldsymbol{H}_{hp}(\mathrm{div}, K)$ satisfying the properties in (A.7), where $\mathcal{P}^p_C(\partial K)$ and $\mathcal{P}^{p-1}(\partial K)$ are defined in (2.18). For this, consider the shape-regular edge-compatible triangulations of each $K \in \mathcal{T}$, denoted by $\mathcal{T}(K) = \{\mathcal{T}_i(K)\}_{i \in I_K}$ (with I_K finite), and define the spaces,

$$\begin{aligned} H^1_{hp}(K) &= \{u \in H^1(K) \mid u|_{\mathcal{T}_i(K)} \in \mathcal{P}^p(\mathcal{T}_i(K)), \forall i \in I_K\}\,, \\ \boldsymbol{H}_{hp}(\mathrm{div}, K) &= \{\boldsymbol{q} \in \boldsymbol{H}(\mathrm{div}, K) \mid \boldsymbol{q}|_{\mathcal{T}_i(K)} \in \mathcal{RT}^p(\mathcal{T}_i(K)), \forall i \in I_K\}\,. \end{aligned} \tag{A.9}$$

It can easily be checked that $\hat{u}_K = u|_{\partial K} \in \mathcal{P}^p_C(\partial K)$ and $(\hat{q}_{\hat{\mathbf{n}}})_K = \boldsymbol{q}|_{\partial K} \cdot \hat{\mathbf{n}}_K \in \mathcal{P}^{p-1}(\partial K)$ for all $u \in H^1_{hp}(K)$ and $\boldsymbol{q} \in \boldsymbol{H}_{hp}(\mathrm{div}, K)$, and that these inclusions are surjective. Thus, $\mathcal{P}^p_C(\partial K) = \mathrm{tr}_{H^1(K)}(H^1_{hp}(K))$ and $\mathcal{P}^{p-1}(\partial K) = \mathrm{tr}_{\boldsymbol{H}(\mathrm{div},K)}(\boldsymbol{H}_{hp}(\mathrm{div}, K))$ as desired. This implies that (A.8) also holds for $H^1_{hp}(K)$ and $\boldsymbol{H}_{hp}(\mathrm{div}, K)$, which are closely related to the skeleton discretizations of $\mathcal{U}_h$.

A.4 Interpolation estimates

The idea is to define a bounded linear interpolation operator $\Pi_{\mathcal{U}^s} : \mathcal{U}^s \to \mathcal{U}_h$ such that $\Pi_{\mathcal{U}^s} \mathfrak{u}_h = \mathfrak{u}_h$ for every $\mathfrak{u}_h \in \mathcal{U}_h$ and $s > \frac{1}{2}$. Typically this implies constructing interpolation operators for every component of $\mathcal{U}$, and moreover, for each component this construction is done locally at every $K \in \mathcal{T}$ in such a way that the inter-element compatibility properties are satisfied.

The first two components of $\mathcal{U}$ are $L^2(\Omega)$ and $\boldsymbol{L}^2(\Omega)$, which are effectively three components $L^2(\Omega)$. The last two skeleton components are $H_0^{1/2+s}(\partial\mathcal{T})$ and $H^{-1/2+s}(\partial\mathcal{T})$. The discretizations of these three spaces

are (see (2.19)),

$$\begin{aligned}
L^2_{hp}(\Omega) &= \left\{u \in L^2(\Omega) \mid u|_K \in \mathcal{P}^{p-1}(K), \forall K \in \mathcal{T}\right\},\\
H^{1/2}_{0,hp}(\partial\mathcal{T}) &= \left\{\hat{u} \in H^{1/2}_0(\partial\mathcal{T}) \mid \hat{u}_K \in \mathcal{P}^p_C(\partial K) = \mathrm{tr}_{H^1(K)}(H^1_{hp}(K)), \forall K \in \mathcal{T}\right\},\\
H^{-1/2}_{hp}(\partial\mathcal{T}) &= \left\{\hat{q}_{\hat{\mathbf{n}}} \in H^{-1/2}(\partial\mathcal{T}) \mid (\hat{q}_{\hat{\mathbf{n}}})_K \in \mathcal{P}^{p-1}(\partial K) = \mathrm{tr}_{\boldsymbol{H}(\mathrm{div},K)}(\boldsymbol{H}_{hp}(\mathrm{div},K)), \forall K \in \mathcal{T}\right\},
\end{aligned} \tag{A.10}$$

where the definitions of $H^1_{hp}(K)$ and $\boldsymbol{H}_{hp}(\mathrm{div},K)$ are in (A.9). Thus, it suffices to construct

$$\begin{aligned}
\Pi_{H^s(\Omega)} &: H^s(\Omega) \to L^2_{hp}(\Omega), \qquad \left(\Pi_{H^s(\Omega)}u\right)\big|_K = \Pi_{H^s(K)}u|_K,\\
\Pi_{H^{1/2+s}_0(\partial\mathcal{T})} &: H^{1/2+s}_0(\partial\mathcal{T}) \to H^{1/2}_{0,hp}(\partial\mathcal{T}), \qquad \left(\Pi_{H^{1/2+s}_0(\partial\mathcal{T})}\hat{u}\right)_K = \Pi_{H^{1/2+s}(\partial K)}\hat{u}_K,\\
\Pi_{H^{-1/2+s}(\partial\mathcal{T})} &: H^{-1/2+s}(\partial\mathcal{T}) \to H^{-1/2}_{hp}(\partial\mathcal{T}), \qquad \left(\Pi_{H^{-1/2+s}(\partial\mathcal{T})}\hat{q}_{\hat{\mathbf{n}}}\right)_K = \Pi_{H^{-1/2+s}(\partial K)}(\hat{q}_{\hat{\mathbf{n}}})_K,
\end{aligned} \tag{A.11}$$

meaning we must define $\Pi_{H^s(K)}$, $\Pi_{H^{1/2+s}(\partial K)}$ and $\Pi_{H^{-1/2+s}(\partial K)}$.

In the case of $\Pi_{H^s(K)}$ it can be chosen as the $L^2(K)$-projection to $\mathcal{P}^{p-1}(K)$ directly on K (so $\Pi_{H^s(K)}\delta u = \delta u$ for $\delta u \in \mathcal{P}^{p-1}(K)$). Consider now a simple scaling by $h_K = \mathrm{diam}(K)$, so that $\hat{K}$ has $\mathrm{diam}(\hat{K}) = 1$. Using (A.8) for $p \geq s > \frac{1}{2}$ results in (written abstractly),

$$\begin{aligned}
\|w - \Pi_{\mathcal{W}^s(\hat{K})}w\|_{\mathcal{W}(\hat{K})} &= \inf_{\delta w \in \mathcal{W}_{hp}(\hat{K})} \|(I - \Pi_{\mathcal{W}^s(\hat{K})})(w - \delta w)\|_{\mathcal{W}(\hat{K})}\\
&\leq \|I - \Pi_{\mathcal{W}^s(\hat{K})}\| \inf_{\delta w \in \mathcal{W}_{hp}(\hat{K})} \|(w - \delta w)\|_{\mathcal{W}^s(\hat{K})} \leq C_{\hat{K}}\|w\|_{\mathcal{W}^s(\hat{K})},
\end{aligned} \tag{A.12}$$

for any $w \in \mathcal{W}^s(\hat{K})$, where $C_{\hat{K}} = C_{\hat{K}}(\hat{K},p,s) > 0$. Scaling appropriately then yields for any $w \in \mathcal{W}^s(K)$,

$$\|w - \Pi_{\mathcal{W}^s(K)}w\|_{\mathcal{W}(K)} \leq C_{\hat{K}}h^s_K\|w\|_{\mathcal{W}^s(K)}. \tag{A.13}$$

The issue with this estimate is that it depends on the element shape K (via $\hat{K}$), so it is inconvenient as it may become much larger with mesh refinements. The solution is to use the bounding triangle and the extension operator defined in (A.2), so that (as in [18]) the interpolation operator is defined for any $w \in \mathcal{W}^s(\Omega)$ as,

$$\Pi_{\mathcal{W}^s(K)}w|_K = \left(\Pi_{\mathcal{W}^s(T_K)}Ew|_{T_K}\right)\big|_K, \tag{A.14}$$

where $\Pi_{\mathcal{W}^s(T_K)}$ is the $L^2(T_K)$-projection. Scaling and rotating transforms the bounding triangle T_K to a unique triangle $\hat{T}_0$ (independent of the element K) with $\mathrm{diam}(\hat{T}_0) = 1$. This means T_K is scaled by $h_{T_K} = \mathrm{diam}(T_K) = \frac{6}{\sqrt{3}}r_{\max} \leq \frac{6}{\sqrt{3}}h_K$, where $r_{\max}$ is the distance of the centroid to the furthest vertex and $h_K = \mathrm{diam}(K)$. Using the same reasoning gives,

$$\|w - \Pi_{\mathcal{W}^s(K)}w\|_{\mathcal{W}(K)} \leq \|Ew - \Pi_{\mathcal{W}^s(T_K)}Ew\|_{\mathcal{W}(T_K)} \leq C_{\hat{T}_0}h^s_K\|Ew\|_{\mathcal{W}^s(T_K)}, \tag{A.15}$$

for every $w \in \mathcal{W}^s(K)$, where $C_{\hat{T}_0} = C_{\hat{T}_0}(p,s) > 0$ is now independent of K.

Next, consider the skeleton variables for an element $K \in \mathcal{T}$ and its respective shape-regular edge-compatible triangulation denoted by $\mathfrak{T}(K) = \{\mathcal{T}_i(K)\}_{i\in I_K}$. The theory of projection-based interpolation [31] implies that for any polygonal domain $A \subseteq \mathbb{R}^2$ and for any $s > \frac{1}{2}$ there exist commuting operators,

$$\begin{array}{ccccc}
H^{1+s}(A) & \xrightarrow{\mathrm{curl}} & \boldsymbol{H}^s(\mathrm{div},A) & \xrightarrow{\nabla\cdot} & H^s(A)\\
\downarrow{\scriptstyle \Pi_{H^{1+s}(A)}} & & \downarrow{\scriptstyle \Pi_{\boldsymbol{H}^s(\mathrm{div},A)}} & & \downarrow{\scriptstyle \Pi_{H^s(A)}}\\
H^1_{hp}(A) & \xrightarrow{\mathrm{curl}} & \boldsymbol{H}_{hp}(\mathrm{div},A) & \xrightarrow{\nabla\cdot} & L^2_{hp}(A).
\end{array} \tag{A.16}$$

Thus, for any $p \geq s > \frac{1}{2}$ and triangle $\mathfrak{T}_i(K)$ (so $\operatorname{diam}(\mathfrak{T}_i(K)) \leq \operatorname{diam}(K) = h_K$), the result in (A.13) applies and yields

$$\|w - \Pi_{\mathcal{W}^s(\mathfrak{T}_i(K))} w\|_{\mathcal{W}(\mathfrak{T}_i(K))} \leq C_{\hat{\mathfrak{T}}_0} h_K^s \|w\|_{\mathcal{W}^s(\mathfrak{T}_i(K))}\,, \tag{A.17}$$

where the K-independent $C_{\hat{\mathfrak{T}}_0} = C_{\hat{\mathfrak{T}}_0}(p,s) > 0$ exists due to the *assumed* uniformity of the shape-regularity of the $\mathfrak{T}_i(K)$ (across all $K \in \mathcal{T}$ and all meshes being considered). Adding among $\mathfrak{T}(K)$ is valid due to the compatibility of the projection-based interpolation in the triangulation, so that

$$\|w - \Pi_{\mathcal{W}^s(K)} w\|^2_{\mathcal{W}(K)} = \sum_{i \in I_K} \|w - \Pi_{\mathcal{W}^s(\mathfrak{T}_i(K))} w\|^2_{\mathcal{W}(\mathfrak{T}_i(K))} \leq C^2_{\hat{\mathfrak{T}}_0} h_K^{2s} \|w\|^2_{\mathcal{W}^s(K)}\,. \tag{A.18}$$

Lastly, consider the well-defined trace interpolation,

$$\Pi_{\mathcal{W}^s(\partial K)} \hat{w}_K = \operatorname{tr}_{\mathcal{W}(K)} \Pi_{\mathcal{W}^s(K)} w\,, \qquad w \in \operatorname{tr}^{-1}_{\mathcal{W}^s(K)} \{\hat{w}_K\}\,, \tag{A.19}$$

so that (see (A.3)),

$$\begin{aligned} \|\hat{w}_K - \Pi_{\mathcal{W}^s(\partial K)} \hat{w}_K\|_{\mathcal{W}(\partial K)} &= \|\operatorname{tr}_{\mathcal{W}(K)}(w - \Pi_{\mathcal{W}^s(K)} w)\|_{\mathcal{W}(\partial K)} \\ &\leq \|w - \Pi_{\mathcal{W}^s(K)} w\|_{\mathcal{W}(K)} \leq C_{\hat{\mathfrak{T}}_0} h_K^s \|w\|_{\mathcal{W}^s(K)}\,. \end{aligned} \tag{A.20}$$

This is true for every $w \in \operatorname{tr}^{-1}_{\mathcal{W}^s(K)} \hat{w}_K$, so take the infimum to yield

$$\|\hat{w}_K - \Pi_{\mathcal{W}^s(\partial K)} \hat{w}_K\|_{\mathcal{W}(\partial K)} \leq C_{\hat{\mathfrak{T}}_0} h_K^s \|\hat{w}_K\|_{\mathcal{W}^s(\partial K)}\,. \tag{A.21}$$

Putting everything together and generalizing for any $p \in \mathbb{N}$ and $s > \frac{1}{2}$, gives

$$\begin{aligned} \|u - \Pi_{H^s(K)} u\|_{L^2(K)} &\leq C_{\hat{T}_0} h_K^{\min\{s,p\}} \|Eu\|_{H^s(T_K)}\,, \\ \|\boldsymbol{q} - \Pi_{(H^s(K))^2} \boldsymbol{q}\|_{\boldsymbol{L}^2(K)} &\leq C_{\hat{T}_0} h_K^{\min\{s,p\}} \|E\boldsymbol{q}\|_{(H^s(T_K))^2}\,, \\ \|\hat{u}_K - \Pi_{H^{1/2+s}(\partial K)} \hat{u}_K\|_{H^{1/2}(\partial K)} &\leq C_{H^{1+s}(\hat{\mathfrak{T}}_0)} h_K^{\min\{s,p\}} \|\hat{u}_K\|_{H^{1/2+s}(\partial K)}\,, \\ \|(\hat{q}_{\hat{\mathbf{n}}})_K - \Pi_{H^{-1/2+s}(\partial K)} (\hat{q}_{\hat{\mathbf{n}}})_K\|_{H^{-1/2}(\partial K)} &\leq C_{\boldsymbol{H}^s(\operatorname{div},\hat{\mathfrak{T}}_0)} h_K^{\min\{s,p\}} \|(\hat{q}_{\hat{\mathbf{n}}})_K\|_{H^{-1/2+s}(\partial K)}\,, \end{aligned} \tag{A.22}$$

where the constants $C_{\hat{T}_0}$, $C_{H^{1+s}(\hat{\mathfrak{T}}_0)}$ and $C_{\boldsymbol{H}^s(\operatorname{div},\hat{\mathfrak{T}}_0)}$ are only dependent on p and s, but not on K (the last two constants depend on the uniform shape-regularity of the edge-compatible triangulations of all elements). Finally, since these constants come from triangles, the theory of projection-based interpolation [31] and the results in [4] (applied to $C_{H^{1+s}(\hat{\mathfrak{T}}_0)}$) imply that in the p-asymptotic limit,

$$C_{\hat{T}_0} = \widetilde{C}_{\hat{T}_0} (\ln p) p^{-s}\,, \qquad C_{H^{1+s}(\hat{\mathfrak{T}}_0)} = \widetilde{C}_{H^{1+s}(\hat{\mathfrak{T}}_0)} p^{-s}\,, \qquad C_{\boldsymbol{H}^s(\operatorname{div},\hat{\mathfrak{T}}_0)} = \widetilde{C}_{\boldsymbol{H}^s(\operatorname{div},\hat{\mathfrak{T}}_0)} (\ln p) p^{-s}\,, \tag{A.23}$$

where $\widetilde{C}_{\hat{T}_0}$, $\widetilde{C}_{H^{1+s}(\hat{\mathfrak{T}}_0)}$ and $\widetilde{C}_{\boldsymbol{H}^s(\operatorname{div},\hat{\mathfrak{T}}_0)}$ are constants independent of p and of any $K \in \mathcal{T}$ across all possible meshes being considered.

A.5 Final convergence estimates

Use the global interpolation operators in (A.11) to construct the bounded linear global interpolation operator $\Pi_{\mathcal{U}^s} : \mathcal{U}^s \to \mathcal{U}_h$. Note that adding (A.22) among $K \in \mathcal{T}$ and using the robust finite overlap condition gives

$$\begin{aligned} \|u - \Pi_{H^s(K)} u\|^2_{L^2(K)} &\leq C^2_{\hat{T}_0} \sum_{K \in \mathcal{T}} h_K^{2\min\{s,p\}} \|Eu\|^2_{H^s(T_K)} \\ &\leq M_{\mathrm{ov}} C^2_{\hat{T}_0} h^{2\min\{s,p\}} \|Eu\|^2_{H^s(\mathbb{R}^2)} \leq C_E M_{\mathrm{ov}} C^2_{\hat{T}_0} h^{2\min\{s,p\}} \|u\|^2_{H^s(\Omega)}\,, \end{aligned} \tag{A.24}$$

where $h = \sup_{K\in\mathcal{T}} h_K$ and $C_E = C_E(s,\Omega)$ is not dependent on p. Then, assume M_F is independent of $K \in \mathcal{T}$ among the meshes being considered, and choose the interpolant in (A.1) along with the estimates in (A.22), so that

$$\|\mathfrak{u} - \mathfrak{u}_h\|_{\mathcal{U}} \leq \frac{\|b\|\, M_F}{\gamma} \|\mathfrak{u} - \Pi_{\mathcal{U}^s}\mathfrak{u}\|_{\mathcal{U}} \leq C h^{\min\{s,p\}} \|\mathfrak{u}\|_{\mathcal{U}^s}\,, \tag{A.25}$$

where $C = C(p,s,\Omega) > 0$, but is independent of the meshes being considered. Moreover, if M_F is p-independent, then in the p-asymptotic limit it follows that

$$\|\mathfrak{u} - \mathfrak{u}_h\|_{\mathcal{U}} \leq \widetilde{C}(\ln p)\frac{h^{\min\{s,p\}}}{p^s} \|\mathfrak{u}\|_{\mathcal{U}^s}\,, \tag{A.26}$$

where $\widetilde{C} = \widetilde{C}(s,\Omega)$ is independent of p. This concludes the results summarized in Theorem 2.1.

References

[1] Ainsworth, M., Davydov, O., and Schumaker, L. L. (2016). Bernstein-Bézier finite elements on tetrahedral–hexahedral–pyramidal partitions. *Comput. Methods Appl. Mech. Engrg.*, 304:140–170.

[2] Arbogast, T. and Correa, M. R. (2016). Two families of $H(\mathrm{div})$ mixed finite elements on quadrilaterals of minimal dimension. *SIAM J. Numer. Anal.*, 54(6):3332–3356.

[3] Arnold, D. N., Falk, R. S., and Winther, R. (2006). Finite element exterior calculus, homological techniques, and applications. *Acta Numer.*, 15:1–155.

[4] Babuška, I. and Suri, M. (1987). The *hp* version of the finite element method with quasiuniform meshes. *RAIRO Modél. Math. Anal. Numér.*, 21(2):199–238.

[5] Beirão da Veiga, L., Dassi, F., and Russo, A. (2017). High-order Virtual Element Method on polyhedral meshes. *ArXiv e-prints*, arXiv:1703.02882 [math.NA].

[6] Beirão da Veiga, L., Brezzi, F., Cangiani, A., Manzini, G., Marini, L. D., and Russo, A. (2013a). Basic principles of virtual element methods. *Math. Models Methods Appl. Sci.*, 23(01):199–214.

[7] Beirão da Veiga, L., Brezzi, F., and Marini, L. D. (2013b). Virtual elements for linear elasticity problems. *SIAM J. Numer. Anal.*, 51(2):794–812.

[8] Beirão da Veiga, L., Brezzi, F., Marini, L. D., and Russo, A. (2016a). $H(\mathrm{div})$ and $H(\mathrm{curl})$-conforming virtual element methods. *Numer. Math.*, 133(2):303–332.

[9] Beirão da Veiga, L., Brezzi, F., Marini, L. D., and Russo, A. (2016b). Virtual element method for general second-order elliptic problems on polygonal meshes. *Math. Models Methods Appl. Sci.*, 26(04):729–750.

[10] Benedetto, M. F., Berrone, S., Pieraccini, S., and Scialò, S. (2014). The virtual element method for discrete fracture network simulations. *Comput. Methods Appl. Mech. Engrg.*, 280:135–156.

[11] Bishop, J. E. (2009). Simulating the pervasive fracture of materials and structures using randomly close packed Voronoi tessellations. *Comput. Mech.*, 44(4):455–471.

[12] Bishop, J. E. (2014). A displacement-based finite element formulation for general polyhedra using harmonic shape functions. *Int. J. Num. Meth. in Eng.*, 97(1).

[13] Bishop, J. E., Martinez, M. J., and Newell, P. (2016). Simulating fragmentation and fluid-induced fracture in disordered media using random finite-element meshes. *Int. J. Multiscale Comput. Eng.*, 14(4).

[14] Bramwell, J., Demkowicz, L., Gopalakrishnan, J., and Weifeng, Q. (2012). A locking-free *hp* DPG method for linear elasticity with symmetric stresses. *Numer. Math.*, 122(4):671–707.

[15] Brezzi, F., Lipnikov, K., and Shashkov, M. (2005a). Convergence of the mimetic finite difference method for diffusion problems on polyhedral meshes. *SIAM J. Numer. Anal.*, 43(5):1872–1896.

[16] Brezzi, F., Lipnikov, K., and Simoncini, V. (2005b). A family of mimetic finite difference methods on polygonal and polyhedral meshes. *Math. Models Methods Appl. Sci.*, 15(10):1533–1551.

[17] Brezzi, F. and Marini, L. D. (2013). Virtual element methods for plate bending problems. *Comput. Methods Appl. Mech. Engrg.*, 253:455–462.

[18] Cangiani, A., Georgoulis, E. H., and Houston, P. (2014). *hp*-version discontinuous Galerkin methods on polygonal and polyhedral meshes. *Math. Models Methods Appl. Sci.*, 24(10):2009–2041.

[19] Carstensen, C., Demkowicz, L., and Gopalakrishnan, J. (2016). Breaking spaces and forms for the DPG method and applications including Maxwell equations. *Comput. Math. Appl.*, 72(3):494–522.

[20] Chan, J., Demkowicz, L., and Moser, R. (2014a). A DPG method for steady viscous compressible flow. *Comput. & Fluids*, 98:69–90.

[21] Chan, J., Heuer, N., Bui-Thanh, T., and Demkowicz, L. (2014b). A robust DPG method for convection-dominated diffusion problems II: Adjoint boundary conditions and mesh-dependent test norms. *Comput. Math. Appl.*, 67(4):771–795.

[22] Chen, L., Wei, H., and Wen, M. (2017). An interface-fitted mesh generator and virtual element methods for elliptic interface problems. *J. Comput. Phys.*, 334(1):327–348.

[23] Chen, W. and Wang, Y. (2016). Minimal degree $H(\mathrm{curl})$ and $H(\mathrm{div})$ conforming finite elements on polytopal meshes. *Math. Comp.*

[24] Chi, H., Beirão da Veiga, L., and Paulino, G. H. (2017). Some basic formulations of the virtual element method (VEM) for finite deformations. *Comput. Methods Appl. Mech. Engrg.*, 318:148–192.

[25] Chi, H., Talischi, C., Lopez-Pamies, O., and H Paulino, G. (2015). Polygonal finite elements for finite elasticity. *Int. J. Num. Meth. in Eng.*, 101(4):305–328.

[26] Chin, E. B., Lasserre, J. B., and Sukumar, N. (2015). Numerical integration of homogeneous functions on convex and nonconvex polygons and polyhedra. *Comput. Mech.*, 56(6):967–981.

[27] Ciarlet, P. G. (2002). *The Finite Element Method for Elliptic Problems*, volume 40 of *Classics in Applied Mathematics*. SIAM.

[28] Cohen, A., Dahmen, W., and Welper, G. (2012). Adaptivity and variational stabilization for convection-diffusion equations. *ESAIM Math. Model. Numer. Anal.*, 46(5):1247–1273.

[29] Cueto, E., Calvo, B., and Doblaré, M. (2002). Modelling three-dimensional piece-wise homogeneous domains using the α-shape-based natural element method. *Int. J. Num. Meth. in Eng.*, 54(6):871–897.

[30] Demkowicz, L. (2006). *Computing with hp Finite Elements. I. One and Two Dimensional Elliptic and Maxwell Problems*. Chapman & Hall/CRC Press, New York.

[31] Demkowicz, L. (2008). Polynomial exact sequences and projection-based interpolation with application to Maxwell equations. In Boffi, D. and Gastaldi, L., editors, *Mixed Finite Elements, Compatibility Conditions, and Applications*, volume 1939 of *Lecture Notes in Mathematics*, pages 101–158. Springer, Berlin.

[32] Demkowicz, L. (2015). Various variational formulations and closed range theorem. ICES Report 15-03, The University of Texas at Austin.

[33] Demkowicz, L. and Gopalakrishnan, J. (2014). An overview of the DPG method. In Feng, X., Karakashian, O., and Xing, Y., editors, *Recent Developments in Discontinuous Galerkin Finite Element Methods for Partial Differential Equations*, volume 157 of *The IMA Volumes in Mathematics and its Applications*, pages 149–180. Springer.

[34] Demkowicz, L., Gopalakrishnan, J., Muga, I., and Zitelli, J. (2012a). Wavenumber explicit analysis of a DPG method for the multidimensional Helmholtz equation. *Comput. Methods Appl. Mech. Engrg.*, 213–216:126–138.

[35] Demkowicz, L., Gopalakrishnan, J., Nagaraj, S., and Sepúlveda, P. (2016). A spacetime DPG method for the Schrödinger equation. *ArXiv e-prints*, arXiv:1610.04678 [math.NA].

[36] Demkowicz, L., Gopalakrishnan, J., and Niemi, A. H. (2012b). A class of discontinuous Petrov–Galerkin methods. Part III: Adaptivity. *Appl. Numer. Math.*, 62(4):396–427.

[37] Demkowicz, L. and Heuer, N. (2013). Robust DPG method for convection-dominated diffusion problems. *SIAM J. Numer. Anal.*, 51(5):2514–2537.

[38] Demkowicz, L., Kurtz, J., Pardo, D., Paszyński, M., Rachowicz, W., and Zdunek, A. (2007). *Computing with hp Finite Elements. II. Frontiers: Three Dimensional Elliptic and Maxwell Problems with Applications*. Chapman & Hall/CRC, New York.

[39] Droniou, J., Eymard, R., Gallouët, T., Guichard, C., and Herbin, R. (2016). *The gradient discretisation method: A framework for the discretisation and numerical analysis of linear and nonlinear elliptic and parabolic problems*. HAL, hal-01382358.

[40] Ellis, T., Demkowicz, L., and Chan, J. (2014). Locally conservative discontinuous Petrov-Galerkin finite elements for fluid problems. *Comput. Math. Appl.*, 68(11):1530–1549.

[41] Fuentes, F., Demkowicz, L., and Wilder, A. (2016a). Using a DPG method to validate DMA experimental calibration of viscoelastic materials. *ArXiv e-prints*, arXiv:1703.10998 [math.NA].

[42] Fuentes, F., Keith, B., Demkowicz, L., and Le Tallec, P. (2016b). Coupled variational formulations of linear elasticity and the DPG methodology. *ArXiv e-prints*, arXiv:1609.08180 [math.NA].

[43] Fuentes, F., Keith, B., Demkowicz, L., and Nagaraj, S. (2015). Orientation embedded high order shape functions for the exact sequence elements of all shapes. *Comput. Math. Appl.*, 70(4):353–458.

[44] Führer, T. and Heuer, N. (2016). Robust coupling of DPG and BEM for a singularly perturbed transmission problem. *ArXiv e-prints*, arXiv:1603.05164 [math.NA].

[45] Gain, A. L., Paulino, G. H., Duarte, L. S., and Menezes, I. F. (2015). Topology optimization using polytopes. *Comput. Methods Appl. Mech. Engrg.*, 293:411–430.

[46] Ghosh, S. and Moorthy, S. (1995). Elastic-plastic analysis of arbitrary heterogeneous materials with the voronoi cell finite element method. *Comput. Methods Appl. Mech. Engrg.*, 121(1-4):373–409.

[47] Gillette, A. (2016). Serendipity and tensor product affine pyramid finite elements. *SMAI J. Comput. Math.*, 2:215–228.

[48] Gillette, A., Rand, A., and Bajaj, C. (2016). Construction of scalar and vector finite element families on polygonal and polyhedral meshes. *Comput. Methods Appl. Math.*, 16(4):667–683.

[49] Gopalakrishnan, J., Muga, I., and Olivares, N. (2014). Dispersive and dissipative errors in the DPG method with scaled norms for Helmholtz equation. *SIAM J. Sci. Comput.*, 36(1):A20–A39.

[50] Gopalakrishnan, J. and Qiu, W. (2014). An analysis of the practical DPG method. *Math. Comp.*, 83(286):537–552.

[51] Heuer, N. and Karkulik, M. (2015). DPG method with optimal test functions for a transmission problem. *Comput. Math. Appl.*, 70(5):1070–1081.

[52] Hiptmair, R., Li, J., and Zou, J. (2012). Universal extension for Sobolev spaces of differential forms and applications. *J. Funct. Anal.*, 263(2):364–382.

[53] Keith, B., Fuentes, F., and Demkowicz, L. (2016a). The DPG methodology applied to different variational formulations of linear elasticity. *Comput. Methods Appl. Mech. Engrg.*, 309:579–609.

[54] Keith, B., Knechtges, P., Roberts, N. V., Elgeti, S., Behr, M., and Demkowicz, L. (2016b). An ultraweak DPG method for viscoelastic fluids. *ArXiv e-prints*, arXiv:1612.03124 [math.NA].

[55] Kuznetsov, Y., Lipnikov, K., and Shashkov, M. (2004). The mimetic finite difference method on polygonal meshes for diffusion-type problems. *Computational Geosciences*, 8(4):301–324.

[56] Leon, S. E., Spring, D. W., and Paulino, G. H. (2014). Reduction in mesh bias for dynamic fracture using adaptive splitting of polygonal finite elements. *Int. J. Num. Meth. in Eng.*, 100(8):555–576.

[57] Manzini, G., Russo, A., and Sukumar, N. (2014). New perspectives on polygonal and polyhedral finite element methods. *Mathematical Models and Methods in Applied Sciences*, 24(08):1665–1699.

[58] McLean, W. (2000). *Strongly Elliptic Systems and Boundary Integral Equations*. Cambridge University Press, Cambridge.

[59] Mousavi, S. E. and Sukumar, N. (2011). Numerical integration of polynomials and discontinuous functions on irregular convex polygons and polyhedrons. *Comput. Mech.*, 47(5):535–554.

[60] Mu, L., Wang, J., and Ye, X. (2015). Weak Galerkin finite element methods on polytopal meshes. *Int. J. Numer. Anal. Model.*, 12(1):31–53.

[61] Mu, L., Wang, J., Ye, X., and Zhao, S. (2016). A new weak Galerkin finite element method for elliptic interface problems. *J. Comput. Phys.*, 325:157–173.

[62] Niemi, A. H., Collier, N., and Calo, V. M. (2013). Automatically stable discontinuous Petrov-Galerkin methods for stationary transport problems: Quasi–optimal test space norm. *Comput. Math. Appl.*, 66(10):2096–2113.

[63] Nigam, N. and Phillips, J. (2012). High-order conforming finite elements on pyramids. *IMA J. Numer. Anal.*, 32(2):448–483.

[64] Petrides, S. and Demkowicz, L. F. (2016). An adaptive DPG method for high frequency time-harmonic wave propagation problems. ICES Report 16-20, The University of Texas at Austin.

[65] Rand, A., Gillette, A., and Bajaj, C. (2014). Quadratic serendipity finite elements on polygons using generalized barycentric coordinates. *Math. Comput.*, 83(290):2691–2716.

[66] Roberts, N. V., Demkowicz, L., and Moser, R. (2015). A discontinuous Petrov–Galerkin methodology for adaptive solutions to the incompressible Navier–Stokes equations. *J. Comput. Phys.*, 301:456–483.

[67] Spring, D. W., Leon, S. E., and Paulino, G. H. (2014). Unstructured polygonal meshes with adaptive refinement for the numerical simulation of dynamic cohesive fracture. *Int. J. Fract.*, 189(1):33–57.

[68] Stein, E. M. (1970). *Singular integrals and differentiability properties of functions*. Princeton university press, Princeton.

[69] Sukumar, N. and Tabarraei, A. (2004). Conforming polygonal finite elements. *Int. J. Num. Meth. in Eng.*, 61(12):2045–2066.

[70] Tabarraei, A. and Sukumar, N. (2007). Adaptive computations using material forces and residual-based error estimators on quadtree meshes. *Comput. Methods Appl. Mech. Engrg.*, 196(25):2657–2680.

[71] Talischi, C., Paulino, G. H., Pereira, A., and Menezes, I. F. (2010). Polygonal finite elements for topology optimization: a unifying paradigm. *Int. J. Num. Meth. in Eng.*, 82(6):671–698.

[72] Talischi, C., Paulino, G. H., Pereira, A., and Menezes, I. F. (2012). Polymesher: a general-purpose mesh generator for polygonal elements written in Matlab. *Struct. Multidiscip. Optim.*, 45(3):309–328.

[73] Wachspress, E. (1975). *A Rational Finite Element Basis*, volume 114 of *Mathematics in Science and Engineering*. Academic Press.

[74] Zitelli, J., Muga, I., Demkowicz, L., Gopalakrishnan, J., Pardo, D., and Calo, V. M. (2011). A class of discontinuous Petrov–Galerkin methods. Part IV: The optimal test norm and time-harmonic wave propagation in 1d. *J. Comput. Phys.*, 230(7):2406–2432.