Discontinuous Galerkin Methods on *hp*-Anisotropic Meshes II: A Posteriori Error Analysis and Adaptivity

Emmanuil H. Georgoulis * Edward Hall † Paul Houston ‡

October 12, 2007

Abstract

We consider the *a posteriori* error analysis and *hp*-adaptation strategies for *hp*-version interior penalty discontinuous Galerkin methods for second–order partial differential equations with nonnegative characteristic form on anisotropically refined computational meshes with anisotropically enriched elemental polynomial degrees. In particular, we exploit duality based *hp*-error estimates for linear target functionals of the solution and design and implement the corresponding adaptive algorithms to ensure reliable and efficient control of the error in the prescribed functional to within a given tolerance. This involves exploiting both local isotropic and anisotropic mesh refinement and isotropic and anisotropic polynomial degree enrichment. The superiority of the proposed algorithm in comparison with standard *hp*–isotropic mesh refinement algorithms and an *h*–anisotropic/*p*–isotropic adaptive procedure is illustrated by a series of numerical experiments.

1 Introduction

Adaptive finite element methods, capable of approximating solutions to complex multi-dimensional partial differential equations with guaranteed error control, are an invaluable computational tool. In general, the construction of an adaptive strategy involves three key steps: the derivation of a sharp *a posteriori* error bound for the finite element approximation of the partial differential

*Department of Mathematics, University of Leicester, Leicester LE1 7RH, UK, email: Emmanuil.Georgoulis@mcs.le.ac.uk.

†School of Mathematical Sciences, University of Nottingham, University Park, Nottingham NG7 2RD, UK, email: Edward.Hall@nottingham.ac.uk. The research of this author was supported by the European Union under the ADIGMA project.

‡School of Mathematical Sciences, University of Nottingham, University Park, Nottingham NG7 2RD, UK, email: Paul.Houston@nottingham.ac.uk. The research of this author was supported by both the EPSRC under grant GR/R76615 and the European Union under the ADIGMA project.

equation under consideration, which is then used as a stopping criterion to terminate the adaptive algorithm once the desired level of accuracy has been achieved; the design of an appropriate refinement indicator to identify regions in the computational domain where the error is locally large; and the design of the corresponding mesh-modification/adaptive algorithm which is capable of automatically selecting the local mesh width h and/or the local degree p of the approximating polynomial in order to deliver reliable and efficient control of the discretisation error. While, in recent years, considerable progress has been made on both the *a posteriori* error analysis of finite element methods for a wide range of partial differential equations of practical interest, and the theoretical and computational assessment of local refinement indicators; see, for example, [1, 5, 10, 38, 41, 42] and the references cited therein, the state of development of "optimal" mesh modification strategies which are capable of delivering the greatest reduction in the error for the least amount of computational cost, is far less advanced.

Clearly, adaptive finite element methods that exploit both local polynomial–degree variation (p–refinement) and local mesh subdivision (h–refinement) offer much greater flexibility and improved efficiency than mesh refinement algorithms which only incorporate h–refinement or p–refinement in isolation. Since the early analytical paper of Gui and Babuška [16], the benefits of hp–version finite element methods have been clearly established for elliptic boundary value problems (see, for example, the monograph of Schwab [36]), particularly in the field of linear elasticity. The application of hp–version finite element methods to hyperbolic/nearly–hyperbolic problems is less standard, although their potential in compressible gas dynamics was first demonstrated by J.E. Flaherty and collaborators (see [6, 8], for example); for more recent work in this area, we refer to our series of papers [23, 25, 26, 39, 40]. The argument in favour of using an hp–version finite element method for the numerical solution of a hyperbolic/nearly–hyperbolic equation rests on the observation that while solutions to these equations may exhibit local singularities and discontinuities, in large parts of the computational domain the solution is typically a real analytic function. Such large variations in the smoothness of the solution can be captured in a particularly simple and flexible manner by using a finite element method based on discontinuous piecewise polynomials, such as the discontinuous Galerkin finite element method.

Typically, singularities arising in the analytical solution of partial differential equations of hyperbolic/nearly–hyperbolic type are anisotropic in character, i.e., the solution exhibits a strong variation in the direction normal to a lower–dimensional manifold, but only a small variation in the tangential direction. For example, in two space dimensions, a shock in the solution to a nonlinear hyperbolic conservation law occurs along a one–dimensional line. Thereby, adaptive methods which only refine the mesh in an isotropic fashion, will in general lead to an excessive number of degrees of freedom in order to accurately resolve such structures present in the analytical solution. In the recent pioneering work of T. Apel and co-workers (see [2, 3, 4], for example), it has been clearly demonstrated that employing adaptive algorithms which can automati-

cally align the computational mesh in the direction of these localised structures in the solution through the means of *anisotropic* mesh refinement offer greater flexibility and efficiency over standard isotropic subdivision methods; see also [7, 11, 29, 30, 31, 32, 35, 37]. However, to date, the development of anisotropic mesh refinement algorithms has largely focused on the h–version of the finite element method, where the degree of the approximating polynomial is kept fixed at some low value.

In this work, we consider the *a posteriori* error estimation of the hp-version of the interior penalty Galerkin finite element method (DGFEM, for short) applied to second–order partial differential equations with nonnegative characteristic form on general finite element spaces consisting of an anisotropic computational mesh with anisotropic polynomial degree approximation orders. Here, we shall be interested in the reliable and efficient approximation of certain linear target functionals of the underlying analytical solution of practical interest, such as the mean value of the field over the computational domain and the normal flux through the outflow boundary. In particular, (weighted) Type I *a posteriori* error bounds are derived, based on employing the dual weighted residual approach, cf. [5, 18, 26, 38], for example. Based on the *a posteriori* error bound we design and implement the corresponding adaptive algorithm utilizing anisotropic hp–refinement to ensure the reliable and efficient control of the error in the prescribed target functional to within a given tolerance. Within this strategy, once elements have been marked for refinement/derefinement, on the basis of the size of the local error indicators, the proposed adaptive algorithm consists of two key steps: (a) Determine whether to undertake h– or p–refinement/derefinement; (b) Select a locally optimal anisotropic/isotropic refinement. Step (a) is based on assessing the local analyticity of the underlying primal and dual solutions, on the basis of the decay rates of Legendre series coefficients; see our previous articles [17, 28, 38], together with [9]. Step (b) is based on employing a competitive refinement strategy, whereby the "optimal" refinement is selected from a series of trial refinements. This entails the numerical solution of a series of local primal and dual problems which is relatively cheap and fully parallelizable. This algorithm represents the extension to the hp–version setting of the strategy developed in [13] for the case when the polynomial order is kept fixed. The superiority of the proposed algorithm in comparison with both standard hp–isotropic mesh refinement, and an h–anisotropic/p–isotopic refinement strategy is illustrated by a series of numerical experiments.

The paper is structured as follows. In order to highlight the general framework of the dual–weighted–residual approach adopted in this paper, in Section 2 we present an overview of goal-oriented *a posteriori* error estimation. In Section 3 we introduce the model problem and formulate its discontinuous Galerkin finite element approximation. Then, in Section 4 we derive a Type I *a posteriori* bound on the error measured in terms of certain linear target functionals of practical interest. Guided by our *a posteriori* error analysis, in Section 5 we design an adaptive finite element algorithm exploiting both local anisotropic h– and anisotropic p–refinement. The performance of the resulting refinement

strategy is then studied in Section 6 through a series of numerical experiments. Finally, in Section 7 we summarise the work presented in this paper and draw some conclusions.

2 A paradigm for a posteriori error estimation

In this section we present an overview of the general theoretical framework of duality–based *a posteriori* error estimation developed by C. Johnson and R. Rannacher and their collaborators. For a detailed discussion, we refer to the series of articles [5, 10, 22, 38], and the references cited therein.

Let X and Y be two Hilbert spaces. Further, we write $\mathcal{B}(\cdot,\cdot) : X \times Y \to \mathbb{R}$ to denote a bi-linear form. We suppose that u is the unique solution to the variational problem: find u in X such that

$$\mathcal{B}(u, v) = 0 \quad \forall v \in Y. \tag{1}$$

Problem (1) can be thought of as the weak formulation of a partial differential equation on X whose unique solution is $u \in X$. In practice (1) cannot be solved in closed form but needs to be approximated numerically. For the purposes of this paper, we shall consider general *hp*–version finite element approximations to (1). In order to construct a Galerkin approximation to this problem, we consider a sequence of finite–dimensional spaces $\{X_{h,p}\}$, parameterised by the positive discretisation parameters h and p; we have $X_{h,p} \subset X$ for each pair $\{h, p\}$ for conforming methods and $X_{h,p} \nsubseteq X$ for non-conforming methods, such as the DGFEM described below. Simultaneously, consider a sequence of finite–dimensional spaces $\{Y_{h,p}\}$. For the purposes of this paper, $X_{h,p}$ and $Y_{h,p}$ can be thought of as finite element spaces consisting of piecewise polynomial functions of degree p on a partition $\mathcal{T}_h$, of granularity h, of the computational domain. The Galerkin approximation $u_{h,p}$ of u is then sought in $X_{h,p}$ as the solution of the finite–dimensional problem

$$\mathcal{B}(u_{h,p}, v_{h,p}) = 0 \quad \forall v_{h,p} \in Y_{h,p}. \tag{2}$$

For simplicity of presentation, we assume that $X_{h,p}$ and $Y_{h,p}$ are suitably chosen finite element spaces to ensure the existence of a unique solution $u_{h,p}$ to (2).

In many problems of physical importance the quantities of interest may be a series of target or error functionals $J_i(\cdot)$, $i = 1, \ldots, N$, $N \geq 1$, of the solution. Relevant examples include the mean flow across a line, the point value of the solution, and the drag and lift coefficients of a body immersed into an inviscid fluid. For simplicity, we restrict ourselves to the case of a single *linear* target functional, i.e., $N = 1$, and write $J(\cdot) \equiv J_1(\cdot)$; for the extension of the proceeding theory to multiple target functionals, see [19]. In order to obtain a computable *a posteriori* bound on the error between the true value of the functional $J(u)$ and the computed value $J(u_{h,p})$, we begin by noting the Galerkin orthogonality of the discretisation (2):

$$\mathcal{B}(u, v_{h,p}) - \mathcal{B}(u_{h,p}, v_{h,p}) = \mathcal{B}(u - u_{h,p}, v_{h,p}) = 0 \quad \forall v_{h,p} \in Y_{h,p}. \tag{3}$$

This will be a key ingredient in the following *a posteriori* error analysis.

We now introduce the following *dual* or *adjoint* problem: find $z \in Y$ such that

$$\mathcal{B}(w, z) = J(w) \quad \forall w \in X. \tag{4}$$

We assume that (4) possesses a unique solution. Clearly, the validity of this assumption depends on both the definition of $\mathcal{B}(\cdot, \cdot)$ and the choice of the functional under consideration. Important examples which are covered by our hypothesis are discussed in [26].

For the proceeding error analysis, we must therefore *assume* that the dual problem (4) is well–posed. Under this assumption, employing the Galerkin orthogonality property (3) we deduce the following error representation formula:

$$\begin{aligned} J(u) - J(u_{h,p}) &= J(u - u_{h,p}) = \mathcal{B}(u - u_{h,p}, z) \\ &= \mathcal{B}(u - u_h, z - z_{h,p}) = -\mathcal{B}(u_h, z - z_{h,p}) \end{aligned} \tag{5}$$

for all $z_{h,p}$ in the finite element space $Y_{h,p}$. On the basis of the general error representation formula (5), *a posteriori* estimates which provide upper bounds on the true error in the computed target functional $J(\cdot)$ may be deduced. The simplest approach is to first decompose the right–hand side of (5) as a summation of local error indicators $\eta_\kappa^{(\mathrm{I})}$ over the elements κ in the computational mesh $\mathcal{T}_h$, i.e., we write

$$J(u) - J(u_{h,p}) = -\mathcal{B}(u_{h,p}, z - z_{h,p}) \equiv \mathcal{E}_\Omega^{(\mathrm{I})}(u_{h,p}, h, p, z - z_{h,p}) = \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa^{(\mathrm{I})};$$

then, upon application of the triangle inequality, we deduce the following weighted or Type I *a posteriori* error bound.

Theorem 2.1 *Let u and $u_{h,p}$ denote the solutions of (1) and (2), respectively, and suppose that the dual problem (4) is well–posed. Then, the following Type I* a posteriori *error bound holds:*

$$|J(u) - J(u_{h,p})| \leq \mathcal{E}_{|\Omega|}^{(\mathrm{I})}(u_{h,p}, h, p, z - z_{h,p}) \equiv \sum_{\kappa \in \mathcal{T}_h} |\eta_\kappa^{(\mathrm{I})}|. \tag{6}$$

We remark that the local error indicators $\eta_\kappa^{(\mathrm{I})}$ appearing on the right–hand side of (6) involve the multiplication of finite element *residuals* depending only on $u_{h,p}$ with local weighting terms involving the difference between the dual solution z satisfying (4) and its projection/interpolant $z_{h,p}$ onto the finite element space $Y_{h,p}$; in Section 4, we provide a concrete example arising from the discontinuous Galerkin approximation of second–order partial differential equations with nonnegative characteristic form. These weights represent the sensitivity of the error in the target functional $J(\cdot)$ with respect to variations of the local element residuals; indeed, they provide invaluable information concerning the global transport of the error, which is essential for efficient error control. Since the solution to the dual problem is usually unknown analytically it must be numerically approximated; see [5, 13, 18] for details. In this article we approximate z using piecewise polynomials of degree $\hat{p}$, $\hat{p} > p$, on the same finite element mesh $\mathcal{T}_h$ employed for the primal problem, cf. Section 5.

3 Model problem and discretization

We start by first introducing the function spaces that will be used throughout this paper. Given a bounded domain ω in $\mathbb{R}^d$, $d \geq 1$, we denote by $H^s(\omega)$ the standard Hilbertian Sobolev space of index $s \geq 0$ of real-valued functions defined on ω; we also set $L^2(\omega) = H^0(\omega)$.

Let Ω be a bounded open (curvilinear) polygonal domain in $\mathbb{R}^d$, $d \geq 1$, and let Γ signify the union of its $(d-1)$–dimensional open faces. We consider the advection–diffusion–reaction equation

$$\mathcal{L}u \equiv -\nabla \cdot (a\nabla u) + \nabla \cdot (\mathbf{b}u) + cu = f \ , \tag{7}$$

where $f \in L^2(\Omega)$ and $c \in L^\infty(\Omega)$ are real–valued, $\mathbf{b} = \{b_i\}_{i=1}^d$ is a vector function whose entries b_i are Lipschitz continuous real–valued functions on $\bar{\Omega}$, and $a = \{a_{ij}\}_{i,j=1}^d$ is a *symmetric* matrix whose entries a_{ij} are bounded, piecewise continuous real–valued functions defined on $\bar{\Omega}$, with

$$\boldsymbol{\zeta}^\top a(\mathbf{x})\boldsymbol{\zeta} \geq 0 \quad \forall \boldsymbol{\zeta} \in \mathbb{R}^d \, , \quad \text{a.e. } \mathbf{x} \in \bar{\Omega} \, . \tag{8}$$

Under this hypothesis, (7) is termed a *partial differential equation with nonnegative characteristic form*. By $\mathbf{n}(\mathbf{x}) = \{n_i(\mathbf{x})\}_{i=1}^d$ we denote the unit outward normal vector to Γ at $\mathbf{x} \in \Gamma$. On introducing the so called *Fichera function* $\mathbf{b} \cdot \mathbf{n}$ (cf. [34]), we define

$$\begin{aligned} \Gamma_0 &= \{\mathbf{x} \in \Gamma : \ \mathbf{n}(\mathbf{x})^\top a(\mathbf{x})\mathbf{n}(\mathbf{x}) > 0\} \, , \\ \Gamma_- &= \{\mathbf{x} \in \Gamma \backslash \Gamma_0 : \ \mathbf{b}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) < 0\} \, , \\ \Gamma_+ &= \{\mathbf{x} \in \Gamma \backslash \Gamma_0 : \ \mathbf{b}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) \geq 0\} \, . \end{aligned}$$

The sets Γ_- and Γ_+ will be referred to as the inflow and outflow boundary, respectively. Evidently, $\Gamma = \Gamma_0 \cup \Gamma_- \cup \Gamma_+$. If Γ_0 is nonempty, we shall further divide it into disjoint subsets Γ_D and Γ_N whose union is Γ_0, with Γ_D nonempty and relatively open in Γ. We supplement (7) with the boundary conditions

$$\begin{aligned} u &= g_\mathrm{D} \quad \text{on} \quad \Gamma_\mathrm{D} \cup \Gamma_- \, , \\ (a\nabla u) \cdot \mathbf{n} &= g_\mathrm{N} \quad \text{on} \quad \Gamma_\mathrm{N} \, , \end{aligned} \tag{9}$$

and adopt the (physically reasonable) hypothesis that $\mathbf{b} \cdot \mathbf{n} \geq 0$ on Γ_N, whenever Γ_N is nonempty. Additionally, we assume throughout that

$$(c_0(\mathbf{x}))^2 \equiv c(\mathbf{x}) + \frac{1}{2} \nabla \cdot \mathbf{b}(\mathbf{x}) \geq 0 \ \text{ a.e. } \mathbf{x} \in \Omega \, . \tag{10}$$

For the well-posedness theory (for weak solutions) of the boundary value problem (7), (9), in the case of homogeneous boundary conditions, we refer to [24, 27].

3.1 Meshes and finite element spaces

For simplicity of presentation, from this point onwards we assume that $d = 2$; however, we note that all of the results presented in this work naturally generalise to the case $d = 3$, by exploiting analogous arguments to those presented

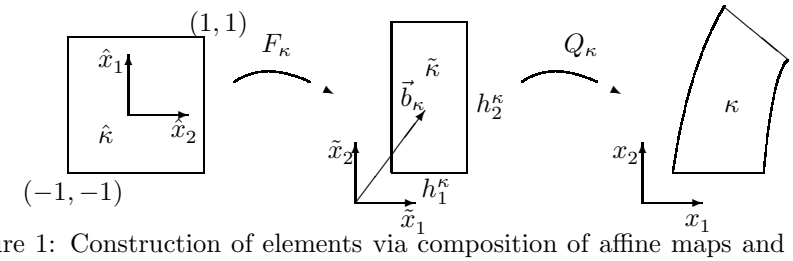

Figure 1: Construction of elements via composition of affine maps and diffeomorphisms.

subsequently in this article. Let $\mathcal{T}_h$ be a subdivision of the polygonal domain $\Omega \subset \mathbb{R}^2$ into disjoint open (curvilinear) quadrilateral elements κ constructed via the mappings $Q_\kappa \circ F_\kappa$, where $F_\kappa : \hat{\kappa} := (-1,1)^2 \to \tilde{\kappa}$ is an affine mapping of the form

$$F_\kappa(\mathbf{x}) := A_\kappa \mathbf{x} + \vec{b}_\kappa, \tag{11}$$

with $A_\kappa := \frac{1}{2}\,\mathrm{diag}(h_1^\kappa, h_2^\kappa)$, where h_1^κ and h_2^κ are the lengths of the edges of $\tilde{\kappa}$ parallel to the $\tilde{x}_1$- and $\tilde{x}_2$-axes, respectively, $\vec{b}_\kappa$ is a two-component real-valued vector, and $Q_\kappa : \tilde{\kappa} \to \kappa$ is a smooth diffeomorphism (cf. Figure 1).

Heuristically, we can say that the affine mapping F_κ defines the size of the element κ and the diffeomorphism Q_κ defines the "shape". For this reason, we shall be working with diffeomorphisms that are close to the identity in the following sense: the Jacobi matrix J_{Q_κ} of Q_κ satisfies

$$\begin{aligned} &C_1^{-1} \le \det J_{Q_\kappa} \le C_1, \\ &\|(J_{Q_\kappa})_{ij}\|_{L^\infty(\kappa)} \le C_2,\ i,j = 1,2 \end{aligned}$$

for all $\kappa \in \mathcal{T}_h$ uniformly throughout the mesh for some positive constants C_1 and C_2.

The above maps are assumed to be constructed so as to ensure that the union of the closures of the disjoint open elements $\kappa \in \mathcal{T}_h$ forms a covering of the closure of Ω, i.e., $\bar{\Omega} = \cup_{\kappa \in \mathcal{T}_h} \bar{\kappa}$. We shall restrict ourselves to meshes that are unions of diffeomorphic images of rectangles and to tensor-product polynomial spaces.

Also, we define the *broken Sobolev space* of composite order $\boldsymbol{s}$ on the open set Ω, subject to the subdivision $\mathcal{T}_h$ of Ω, as

$$H^{\boldsymbol{s}}(\Omega, \mathcal{T}_h) = \{u \in L^2(\Omega) : u|_\kappa \in H^{s_\kappa}(\kappa)\ \forall \kappa \in \mathcal{T}_h\},$$

with s_κ denoting the *local* Sobolev index on the element κ, and $\boldsymbol{s} := (s_\kappa : \kappa \in \mathcal{T}_h)$, together with the corresponding norm. When $s_\kappa = s$ for all $\kappa \in \mathcal{T}_h$, we shall write $H^s(\Omega, \mathcal{T}_h)$.

Let $\hat{I} \equiv (-1,1)$ and $\hat{\kappa} \equiv \hat{I} \times \hat{I} = (-1,1)^2$. On the interval $\hat{I}$ we denote the space of polynomials of degree p or less by $\mathcal{P}_p(\hat{I})$. Then, for $\mathbf{p} := (p_1, p_2)$, the *anisotropic tensor-product polynomial space* $\mathcal{Q}_{\mathbf{p}}$ on $\hat{\kappa}$ is defined by $\mathcal{Q}_{\mathbf{p}}(\hat{\kappa}) := \mathcal{P}_{p_1}(\hat{I}) \otimes \mathcal{P}_{p_2}(\hat{I})$, where $\otimes$ denotes the standard functional tensor product.

Let $\mathcal{T}_h$ be a subdivision of the computational domain Ω into elements $\kappa \in \mathcal{T}_h$ and let $\mathbf{F} = \{F_\kappa : \kappa \in \mathcal{T}_h\}$, $\mathbf{Q} = \{Q_\kappa : \kappa \in \mathcal{T}_h\}$, where F_κ, Q_κ are the maps defined above.

Definition 3.1 *Let* $\mathbf{p} := (\mathbf{p}_\kappa : \kappa \in \mathcal{T}_h)$ *be the composite polynomial degree vector of the elements in a given subdivision* $\mathcal{T}_h$*. We define the* finite element space *with respect to* Ω, $\mathcal{T}_h$, $\mathbf{F}$, *and* $\mathbf{p}$ *by*

$$S_{h,\mathbf{p}}(\Omega, \mathcal{T}_h) = \{u \in L^2(\Omega) : u|_\kappa \circ Q_\kappa \circ F_\kappa \in \mathcal{Q}_{\mathbf{p}_\kappa}(\hat{\kappa})\}.$$

For brevity we write $S_{h,\mathbf{p}}$ in lieu of $S_{h,\mathbf{p}}(\Omega, \mathcal{T}_h)$ when it is clear from the context the definition of the computational domain Ω and the finite element mesh $\mathcal{T}_h$ employed.

3.2 Interior penalty discontinuous Galerkin method

We introduce the (symmetric) interior penalty DGFEM discretization of the advection–diffusion–reaction problem (7), (9). To this end, we introduce the following notation.

An *interior edge* of $\mathcal{T}_h$ is defined as the (non-empty) one–dimensional interior of $\partial\kappa_i \cap \partial\kappa_j$, where κ_i and κ_j are two adjacent elements of $\mathcal{T}_h$, not necessarily matching. A *boundary edge* of $\mathcal{T}_h$ is defined as the (non-empty) one–dimensional interior of $\partial\kappa \cap \Gamma$, where κ is a boundary element of $\mathcal{T}_h$. We denote by Γ_{int} the union of all interior edges of $\mathcal{T}_h$. Given an edge $e \subset \Gamma_{\text{int}}$, shared by the two elements κ_i and κ_j, where the indices i and j satisfy $i > j$, we write $\mathbf{n}_e$ to denote the (numbering–dependent) unit normal vector which points from κ_i to κ_j; on boundary edges, we put $\mathbf{n}_e = \mathbf{n}$. Further, for $v \in H^1(\Omega, \mathcal{T}_h)$ we define the jump of v across e and the mean value of v on e, respectively, by $[v] = v|_{\partial\kappa_i \cap e} - v|_{\partial\kappa_j \cap e}$ and $\langle v \rangle = \frac{1}{2}\left(v|_{\partial\kappa_i \cap e} + v|_{\partial\kappa_j \cap e}\right)$.

On a boundary edge $e \subset \partial\kappa$, we set $[v] = v|_{\partial\kappa \cap e}$ and $\langle v \rangle = v|_{\partial\kappa \cap e}$. Finally, given a function $v \in H^1(\Omega, \mathcal{T}_h)$ and an element $\kappa \in \mathcal{T}_h$, we denote by v_κ^+ (respectively, v_κ^-) the interior (respectively, exterior) trace of v defined on $\partial\kappa$ (respectively, $\partial\kappa \backslash \Gamma$). Since below it will always be clear from the context which element κ in the subdivision $\mathcal{T}_h$ the quantities v_κ^+ and v_κ^- correspond to, for the sake of notational simplicity, we shall suppress the letter κ in the subscript and write, respectively, v^+ and v^- instead.

Given that κ is an element in the subdivision $\mathcal{T}_h$, we denote by $\partial\kappa$ the union of one–dimensional open edges of κ. Let $\mathbf{x} \in \partial\kappa$ and suppose that $\mathbf{n}_\kappa(\mathbf{x})$ denotes the unit outward normal vector to $\partial\kappa$ at $\mathbf{x}$. With these conventions, we define the inflow and outflow parts of $\partial\kappa$, respectively, by

$$\partial_-\kappa = \{\mathbf{x} \in \partial\kappa : \ \mathbf{b}(\mathbf{x}) \cdot \mathbf{n}_\kappa(\mathbf{x}) < 0\}\,, \quad \partial_+\kappa = \{\mathbf{x} \in \partial\kappa : \ \mathbf{b}(\mathbf{x}) \cdot \mathbf{n}_\kappa(\mathbf{x}) \geq 0\}\,.$$

For simplicity of presentation, we suppose that the entries of the matrix a are constant on each element κ in $\mathcal{T}_h$; i.e., $a \in [S_{h,0}]^{2\times 2}_{\text{sym}}$. We note that, with minor changes only, our results can easily be extended to the case of $\sqrt{a} \in [S_{h,q}]^{2\times 2}_{\text{sym}}$, $q \geq 0$; moreover, for general $a \in L^\infty(\Omega)^{2\times 2}_{\text{sym}}$, the analysis proceeds in a similar

manner, based on employing the modified DG method proposed in [15]. In the following, we write $\bar{a} = |\sqrt{a}\,|_2^2$, where $|\cdot|_2$ denotes the matrix norm subordinate to the l_2–vector norm on $\mathbb{R}^2$ and $\bar{a}_\kappa = \bar{a}|_\kappa$.

The DGFEM approximation of (7), (9) is defined as follows: find $u_{\rm DG}$ in $S_{h,\mathbf{p}}$ such that

$$B(u_{\rm DG}, v) = \ell(v) \tag{12}$$

for all $v \in S_{h,\mathbf{p}}$. Here, the bilinear form $B(\cdot,\cdot)$ is defined by

$$\begin{aligned} B(w,v) \;=\; & \sum_{\kappa\in\mathcal{T}_h} \Big\{ \int_\kappa a\nabla w\cdot\nabla v\,\mathrm{d}x - \int_\kappa (w\,\mathbf{b}\cdot\nabla v - cwv)\,\mathrm{d}x \\ & + \int_{\partial_+\kappa} (\mathbf{b}\cdot\mathbf{n}_\kappa)\,w^+v^+\,\mathrm{d}s + \int_{\partial_-\kappa\setminus\Gamma} (\mathbf{b}\cdot\mathbf{n}_\kappa)\,w^-v^+\,\mathrm{d}s\Big\} \\ & - \int_{\Gamma_{\rm int}\cup\Gamma_{\rm D}} \langle (a\nabla w)\cdot\mathbf{n}_e\rangle [v]\,\mathrm{d}s - \int_{\Gamma_{\rm int}\cup\Gamma_{\rm D}} \langle (a\nabla v)\cdot\mathbf{n}_e\rangle [w]\,\mathrm{d}s \\ & + \int_{\Gamma_{\rm int}\cup\Gamma_{\rm D}} \vartheta [w][v]\,\mathrm{d}s\;, \end{aligned}$$

and the linear functional $\ell(\cdot)$ is given by

$$\begin{aligned} \ell(v) \;=\; & \sum_{\kappa\in\mathcal{T}_h} \Big\{ \int_\kappa fv\,\mathrm{d}x + \int_{\partial\kappa\cap\Gamma_{\rm N}} g_{\rm N}v^+\,\mathrm{d}s - \int_{\partial_-\kappa\cap(\Gamma_{\rm D}\cup\Gamma_-)} (\mathbf{b}\cdot\mathbf{n}_\kappa)\,g_{\rm D}\,v^+\,\mathrm{d}s \\ & - \int_{\partial\kappa\cap\Gamma_{\rm D}} g_{\rm D}((a\nabla v^+)\cdot\mathbf{n}_\kappa)\,\mathrm{d}s + \int_{\partial\kappa\cap\Gamma_{\rm D}} \vartheta g_{\rm D}v^+\,\mathrm{d}s\Big\}\,. \end{aligned}$$

Here, ϑ is called the *discontinuity-penalization* function and is defined by $\vartheta|_e = \vartheta_e$ for $e \subset \Gamma_{\rm int}\cup\Gamma_{\rm D}$, where ϑ_e is a nonnegative constant on edge e. Before we give the precise definition for the discontinuity-penalization function ϑ, we require some notation. We define the function $\mathtt{h}$ in $L^\infty(\Gamma_{\rm int}\cup\Gamma_{\rm D})$, as $\mathtt{h}(\mathbf{x}) = \min\{h_j^\kappa, h_j^{\kappa'}\}$, if $\mathbf{x}$ is in the interior of $e = \partial\kappa\cap\partial\kappa'$ for two neighboring elements κ, κ' in the mesh $\mathcal{T}_h$, and $\tilde{e} = Q_\kappa^{-1}(e)$ is parallel to the $\tilde{x}_i$-axis, for $i,j = 1,2$, $i \neq j$; we also define $\mathtt{h}(\mathbf{x}) = h_j^\kappa$, if $\mathbf{x}$ is in the interior of $e = \partial\kappa\cap\Gamma_{\rm D}$ and $\tilde{e} = Q_\kappa^{-1}(e)$ is parallel to the $\tilde{x}_i$-axis, for $i,j = 1,2$, $i \neq j$. We note that in the isotropic setting we observe that $\mathtt{h} \sim h$, where h denotes the local mesh size. Similarly, we define the function $\mathtt{p}$ in $L^\infty(\Gamma_{\rm int}\cup\Gamma_{\rm D})$, as $\mathtt{p}(\mathbf{x}) = \max\{p_j^\kappa, p_j^{\kappa'}\}$, for κ, κ' as above; we also write $\mathtt{p}(\mathbf{x}) = p_j^\kappa$, if $\mathbf{x}$ is in the interior of a boundary edge as above. Also, we define the function $\mathtt{a}$ in $L^\infty(\Gamma_{\rm int}\cup\Gamma_{\rm D})$ by $\mathtt{a}(\mathbf{x}) = \max\{\bar{a}_\kappa, \bar{a}_{\kappa'}\}$ if $\mathbf{x}$ is in the interior of $e = \partial\kappa\cap\partial\kappa'$, and $\mathtt{a}(\mathbf{x}) = \bar{a}_\kappa$ if $\mathbf{x}$ is in the interior of $\partial\kappa\cap\Gamma_{\rm D}$.

With this notation, we define the discontinuity-penalization parameter ϑ arising in (12) by

$$\vartheta|_e \equiv \vartheta_e = C_\vartheta \frac{\mathtt{ap}^2}{\mathtt{h}} \quad \text{for} \quad e \subset \Gamma_{\rm int}\cup\Gamma_{\rm D}, \tag{13}$$

where C_ϑ is a positive constant. Selecting C_ϑ to be sufficiently large (a value of 10 usually suffices) guarantees the well–posed of the interior penalty DG method (12); see [12, 14] for details.

We shall assume that the solution u to the boundary value problem (7), (9) is sufficiently smooth: namely, $u \in H^{3/2+\varepsilon}(\Omega, \mathcal{T}_h)$, $\varepsilon > 0$, and the functions u and $(a\nabla u) \cdot \mathbf{n}_e$ are continuous across each edge $e \subset \partial\kappa\backslash\Gamma$ that intersects the subdomain of ellipticity, $\Omega_a = \{\mathbf{x} \in \bar{\Omega} : \boldsymbol{\zeta}^\top a(\mathbf{x})\boldsymbol{\zeta} > 0 \ \ \forall \boldsymbol{\zeta} \in \mathbb{R}^d\}$. Of course, in regions where a vanishes the above smoothness requirements can be further weakened; we avoid such issues of a technical nature, however, to enhance the clarity of the presentation. If this smoothness requirement is violated, the discretization method has to be modified accordingly, cf. [24]. We note that under these assumptions, the following Galerkin orthogonality property holds:

$$B_{\mathrm{DG}}(u - u_{\mathrm{DG}}, v) = 0 \quad \forall v \in S_{h,\mathbf{p}}\,. \tag{14}$$

4 *A posteriori* error estimation

Suppose that $\mathtt{TOL} > 0$ is a prescribed tolerance and $J(\cdot)$ is a given target functional. In this section, we consider the measurement problem concerned with computing the numerical approximation u_{DG} from $S_{h,\mathbf{p}}$ such that

$$|J(u) - J(u_{\mathrm{DG}})| \leq \mathtt{TOL}. \tag{15}$$

Recalling the notation introduced in Section 2, we define the following *dual* or *adjoint* problem: find $z \in H^{3/2+\varepsilon}(\Omega, \mathcal{T}_h)$, $\varepsilon > 0$, such that

$$B(w, z) = J(w) \quad \forall w \in H^2(\Omega, \mathcal{T}_h). \tag{16}$$

For the purposes of the proceeding analysis, we assume that (16) possesses a unique solution. Clearly, the validity of this assumption depends on the choice of the linear functional under consideration, cf. [26]. Under this assumption, we have the following general result.

Theorem 4.1 *Let u and u_{DG} denote the solutions of (7), (9) and (12), respectively, and suppose that the dual solution z is defined by (16). Then, the following error representation formula holds:*

$$J(u) - J(u_{\mathrm{DG}}) \quad = \quad \mathcal{E}_\Omega(u_{\mathrm{DG}}, h, p, z - z_{h,p}) \equiv \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa\,, \tag{17}$$

where

$$\begin{aligned}
\eta_\kappa \quad = \quad & \int_\kappa R_{\mathrm{int}}(z - z_{h,p})\,\mathrm{d}x - \int_{\partial_- \kappa \cap \Gamma} (\mathbf{b}\cdot\mathbf{n}_\kappa)\, R_{\mathrm{D}}(z - z_{h,p})^+\,\mathrm{d}s \\
& + \int_{\partial_- \kappa \backslash \Gamma} (\mathbf{b}\cdot\mathbf{n}_\kappa)\, [u_{\mathrm{DG}}](z - z_{h,p})^+\,\mathrm{d}s \\
& - \int_{\partial\kappa \cap \Gamma_{\mathrm{D}}} R_{\mathrm{D}}((a\nabla(z - z_{h,p})^+)\cdot\mathbf{n}_\kappa)\,\mathrm{d}s + \int_{\partial\kappa\cap\Gamma_{\mathrm{D}}} \vartheta R_{\mathrm{D}}(z - z_{h,p})^+\,\mathrm{d}s \\
& + \int_{\partial\kappa\cap\Gamma_{\mathrm{N}}} R_{\mathrm{N}}(z - z_{h,p})^+\,\mathrm{d}s - \int_{\partial\kappa\backslash\Gamma} \vartheta [u_{\mathrm{DG}}](z - z_{h,p})^+\,\mathrm{d}s \\
& + \frac{1}{2}\int_{\partial\kappa\backslash\Gamma} \left\{\, [u_{\mathrm{DG}}](a\nabla(z - z_{h,p})^+)\cdot\mathbf{n}_\kappa - [(a\nabla u_{\mathrm{DG}})\cdot\mathbf{n}_\kappa](z - z_{h,p})^+ \right\}\mathrm{d}s,
\end{aligned} \tag{18}$$

for all $z_{h,p} \in S_{h,\mathbf{p}}$. *Additionally, the* internal *finite element residual* $R_{\rm int}$ *and the* boundary *residuals* $R_{\rm D}$ *and* $R_{\rm N}$ *are defined, respectively, by*

$$\begin{aligned} R_{\rm int}|_{\kappa} &= (f - \mathcal{L}u_{\rm DG})|_{\kappa}\,, \\ R_{\rm D}|_{\partial\kappa\cap(\Gamma_{\rm D}\cup\Gamma_-)} &= (g_{\rm D} - u^+_{\rm DG})|_{\partial\kappa\cap(\Gamma_{\rm D}\cup\Gamma_-)}\,, \\ R_{\rm N}|_{\partial\kappa\cap\Gamma_{\rm N}} &= (g_{\rm N} - (a\nabla u^+_{\rm DG})\cdot\mathbf{n})|_{\partial\kappa\cap\Gamma_{\rm N}} \end{aligned}$$

for all elements κ *in the finite element mesh* $\mathcal{T}_h$.

Thereby, on application of the triangle inequality, we deduce the following Type I *a posteriori* error bound.

Corollary 4.2 *Under the assumptions of Theorem 4.1, the following Type I* a posteriori *error bound holds:*

$$|J(u) - J(u_{\rm DG})| \leq \mathcal{E}_{|\Omega|}(u_{\rm DG}, h, p, z - z_{h,p}) \equiv \sum_{\kappa\in\mathcal{T}_h} |\eta_\kappa|\,, \tag{19}$$

where η_κ *is defined as in (19).*

As discussed in [18, 38], the local weighting terms involving the difference between the dual solution z and its projection/interpolant $z_{h,p}$ onto $S_{h,\mathbf{p}}$ appearing in the Type I bound (19) provide invaluable information concerning the global transport of the error. Thereby, we refrain from eliminating the weighting terms involving the (unknown) dual solution z and approximate z numerically; this will be discussed in Section 5.

5 Adaptive algorithm

The *a priori* error analysis developed in the companion paper [14] clearly highlights that in order to minimize the error in the computed target functional $J(\cdot)$ with respect to the number of degrees of freedom in the finite element space $S_{h,\mathbf{p}}$, the design of an optimal *hp*–mesh distribution must exploit anisotropic information emanating from both the primal and dual solutions u and z, respectively. Indeed, a mesh solely optimized for u may be completely inappropriate for z, and vice versa, thus there must be a trade-off between aligning the elements with respect to either solution in order to minimize the overall error in $J(\cdot)$; cf., also, [13]. Moreover, the theoretical results in [14] highlight the importance of the ability to resolve directional features of the solution using a combination of anisotropic meshes with anisotropic polynomial degree distributions. The construction of an automated procedure which is capable of producing such desirable anisotropic finite element spaces is the subject of the current section.

Recalling the measurement problem stated in Section 4: the aim of the computation is to design an appropriate "optimal" finite element space $S_{h,\mathbf{p}}$ such that

$$|J(u) - J(u_{\rm DG})| \leq \texttt{TOL}\,,$$

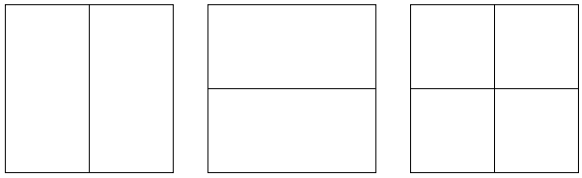

Figure 2: Cartesian refinement in 2D: (a) & (b) Anisotropic refinement; (c) Isotropic refinement.

where $\texttt{TOL} > 0$ is a given user-defined tolerance. By optimal we mean that the above error control should be attained using a minimal number of degrees of freedom. For simplicity of presentation, in this section we only consider the case when $\Omega \subset \mathbb{R}^2$ and $\mathcal{T}_h$ consists of *1–irregular* quadrilateral elements. Following the discussion presented [26], we exploit the *a posteriori* error bound (19) with z replaced by a discontinuous Galerkin approximation $\hat{z}$ computed on the same mesh $\mathcal{T}_h$ used for the primal solution $u_{\rm DG}$, but with a higher degree polynomial, i.e., $\hat{z} \in S_{h,\hat{\mathbf{p}}}$, $\hat{\mathbf{p}} = \mathbf{p} + p_{\rm inc}$; in Section 6, we set $p_{\rm inc} = 1$, cf. [18, 38]. Thereby, in practice we enforce the stopping criterion

$$\hat{\mathcal{E}}_{|\Omega|} \equiv \mathcal{E}_{|\Omega|}(u_{\rm DG}, \hat{z} - z_{h,p}) \leq \texttt{TOL}\,. \tag{20}$$

If (20) is not satisfied, then the elements are marked for refinement/derefinement according to the size of the (approximate) error indicators $|\hat{\eta}_\kappa|$; these are defined analogously to $|\eta_\kappa|$ in (19) with z replaced by $\hat{z}$. In Section 6 we use the fixed fraction mesh refinement algorithm, with refinement and derefinement fractions set to 20% and 10%, respectively.

Once an element has been selected for refinement/derefinement a decision must first be made whether to carry out an h-refinement/derefinement or p-enrichment/derefinement. To this end, we exploit the technique developed in [28], whereby the analyticity of the solutions u and z is assessed by studying the decay rates of their underlying Legendre coefficients; see, also, [9] for related work. An approximation of the first few Legendre coefficients of u and z are readily obtained from the approximate solutions $u_{\rm DG}$ and $\hat{z}$, respectively, and hence a measure of the smoothness of the respective solutions is available for minimal computational effort. The *a priori* estimates developed in the prequel [14], clearly indicate that if either u or z are smooth then a high polynomial degree is preferable to a small mesh size, whereas if u and z are both nonsmooth then a small mesh size should be utilized, cf. [17, 38]. With this in mind, should an element be selected for refinement and both u and z be nonsmooth we perform a mesh subdivision, otherwise we perform polynomial enrichment. Similarly, if an element is flagged for derefinement, then if neither u nor z are smooth we carry out a p-derefinement, else an h-derefinement is undertaken.

Once the h– and p–refinement flags have been determined on the basis of the above strategy, a decision regarding the type refinement to be undertaken

— isotropic or anisotropic — must be made. Motivated by the work in our previous article [13], we employ a competitive refinement technique, whereby the "optimal" refinement is selected from a series of trial refinements. In the h–version setting, article [13] proposed two such algorithms, which we shall now briefly recall. Elements which have been flagged for an h-refinement are subdivided employing a simple Cartesian refinement strategy; more precisely, elements may be subdivided either anisotropically or isotropically according to the three refinements (in two–dimensions, i.e., $d = 2$) depicted in Figure 2. The optimal refinement is then selected on the basis of one of the following two strategies.

`Algorithm 1:` Given an element κ in the computational mesh $\mathcal{T}_h$ (which has been marked for h–refinement), we first construct the mesh patches $\mathcal{T}_{h,i}$, $i = 1, 2, 3$, based on refining κ according to Figures 2(a), (b), & (c), respectively. On each mesh patch, $\mathcal{T}_{h,i}$, $i = 1, 2, 3$, we compute the approximate error estimators

$$\hat{\mathcal{E}}_{\kappa,i}(u_{\mathrm{DG},i}, \hat{z}_i - z_{h,p}) = \sum_{\kappa' \in \mathcal{T}_{h,i}} \eta_{\kappa',i},$$

for $i = 1, 2, 3$, respectively. Here, $u_{\mathrm{DG},i}$, $i = 1, 2, 3$, is the discontinuous Galerkin approximation to (7), (9) computed on the mesh patch $\mathcal{T}_{h,i}$, $i = 1, 2, 3$, respectively, based on enforcing appropriate boundary conditions on $\partial\kappa$ computed from the original discontinuous Galerkin solution u_{DG} on the portion of the boundary $\partial\kappa$ of κ which is interior to the computational domain Ω, i.e., where $\partial\kappa \cap \Gamma = \emptyset$. Similarly, $\hat{z}_i$ denotes the discontinuous Galerkin approximation to z computed on the local mesh patch $\mathcal{T}_{h,i}$, $i = 1, 2, 3$, respectively, with polynomials of degree $\hat{\mathbf{p}}$, based on employing suitable boundary conditions on $\partial\kappa \cap \Gamma = \emptyset$ derived from $\hat{z}$. Finally, $\eta_{\kappa',i}$, $i = 1, 2, 3$, is defined in an analogous manner to η_κ, cf. (19) above, with u_{DG} and z replaced by $u_{\mathrm{DG},i}$ and $\hat{z}_i$, respectively.

The element κ is then refined according to the subdivision of κ which satisfies

$$\min_{i=1,2,3} \frac{|\eta_\kappa| - |\hat{\mathcal{E}}_{\kappa,i}(u_{\mathrm{DG},i}, \hat{z}_i - z_{h,p})|}{\#\mathrm{dofs}(S_{h,\mathbf{p}}(\kappa, \mathcal{T}_{h,i})) - \#\mathrm{dofs}(S_{h,\mathbf{p}}(\kappa, \kappa))}$$

where $\#\mathrm{dofs}(S_{h,\mathbf{p}}(\kappa, \kappa))$ and $\#\mathrm{dofs}(S_{h,\mathbf{p}}(\kappa, \mathcal{T}_{h,i}))$, $i = 1, 2, 3$, denote the number of degrees of freedom associated with the local finite element spaces $S_{h,\mathbf{p}}(\kappa, \kappa)$ and $S_{h,\mathbf{p}}(\kappa, \mathcal{T}_{h,i})$, $i = 1, 2, 3$, respectively.

`Algorithm 2:` This is very similar to `Algorithm 1`; however, here we only construct the mesh patches $\mathcal{T}_{h,i}$, $i = 1, 2$, and compute the approximate local primal and dual solutions on these meshes only. Given an anisotropy parameter $\theta \geq 1$, isotropic refinement is selected when

$$\frac{\max_{i=1,2} |\hat{\mathcal{E}}_{\kappa,i}(u_{\mathrm{DG},i}, \hat{z}_i - z_{h,p})|}{\min_{i=1,2} |\hat{\mathcal{E}}_{\kappa,i}(u_{\mathrm{DG},i}, \hat{z}_i - z_{h,p})|} < \theta;$$

otherwise an anisotropic refinement is performed based on which refinement gives rise to the smallest predicted error indicator, i.e., the subdivision for which

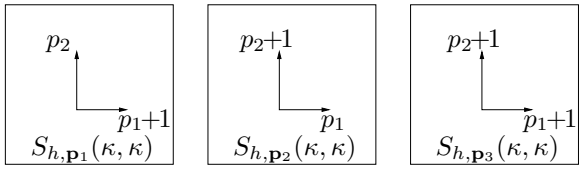

Figure 3: Polynomial Enrichment in 2D: (a) & (b) Anisotropic Enrichment; (c) Isotropic Enrichment.

$|\hat{\mathcal{E}}_{\kappa,i}(u_{\mathrm{DG},i}, \hat{z}_i - z_{h,p})|$, $i = 1, 2$, is minimal. Based on computational experience, we select θ in the range $[2, 3]$.

In [13] Algorithm 2 proved itself to be the most effective algorithm to use for h-refinement by giving comparable errors to Algorithm 1, but with reduced computational cost; hence, in this article we restrict ourselves to only employing this strategy when h-refinement is selected.

For the case when an element has been selected for polynomial enrichment we consider the p-version counterparts of the above algorithms and solve local problems based on increasing the polynomial degrees anisotropically in one direction at a time by one degree, or isotropically by one degree. Figure 3 provides a visualisation of the local mesh patches in two-dimensions, where the original polynomial degree vector on the element of interest is $\mathbf{p} = [p_1, p_2]$. More precisely, we consider the following two strategies.

`Algorithm 3:` This algorithm represents the p–version of `Algorithm 1` above. Given an element κ in the computational mesh $\mathcal{T}_h$ (which has been marked for p–refinement), we first construct the local finite element spaces $S_{h,\mathbf{p}_i}(\kappa,\kappa)$, $i = 1, 2, 3$, based on enriching $\mathbf{p}$ according to Figures 3(a), (b) and (c), respectively. On each finite element space $S_{h,\mathbf{p}_i}(\kappa,\kappa)$, $i = 1, 2, 3$, we compute the approximate error estimators

$$\bar{\mathcal{E}}_{\kappa,i}(\bar{u}_{\mathrm{DG},i}, \bar{z}_i - \bar{z}_{h,p}) \equiv \bar{\eta}_{\kappa,i},$$

for $i = 1, 2, 3$, respectively. Here, $\bar{u}_{\mathrm{DG},i}$, $i = 1, 2, 3$, is the discontinuous Galerkin approximation to (7), (9) computed on the finite element space $S_{h,\mathbf{p}_i}(\kappa,\kappa)$, $i = 1, 2, 3$. Similarly, $\bar{z}_i$ denotes the discontinuous Galerkin approximation to z computed on $S_{h,\mathbf{p}_i+p_{\mathrm{inc}}}(\kappa,\kappa))$, $i = 1, 2, 3$, respectively, with polynomials of degree $\mathbf{p}_i + p_{\mathrm{inc}}$.

The element κ is then refined according to the subdivision of κ which satisfies

$$\min_{i=1,2,3} \frac{|\eta_\kappa| - |\bar{\mathcal{E}}_{\kappa,i}(\bar{u}_{\mathrm{DG},i}, \bar{z}_i - \bar{z}_{h,p})|}{\#\mathrm{dofs}(S_{h,\mathbf{p}_i}(\kappa,\kappa)) - \#\mathrm{dofs}(S_{h,\mathbf{p}}(\kappa,\kappa))}.$$

`Algorithm 4:` This is very similar to `Algorithm 3`; here only the finite element spaces $S_{h,\mathbf{p}_i}(\kappa,\kappa)$, $i = 1, 2$, corresponding to the anisotropic polynomial

enrichments are undertaken. Given an anisotropy parameter $\omega \geq 1$, isotropic enrichment is selected when

$$\frac{\max_{i=1,2}((|\eta_\kappa| - |\bar{\mathcal{E}}_{\kappa,i}|)/(\#\text{dofs}(S_{h,\mathbf{p}_i}(\kappa,\kappa)) - \#\text{dofs}(S_{h,\mathbf{p}}(\kappa,\kappa))))}{\min_{i=1,2}((|\eta_\kappa| - |\bar{\mathcal{E}}_{\kappa,i}|)/(\#\text{dofs}(S_{h,\mathbf{p}_i}(\kappa,\kappa)) - \#\text{dofs}(S_{h,\mathbf{p}}(\kappa,\kappa))))} < \omega,$$

otherwise an anisotropic enrichment is performed based on which enrichment gives rise to the smallest predicted error indicator, i.e., the subdivision for which $(|\eta_\kappa| - |\bar{\mathcal{E}}_{\kappa,i}|)/(\#\text{dofs}(S_{h,\mathbf{p}_i}(\kappa,\kappa)) - \#\text{dofs}(S_{h,\mathbf{p}}(\kappa,\kappa)))$, $i = 1, 2$, is minimal. Here, for brevity, we have written $\bar{\mathcal{E}}_{\kappa,i}$ in lieu of $\bar{\mathcal{E}}_{\kappa,i}(\bar{u}_{\text{DG},i}, \bar{z}_i - \bar{z}_{h,p})$, $i = 1, 2$. As before, based on computational experience, we select ω in the range $[2, 3]$.

For clarity, the fully anisotropic *hp*-adaptive algorithm presented above can be viewed as a flowchart in Figure 4.

In the following section we shall study the performance of the fully adaptive anisotropic *hp*–refinement algorithm combining `Algorithm 2` with either `Algorithm 3` or `Algorithm 4`. As in the *h*–version setting, we shall see that both anisotropic *p*–refinement strategies lead to quantitatively the same error reduction in the target functional $J(\cdot)$ of interest for a given number of degrees of freedom. Thereby, the combination of `Algorithm 2` and `Algorithm 4` is advocated, on the basis of computational efficiency.

6 Numerical experiments

In this section we present a number of experiments to numerically demonstrate the performance of the *hp*–anisotropic adaptive algorithms outlined in Section 5.

6.1 Example 1

We consider the following (singularly perturbed) advection–diffusion problem equation

$$-\varepsilon\Delta u + u_x + u_y = f,$$

for $(x, y) \in (0, 1)^2$, where $0 < \varepsilon \ll 1$ and f is chosen so that

$$u(x, y) = x + y(1 - x) + [\mathrm{e}^{-1/\varepsilon} - \mathrm{e}^{-(1-x)(1-y)/\varepsilon}]\,[1 - \mathrm{e}^{-1/\varepsilon}]^{-1}. \tag{21}$$

This is a multidimensional variant of the one-dimensional problem considered by [33], cf. [24]. For $0 < \varepsilon \ll 1$ the solution (21) has boundary layers along $x = 1$ and $y = 1$. Here, we suppose that the aim of the computation is to calculate the value of the (weighted) mean-value of u over the computational domain Ω, i.e.,

$$J(u) = \int_\Omega u\psi \,\mathrm{d}\mathbf{x},$$

where the weight function ψ is chosen as follows:

$$\begin{aligned}\psi(x, y) \quad = \quad & 4(1 - 2y)(1 - \mathrm{e}^{-\alpha(1-x)} - (1 - \mathrm{e}^{-\alpha})(1 - x)) \\ & +4y(y - 1)(\mathrm{e}^{-\alpha(1-x)}(\alpha - (1 - \mathrm{e}^{-\alpha})).\end{aligned}$$

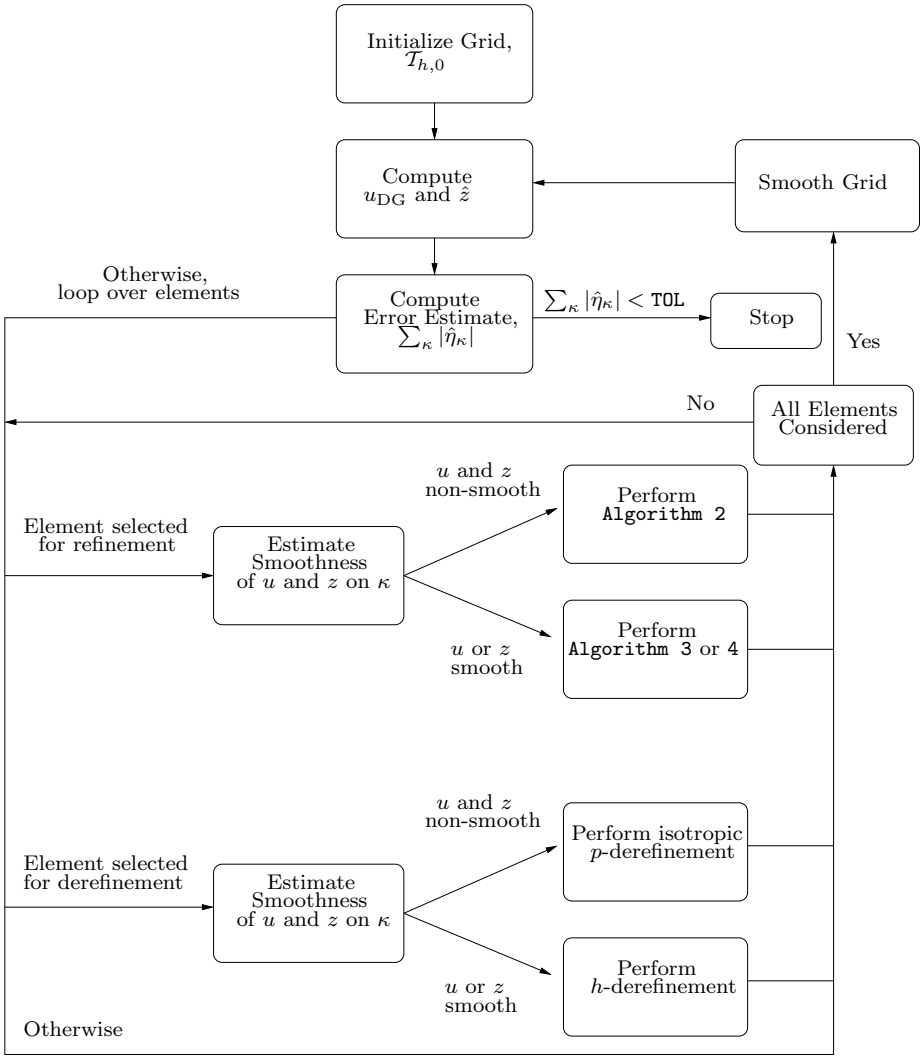

Figure 4: Anisotropic *hp*-adaptive algorithm.

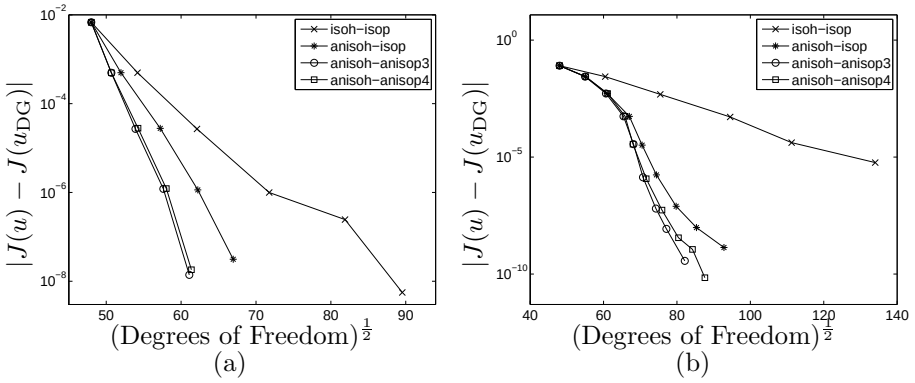

Figure 5: Example 1. Comparison between adaptive *hp*–refinement strategies: (a) $\varepsilon = 10^{-2}$; (b) $\varepsilon = 10^{-3}$.

Setting $\alpha = 100$ gives rise to a strong boundary layer in the analytical solution z to the corresponding dual problem (16) along the boundary $x = 1$ and a weaker boundary layer along $y = 0$. We remark that ψ has been chosen so that both u and z have boundary layers along the same portion of the computational boundary, as this kind of behaviour is quite typical in practical applications; for example, the computation of the lift or drag on an airfoil immersed within a viscous fluid, cf., e.g., [20].

Here, we compare the performance of the anisotropic *hp*-refinement adaptive strategies outlined in the previous section using a combination of either `Algorithm 2` and `Algorithm 3` or `Algorithm 2` and `Algorithm 4`, together with a (standard) isotropic *hp*–refinement strategy, and an *h*–anisotropic/*p*–isotropic refinement algorithm based on employing `Algorithm 2` to decide the anisotropy in the mesh. In both of these two latter strategies, the decision to perform either *h*– or *p*–refinement/derefinement is again based on estimating the local analyticity of the primal and dual solutions u and z, cf. Section 5. In all cases, we begin with a uniform (square) mesh with 17 points in each coordinate direction and assign a uniform polynomial degree vector $\mathbf{p} = [2, 2]$ on each element.

In Figures 5(a) & (b) we plot the (square root of the) degrees of freedom employed in the finite element space $S_{h,\mathbf{p}}$ against the error in the computed target functional $J(\cdot)$, for $\varepsilon = 10^{-2}, 10^{-3}$, respectively, using each of the four *hp*–mesh refinement algorithms defined above. Firstly, we note that in all cases, the convergence lines are (on average) straight, indicating exponential rates of convergence have been achieved using all four refinement strategies for each ε, cf. [14]. Secondly, for each ε, we observe that the computed error, for a given number of degrees of freedom, employing the *h*–isotropic/*p*–isotropic strategy is always inferior to the algorithm employing *h*–anisotropic/*p*–isotropic refinement. Similarly, this latter strategy is inferior to exploiting one of the two *h*–anisotropic/*p*–

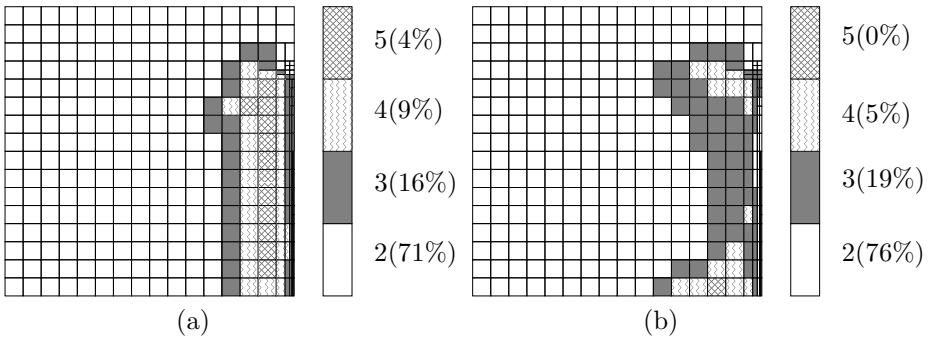

Figure 6: Example 1. Anisotropic *hp*-meshes after 4 refinement steps employing `Algorithm 2` and `Algorithm 4`, with 316 elements and 3767 degrees of freedom: (a) p_x and (b) p_y, for $\varepsilon = 10^{-2}$.

anisotropic refinement algorithms proposed in this article. Indeed, for $\varepsilon = 10^{-2}$, after the final refinement step, the anisotropic *hp*–strategies yield over two orders of magnitude improvement over the *h*–anisotropic/*p*–isotropic case and nearly 4 orders of magnitude improvement over the isotropic *hp*–method. For $\varepsilon = 10^{-3}$, the anisotropic *hp*–strategies yields around *seven* orders of magnitude improvement in the error in the computed target functional $J(\cdot)$ after the final refinement step, for the same numbers of degrees of freedom, in comparison to the isotropic *hp*–refinement strategy, and two orders of magnitude improvement over the *h*–anisotropic/*p*–isotropic refinement algorithm. In this latter case, we note that the anisotropic *hp*–refinement algorithms and the *h*–anisotropic/*p*–isotropic strategy perform equally well during the first few refinement steps, since only *h*–adaptation is undertaken. However, as soon as *p*–enrichment is required the use of anisotropic polynomial degrees becomes clearly advantageous. In contrast, in the former case when $\varepsilon = 10^{-2}$, we observe an immediate improvement when employing anisotropic *hp*–adaptivity. Finally, we mention that both `Algorithm 3` and `Algorithm 4` employed within the anisotropic *hp*–adaptive strategy to determine the anisotropy in the spectral orders, give rise to quantitatively similar convergence histories, indicating that `Algorithm 4` may be preferable as it is more computationally efficient.

Figure 6 shows the resultant *hp*–mesh distribution employing `Algorithm 2` and `Algorithm 4` after 4 anisotropic *hp*–refinement steps for $\varepsilon = 10^{-2}$; here, Figures 6(a) and (b) show the polynomial degrees employed in the *x*– and *y*–directions, respectively. We observe that anisotropic *h*–refinement has been employed in order to resolve the right–hand side boundary layer and anisotropic *p*-refinement has been utilized further inside the computational domain. In particular, we notice that the polynomial degrees have been increased to a higher level in the *x*–direction, than in the orthogonal direction, as we would expect. Quantitatively similar *hp*–mesh distributions are generated for $\varepsilon = 10^{-3}$; for brevity, we omit these results.

6.2 Example 2

In this second example we investigate the performance of the proposed *hp*–anisotropic refinement algorithms applied to a mixed hyperbolic–elliptic problem with discontinuous boundary data. To this end, we let $\Omega = (0,2) \times (0,1)$, $a = \varepsilon(x)I$, where $\varepsilon = (1 - \tanh(100(r_1 - 0.12)(r_1 + 0.12)))(1 - \tanh(100(r_2 - 0.12)(r_2 + 0.12)))/1000$, $r_1 = x - 1.3$ and $r_2 = y - 0.3$. Furthermore, we set

$$\mathbf{b} = \begin{cases} (y, 1-x)^\top & \text{if } x < 1, \\ (1, 1/10)^\top & \text{if } x \geq 1, \end{cases}$$

$c = 0$, and $f = 0$. On the inflow boundary Γ_-, we select $u(x,y) = 1$ along $y = 0$, $1/8 < x < 3/4$, and $u(x,y) = 0$, elsewhere. This is a variant of the test problem presented in [21]. We note that the diffusion parameter ε will be approximately equal to 3.6×10^{-3} in the square region $(1.18, 1.42) \times (0.18, 0.42)$, where the underlying partial differential equation is uniformly elliptic. As (x,y) moves outside of this region, ε rapidly decreases through a layer of width $\mathcal{O}(0.1)$; for example, when $x = 1.3$ and $y > 0.7$ we have $\varepsilon < 10^{-15}$, so from the computational point of view ε is zero to within rounding error; in this region, the partial differential equation undergoes a change of type becoming, in effect, hyperbolic. Thus, we shall refer to the part of Ω containing this square region (including a strip of size $\mathcal{O}(0.1)$) as the *elliptic region*, while the remainder of the computational domain will be referred to as the *hyperbolic region*. (Strictly speaking, the partial differential equation is elliptic in the whole of $\bar{\Omega}$.)

Here, we suppose that the aim of the computation is to calculate the value of the (weighted) outflow advective flux along $x = 2$, $0 \leq y \leq 1$, i.e., $J(u) = \int_0^1 (\mathbf{b} \cdot \mathbf{n}) u(2,y) \psi(y) \, \mathrm{d}y$, where the weight function, in a modification to [13], is

$$\psi(y) = \begin{cases} (\tanh(50(y - 7/40)) + 1)/2 & y < 17/40, \\ (\tanh(-50(y - 27/40)) + 1)/2 & y \geq 17/40. \end{cases}$$

The true value of the functional is given by $J(u) = 0.324999805677598$.

Given the qualitatively comparable results shown in the previous example, between exploiting the *p*-anisotropic algorithms `Algorithm 3` and `Algorithm 4`, in conjunction with the *h*–anisotropic algorithm `Algorithm 2`, in this section we shall only consider the latter approach, i.e., the *hp*–anisotropic algorithm exploiting `Algorithm 2` and `Algorithm 4`, as it is more computationally efficient. However, once again we compare this strategy with both the *hp*–isotropic and *h*–anisotropic/*p*–isotropic refinement algorithms described in Section 6.1. In all cases the starting *hp*–mesh distribution is a uniform 17×9 grid, consisting of uniform square elements, with the uniform polynomial degree distribution $\mathbf{p} = [2,2]$ on each element.

In Figure 7 we plot the (square root of the) degrees of freedom employed in the finite element space $S_{h,\mathbf{p}}$ against the error in the computed target functional $J(\cdot)$, using each of the three *hp*–mesh refinement algorithms defined above, namely *hp*–isotropic refinement, *h*–anisotropic/*p*–isotropic refinement, and *hp*–anisotropic refinement (employing `Algorithm 2` and `Algorithm 4`). As in the

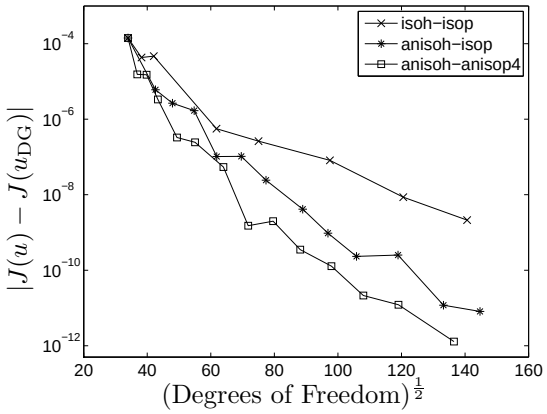

Figure 7: Example 2. Comparison between adaptive *hp*–refinement strategies.

previous example, we note that in all cases, after an initial transient, the convergence lines are (on average) straight, indicating exponential rates of convergence have been achieved using all three refinement strategies. Similarly, we again observe that the computed error, for a given number of degrees of freedom, employing the *h*–isotropic and *p*–isotropic strategy is always inferior to the algorithm employing *h*–anisotropic and *p*–isotropic refinement, which is in turn inferior to *hp*–anisotropic refinement algorithm proposed in this article. Evidently the majority of improvement over the *hp*–isotropic strategy is due to employing anisotropic *h*–refinement, cf. the previous example when $\varepsilon = 10^{-3}$, yet in the asymptotic regime the *hp*–anisotropic strategy consistently shows around an order of magnitude improvement in the error for the same number of degrees of freedom, when compared with the *h*–anisotropic/*p*–isotropic refinement strategy.

Finally, Figures 8(a) and (b) show the resultant computational mesh and polynomial degree distribution in the x– and y–directions, respectively, after 8 steps of our *hp*–anisotropic refinement strategy. Here, we see that the majority of *h*–refinement has taken place primarily along the layer of the analytical solution u emanating from the point $(x, y) = (3/4, 0)$. In other regions *p*–enrichment has been favoured; indeed there is a marked difference between the polynomial degrees employed in the x– and y–directions, with the majority of elements having had no *p*–enrichment in the x–direction, while most element have had some *p*–enrichment in the y–direction. The *p*–enrichment in the x–direction has been concentrated in the left half of the domain as this is where layers in the primal and dual solutions run parallel to the y–axis, while for the same reason *p*–enrichment in the y–direction is concentrated in the right portion of the computational domain.

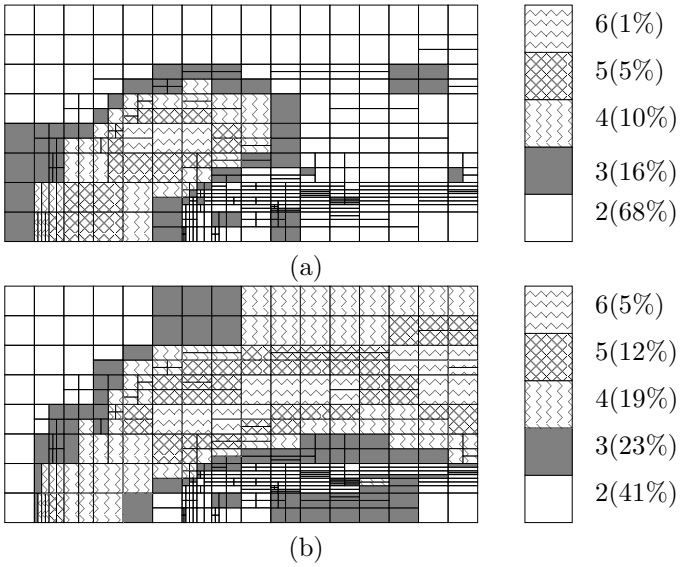

Figure 8: Example 2. Anisotropic *hp*–meshes after 8 refinement steps employing `Algorithm 2` and `Algorithm 4`, with 410 elements and 6338 degrees of freedom: (a) p_x and (b) p_y.

7 Concluding Remarks

This article and its companion paper [14] have been concerned with the *a priori* and *a posteriori* error analyses of the (symmetric) interior penalty discontinuous Galerkin finite element discretization of second–order partial differential equations with nonnegative characteristic form, based on employing anisotropically refined computational meshes with anisotropic polynomial spaces. Indeed in the present work, we have developed so–called Type I (weighted) *a posteriori* error bounds for general linear functionals of practical relevance using the general dual–weighted–residual method developed by R. Rannacher and co-workers. On the basis of this computable bound, the key aspect of this article has been the development of a general purpose *hp*–adaptive finite element algorithm which is capable of both anisotropically refining the computational mesh and anisotropically enriching the local polynomial degrees. Here, once elements have been marked for refinement/derefinement, on the basis of the size of the local error indicators, the proposed adaptive algorithm consists of two key steps: (a) Determine whether to undertake h– of p–refinement/derefinement; (b) Select a locally optimal anisotropic/isotropic refinement. Step (a) is based on assessing the local analyticity of the underlying primal and dual solutions, on the basis of the decay rates of Legendre series coefficients; cf., [17, 38, 28], together with [9]. Step (b) is based on employing a competitive refinement strategy, whereby the "optimal" refinement is selected from a series of trial refinements.

This entails the numerical solution of a series of local primal and dual problems which is relatively cheap and fully parallelizable. The performance of the proposed *hp*–anisotropic refinement strategy has been studied through a series of numerical experiments. In particular, we have demonstrated the superiority of the proposed approach in comparison with both standard *hp*–isotropic mesh refinement, and an *h*–anisotropic/*p*–isotopic refinement strategy.

References

[1] M. Ainsworth and J.T. Oden. *A Posteriori Error Estimation in Finite Element Analysis*. Series in Computational and Applied Mathematics. Elsevier, 1996.

[2] T. Apel. *Anisotropic finite elements: Local estimates and applications*. Advances in Numerical Mathematics, Teubner, Stuttgart, 1999.

[3] T. Apel, S. Grosman, P.K. Jimack, and A. Meyer. A new methodology for anisotropic mesh refinement based upon error gradients. *Appl. Numer. Math.*, 50:329–341, 2004.

[4] T. Apel and G. Lube. Anisotropic mesh refinement in stabilized Galerkin methods. *Numer. Math.*, 74:261–282, 1996.

[5] R. Becker and R. Rannacher. An optimal control approach to a-posteriori error estimation in finite element methods. In A. Iserles, editor, *Acta Numerica*. Cambridge University Press, 2001.

[6] R. Biswas, K.D. Devine, and J.E. Flaherty. Parallel, adaptive finite element methods for conservation laws. *Appl. Numer. Math.*, 14:255–283, 1994.

[7] W. Cao. On the error of linear interpolation and the orientation, aspect ratio, and internal angles of a triangle. *SIAM J. Numer. Anal.*, 43(1):19–40, 2005.

[8] K.D. Devine and J.E. Flaherty. Parallel adaptive *hp*–refinement techniques for conservation laws. *Appl. Numer. Math.*, 20(4):367–386, 1996.

[9] T. Eibner and J. M. Melenk. An adaptive strategy for *hp*-fem based on testing for analyticity. Technical Report 12/2004, University of Reading, Department of Mathematics, 2004.

[10] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. Introduction to adaptive methods for differential equations. In A. Iserles, editor, *Acta Numerica*, pages 105–158. Cambridge University Press, 1995.

[11] L. Formaggia and S. Perotto. New anisotropic a priori error estimates. *Numer. Math.*, 89:641–667, 2001.

[12] E.H. Georgoulis. *hp*–version interior penalty discontinuous Galerkin finite element methods on anisotropic meshes. *Int. J. Numer. Anal. Model.*, 3:52–79, 2006.

[13] E.H. Georgoulis, E. Hall, and P. Houston. Discontinuous Galerkin methods for advection–diffusion–reaction problems on anisotropically refined meshes. *Submitted for publication.*

[14] E.H. Georgoulis, E. Hall, and P. Houston. Discontinuous Galerkin methods on *hp*-anisotropic meshes I: A priori error analysis. *International Journal of Computing Science and Mathematics, to appear.*

[15] E.H. Georgoulis and A. Lasis. A note on the design of *hp*–version interior penalty discontinuous Galerkin finite element methods for degenerate problems. *IMA J. Numer. Anal.*, 26(2):381–390, 2006.

[16] W. Gui and I. Babuška. The h, p and h–p versions of the finite element method in 1 dimension. Part III. The adaptive h–p version. *Numer. Math.*, 49:659–683, 1986.

[17] K. Harriman, P. Houston, B. Senior, and E. Süli. *hp*–Version discontinuous Galerkin methods with interior penalty for partial differential equations with nonnegative characteristic form. In C.-W. Shu, T. Tang, and S.-Y. Cheng, editors, *Recent Advances in Scientific Computing and Partial Differential Equations. Contemporary Mathematics Vol. 330*, pages 89–119. AMS, 2003.

[18] R. Hartmann and P. Houston. Adaptive discontinuous Galerkin finite element methods for nonlinear hyperbolic conservation laws. *SIAM J. Sci. Comput.*, 24:979–1004, 2002.

[19] R. Hartmann and P. Houston. Goal-oriented a posteriori error estimation for multiple target functionals. In T.Y. Hou and E. Tadmor, editors, *Hyperbolic Problems: Theory, Numerics, Applications*, pages 579–588. Springer-Verlag, 2003.

[20] R. Hartmann and P. Houston. Symmetric interior penalty DG methods for the compressible Navier–Stokes equations II: Goal–oriented a posteriori error estimation. *Int. J. Numer. Anal. Model.*, 3(2):141–162, 2006.

[21] P. Houston, E.H. Georgoulis, and E. Hall. Adaptivity and a posteriori error estimation for DG methods on anisotropic meshes. In G. Lube and G. Rapin, editors, *Proceedings of the International Conference on Boundary and Interior Layers (BAIL) - Computational and Asymptotic Methods.* 2006.

[22] P. Houston, R. Rannacher, and E. Süli. A posteriori error analysis of stabilised finite element approximations of transport problems. *Comput. Methods Appl. Mech. Engrg.*, 190(11-12):1483–1508, 2000.

[23] P. Houston, D. Schötzau, and T. P. Wihler. An *hp*-adaptive mixed discontinuous Galerkin FEM for nearly incompressible linear elasticity. *Comput. Methods Appl. Mech. Engrg.*, 17(1):33–62, 2007.

[24] P. Houston, C. Schwab, and E. Süli. Discontinuous *hp*-finite element methods for advection–diffusion–reaction problems. *SIAM J. Numer. Anal.*, 39:2133–2163, 2002.

[25] P. Houston, B. Senior, and E. Süli. Sobolev regularity estimation for *hp*–adaptive finite element methods. In F. Brezzi, A. Buffa, S. Corsaro, and A. Murli, editors, *Numerical Mathematics and Advanced Applications ENUMATH 2001*, pages 631–656. Springer, 2003.

[26] P. Houston and E. Süli. *hp*–Adaptive discontinuous Galerkin finite element methods for hyperbolic problems. *SIAM J. Sci. Comput.*, 23:1225–1251, 2001.

[27] P. Houston and E. Süli. Stabilized *hp*-finite element approximation of partial differential equations with non-negative characteristic form. *Computing*, 66:99–119, 2001.

[28] P. Houston and E. Süli. A note on the design of *hp*–adaptive finite element methods for elliptic partial differential equations. *Comput. Methods Appl. Mech. Engrg.*, 194(2-5):229–243, 2005.

[29] W. Huang. Measuring mesh qualities and application to variational mesh adaptation. *SIAM J. Sci. Comput.*, 26:1643–1666, 2005.

[30] W. Huang. Metric tensors for anisotropic mesh generation. *J. Comput. Phys.*, 204:633–665, 2005.

[31] W. Huang. Mathematical principles of anisotropic mesh adaptation. *Commun. Comput. Phys.*, 1(2):276–310, 2006.

[32] G. Kunert. *A posteriori error estimation for anisotropic tetrahedral and triangular finite element meshes*. PhD thesis, TU Chemnitz, 1999.

[33] J.M. Melenk and CH. Schwab. An *hp* finite element method for convection-diffusion problems. *IMA J. Numer. Anal.*, 19:425–453, 1999.

[34] O.A. Oleinik and E.V. Radkevič. *Second Order Equations with Nonnegative Characteristic Form*. American Mathematical Society, Providence, R.I., 1973.

[35] R. Schneider and P. Jimack. Toward anisotropic mesh adaptation based upon sensitivity of a posteriori estimates. Technical Report 2005.03, School of Computing, University of Leeds, 2005.

[36] C. Schwab. *p- and hp-FEM – Theory and Application to Solid and Fluid Mechanics*. Oxford University Press, Oxford, 1998.

[37] K.G. Siebert. An a posteriori error estimator for anisotropic refinement. *Numer. Math.*, 73:373–398, 1996.

[38] E. Süli and P. Houston. Adaptive finite element approximation of hyperbolic problems. In T. Barth and H. Deconinck, editors, *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics. Lect. Notes Comput. Sci. Engrg.*, volume 25, pages 269–344. Springer, 2002.

[39] E. Süli, P. Houston, and Ch. Schwab. *hp*–Finite element methods for hyperbolic problems. In J.R. Whiteman, editor, *The Mathematics of Finite Elements and Applications X*, pages 143–162. Elsevier, 2000.

[40] E. Süli, P. Houston, and B. Senior. *hp*–Discontinuous Galerkin finite element methods for nonlinear hyperbolic problems. *Internat. J. Numer. Methods Fluids*, 40(1-2):153–169, 2002.

[41] B. Szabó and I. Babuška. *Finite Element Analysis*. J. Wiley & Sons, New York, 1991.

[42] R. Verfürth. *A Review of a Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*. B.G. Teubner, Stuttgart, 1996.