Adaptive Refinement for hp–Version Trefftz Discontinuous Galerkin Methods for the Homogeneous Helmholtz Problem

Scott Congreve,* Paul Houston,† Ilaria Perugia‡

Abstract

In this article we develop an hp–adaptive refinement procedure for Trefftz discontinuous Galerkin methods applied to the homogeneous Helmholtz problem. Our approach combines not only mesh subdivision (h–refinement) and local basis enrichment (p–refinement), but also incorporates local directional adaptivity, whereby the elementwise plane wave basis is aligned with the dominant scattering direction. Numerical experiments based on employing an empirical *a posteriori* error indicator clearly highlight the efficiency of the proposed approach for various examples.

Keywords Homogeneous Helmholtz problem, Discontinuous Galerkin methods, Trefftz methods, Adaptivity, hp-finite element methods

Mathematics Subject Classification (2010) 65N30, 65N50, 35J05

1 Introduction

Trefftz discontinuous Galerkin (TDG) methods are finite element schemes which employ discontinuous test and trial functions whose restriction to each mesh element belongs to the kernel of the differential operator to be discretized. For time-harmonic wave problems, Trefftz discretization spaces are made of oscillating functions with the same frequency as that of the underlying analytical solution. This results in improved approximation properties, as compared to standard piecewise polynomial spaces. Moreover, based on Trefftz spaces, one can construct discontinuous Galerkin methods which feature unconditional unique solvability, as well as coercivity of the discrete bilinear forms in suitable (mesh-dependent) norms. We focus here on the case of the Helmholtz problem and refer, e.g., to the survey [19] for a review of the construction, properties, and relevant literature of Trefftz methods for its numerical approximation.

The purpose of this article is to develop an efficient hp–adaptive refinement algorithm for TDG methods applied to the homogeneous Helmholtz problem; we will specifically consider the ultra-weak variational formulation with plane wave basis functions [9]. Within the adaptive procedure, elements will be marked for refinement based on employing an empirical *a posteriori* error indicator, stimulated by the upper bounds derived in [23] for the h–version of the TDG method. For the h–version of the plane wave discontinuous Galerkin method, incorporating Lagrange multipliers, a similar error indicator has been presented in [3]. Once an element has been marked for refinement, a decision must then be made regarding the type of refinement to be undertaken, i.e., whether the element should be subdivided (h–refinement), or whether the local basis should be enriched (p–refinement). The choice of whether to h– or p–refine an element is typically based on the observation that when the underlying solution is smooth, then p–refinement will be more efficient

---

*Faculty of Mathematics, University of Vienna, Oskar-Morgenstern-Platz 1, 1090 Vienna, Austria (`scott.congreve@univie.ac.at`)

†School of Mathematical Sciences, University of Nottingham, University Park, Nottingham, NG7 2RD, UK (`Paul.Houston@nottingham.ac.uk`)

‡Faculty of Mathematics, University of Vienna, Oskar-Morgenstern-Platz 1, 1090 Vienna, Austria (`ilaria.perugia@univie.ac.at`)

in terms of reducing the error, for a given increase in the number of degrees of freedom, than if the element is subdivided. On the other hand, if the solution is not smooth, then h–refinement should be employed. In general, *a posteriori* error estimators only provide an estimate of the local elementwise error, but do not indicate which type of refinement should be employed. Within the existing literature a number of algorithms have been devised for determining the type of refinement (h– or p–) to be undertaken. For a comprehensive review of this subject, we refer to [26, 27], and the references cited therein. In the present context, given the oscillatory nature of solutions to high-frequency scattering problems, the exploitation of hp–strategies based on local regularity estimation techniques is not generally applicable. Thereby, we consider an alternative approach based on estimating the predicted decay rate of the *a posteriori* estimator, given the refinement history of each element; see, for example, [25]. For *a posteriori* error estimation of conforming finite element approximations of the Helmholtz problem, we refer, e.g., to [4, 5] and [11]; analogous bounds have been established for polynomial-based discontinuous Galerkin finite element methods in [29, 32].

In addition to standard h– and hp–adaptivity, we also consider the issue of directional refinement of the underlying plane wave basis employed within our TDG scheme. In particular, we rotate the underlying elementwise plane wave basis in order that the first basis function is aligned with the local dominant propagation direction; strategies for determining the local dominant propagation direction have been proposed in [2, 6, 7, 15], for example. Stimulated by the work undertaken on anisotropic mesh adaptation in [12, 13], cf., also, [14, 17], we propose an alternative approach based on studying the properties of the Hessian of the computed TDG solution. More precisely, the principal eigenvector of the Hessian of the solution indicates the dominant direction of wave propagation. However, since eigenvectors are only unique up to scalar multiples, the precise wave direction must be fixed, based on exploiting an impedance condition. In this way, we can locally orientate the elementwise plane wave basis to reduce the error in the underlying computed TDG solution in a simple and computationally cheap manner. When combined with hp–refinement, the resulting adaptive procedure is capable of generating highly optimized hp–refined Trefftz spaces. Indeed, the efficiency of the proposed strategy is illustrated for a number of test problems, where we compare the performance between an h– and hp–refinement algorithm, both with and without directional adaptivity.

The outline of this article is as follows: in Section 2 we introduce the model problem to be studied within this article, together with its TDG discretization. Then in Section 3 we develop an hp–refinement algorithm, based on employing both local mesh subdivision and local basis enrichment, together with directional adaptivity for the underlying Trefftz space. The performance of this procedure is studied in Section 4 through a series of two– and three–dimensional examples. Finally, in Section 5 we summarize the work undertaken within this article and highlight potential future directions of research.

2 Model problem and TDG discretization

In this section we state the model problem to be studied in this article, together with its TDG discretization; for further details, we refer to [19], for example.

2.1 Model problem

We study the homogeneous Helmholtz equation; to this end, we let $\Omega \subset \mathbb{R}^d$, $d = 2, 3$, be an open bounded, Lipschitz domain with boundary $\partial\Omega$. Thereby, we seek $u : \Omega \mapsto \mathbb{C}$ such that

$$
\begin{aligned}
-\Delta u - k^2 u &= 0 && \text{in } \Omega \,, \\
\frac{\partial u}{\partial \boldsymbol{n}} + ik\vartheta u &= g_R && \text{on } \Gamma_R \,, \\
u &= g_D && \text{on } \Gamma_D \,,
\end{aligned}
\tag{1}
$$

where $\boldsymbol{n}$ denotes the unit outward normal vector on the boundary $\partial\Omega$, and Γ_R and Γ_D are non-overlapping open subsets of $\partial\Omega$, such that $\partial\Omega = \overline{\Gamma}_R \cup \overline{\Gamma}_D$. Furthermore, i is the imaginary unit, $\vartheta = \pm 1$, $g_R \in L^2(\Gamma_R)$, and we assume, for the moment, that the (real-valued) wavenumber k is constant in Ω.

2.2 Meshes and spaces

We partition Ω into computational meshes $\{\mathcal{T}_h\}_{h>0}$ consisting of non-overlapping (curvilinear) polygons/polyhedra K, which potentially include hanging nodes, such that $\overline{\Omega} = \bigcup_{K \in \mathcal{T}_h} \overline{K}$. Moreover, we assume that the family of subdivisions $\{\mathcal{T}_h\}_{h>0}$ is shape regular [8, pp. 61, 114, and 118]. For each element $K \in \mathcal{T}_h$, we write h_K to denote its diameter and $\boldsymbol{n}_K$ signifies the unit outward normal vector to K on ∂K; we set $h := \max_{K \in \mathcal{T}_h} h_K$. Furthermore, we introduce the mesh skeleton $\mathcal{F}_h$, defined by $\mathcal{F}_h = \cup_{K \in \mathcal{T}_h} \partial K$; we write $\mathcal{F}_h^I$ and $\mathcal{F}_h^B$ to denote the interior and boundary skeletons, respectively, defined by $\mathcal{F}_h^I = \mathcal{F}_h \setminus \partial\Omega$ and $\mathcal{F}_h^B = \partial\Omega$. Implicitly, we assume that the finite element mesh $\mathcal{T}_h$ respects the decomposition of the boundary, in the sense that, given an element face $f \subset \partial K$, $K \in \mathcal{T}_h$, which lies on the boundary $\partial\Omega$, i.e., $f \subset \partial\Omega$, then f is entirely contained within either Γ_R or Γ_D.

Let K and K' be two adjacent elements of $\mathcal{T}_h$, and $\boldsymbol{x}$ an arbitrary point on the interior face $f \subset \mathcal{F}_h^I$ given by $f = (\partial K \cap \partial K')^{\circ}$. Furthermore, let v and $\boldsymbol{w}$ be scalar- and vector-valued functions, respectively, that are sufficiently smooth inside each element K, K'. Then, the averages of v and $\boldsymbol{w}$ at $\boldsymbol{x} \in f$ are given by

$$\{\!\{v\}\!\} = \frac{1}{2}(v|_K + v|_{K'}), \qquad \{\!\{\boldsymbol{w}\}\!\} = \frac{1}{2}(\boldsymbol{w}|_K + \boldsymbol{w}|_{K'}),$$

respectively. Similarly, the jumps of v and $\boldsymbol{w}$ at $\boldsymbol{x} \in f$ are given by

$$[\![v]\!] = v|_K\, \boldsymbol{n}_K + v|_{K'}\, \boldsymbol{n}_{K'}, \qquad [\![\boldsymbol{w}]\!] = \boldsymbol{w}|_K \cdot \boldsymbol{n}_K + \boldsymbol{w}|_{K'} \cdot \boldsymbol{n}_{K'},$$

respectively.

Given $K \in \mathcal{T}_h$ the local Trefftz space is defined by

$$T(K) := \{v \in H^1(K) : -\Delta v - k^2 v = 0\};$$

with this notation, we write

$$T(\mathcal{T}_h) := \{v \in L^2(\Omega) : v|_K \in T(K), K \in \mathcal{T}_h\}.$$

Thereby, given a local space $V_{p_K}(K) \subset T(K)$, of finite dimension $p_K \geq 1$, the corresponding TDG finite element space is defined by

$$V_{\boldsymbol{p}}(\mathcal{T}_h) := \{v \in T(\mathcal{T}_h) : v|_K \in V_{p_K}(K), K \in \mathcal{T}_h\},$$

where $\boldsymbol{p} = \{p_K : K \in \mathcal{T}_h\}$.

2.3 TDG discretization

Equipped with the TDG finite element space $V_{\boldsymbol{p}}(\mathcal{T}_h)$ defined on the mesh partition $\mathcal{T}_h$ of Ω, the TDG approximation of (1) is given by: find $u_{hp} \in V_{\boldsymbol{p}}(\mathcal{T}_h)$ such that

$$\mathcal{A}_h(u_{hp}, v_{hp}) = \ell_h(v_{hp}) \tag{2}$$

for all $v_{hp} \in V_{\boldsymbol{p}}(\mathcal{T}_h)$, where

$$
\begin{aligned}
\mathcal{A}_h(u,v) &= \int_{\mathcal{F}_h^I} \left(\{\!\!\{u\}\!\!\} [\![\nabla_h \overline{v}]\!] - \beta(ik)^{-1} [\![\nabla_h u]\!] [\![\nabla_h \overline{v}]\!] - \{\!\!\{\nabla_h u\}\!\!\} \cdot [\![\overline{v}]\!] + \alpha ik [\![u]\!] \cdot [\![\overline{v}]\!] \right) \mathrm{d}s \\
&\quad + \int_{\Gamma_R} \left((1-\delta)(u\nabla_h \overline{v} \cdot \boldsymbol{n} + ik\vartheta u\overline{v}) - \delta((ik\vartheta)^{-1}(\nabla_h u \cdot \boldsymbol{n})(\nabla_h \overline{v} \cdot \boldsymbol{n}) + \nabla_h u \cdot \boldsymbol{n}\, \overline{v}) \right) \mathrm{d}s \\
&\quad + \int_{\Gamma_D} (-\nabla_h u \cdot \boldsymbol{n}\, \overline{v} + \alpha ik u\overline{v}) \, \mathrm{d}s, \\
\ell_h(v) &= \int_{\Gamma_R} g_R((1-\delta)\overline{v} - \delta(ik\vartheta)^{-1}\nabla_h \overline{v} \cdot \boldsymbol{n}) \, \mathrm{d}s + \int_{\Gamma_D} g_D(\alpha ik\overline{v} - \nabla_h \overline{v} \cdot \boldsymbol{n}) \, \mathrm{d}s,
\end{aligned}
$$

and ∇_h denotes the broken gradient operator, defined elementwise. Here, $\alpha > 0$, $\beta > 0$ and $0 < \delta \leq 1/2$ are given penalty parameters. We note that the selection of these penalty parameters has been studied in a number of different contexts within the literature; in particular, here we mention the ultra-weak variational formulation (UWVF), cf. [9], the DG-type scheme studied in [16], and [18] which considered their selection on locally refined meshes; cf. [19, Table 1]. We note that the *a priori* analysis presented in [18] covers the case when the underlying computational mesh contains hanging nodes; in that setting, the penalty parameters were chosen to depend on the local mesh size. For the purposes of this article we consider the UWVF, corresponding to the choice $\alpha = \beta = \delta = 1/2$. We note that, in our computational work, we have not observed any quantitative differences in the performance of the underlying numerical scheme when employing this latter selection of the penalty parameters, when compared to those proposed in [18].

2.4 Plane wave basis functions

Finally, in this section we outline the choice of the underlying discrete space $V_{p_K}(K)$, $K \in \mathcal{T}_h$. To this end, we select $V_{p_K}(K)$ to be a local space consisting of plane waves in p_K different directions, all with the same wavenumber k. We note that, under suitable assumptions on K and the choice of plane wave directions, $V_{p_K}(K)$ approximates smooth Trefftz functions with the same order of convergence as polynomials of degree q_K, where

$$
p_K = \begin{cases} 2q_K + 1, & d = 2, \\ (q_K + 1)^2, & d = 3; \end{cases} \tag{3}
$$

see [28]. Thereby, q_K is referred to as the *effective polynomial degree* of the discrete Trefftz space; we set $\boldsymbol{q} = \{q_K : K \in \mathcal{T}_h\}$. More precisely, we write

$$
V_{p_K}(K) := \left\{ v \in T(K) : v(\boldsymbol{x}) = \sum_{\ell=0}^{p_K-1} \alpha_\ell \mathrm{e}^{ik\boldsymbol{d}_{K,\ell} \cdot (\boldsymbol{x}-\boldsymbol{x}_K)}, \alpha_\ell \in \mathbb{C} \right\}, \tag{4}
$$

where $\boldsymbol{x}_K$ is the center of mass of element K and $\boldsymbol{d}_{K,\ell}$, $\ell = 0, \ldots, p_K - 1$, are p_K evenly distributed unit direction vectors (with respect to the unit ball). For $d = 2$ we can simply define

$$
\boldsymbol{d}_{K,\ell} = (\cos(2\pi\ell/p_K), \sin(2\pi\ell/p_K))^\top, \qquad \ell = 0, \ldots, p_K - 1; \tag{5}
$$

for $d = 3$ we employ the directions determined by the extremal (maximum determinant) points on S^2, cf. [30, 31].

3 Adaptive mesh refinement

In this section we develop an automatic adaptive refinement algorithm which is capable of not only marking elements for refinement, but also determining the type of refinement to be undertaken. In particular, here we consider both h– and p–refinement, whereby the local element is subdivided, or the number of elementwise plane wave directions is enriched, respectively, as well as directional refinement which seeks to rotate the local plane wave basis in order to align it with the principal scattering direction.

3.1 *A posteriori* error indicator

In the absence of rigorous *a posteriori* error bounds for the numerical approximation of (1) by the TDG scheme (2), which are sharp with respect to both the local mesh size h_K and the number of local plane waves p_K employed on each element $K \in \mathcal{T}_h$, we employ an empirical error estimator stimulated by the work undertaken in [23] in the h–version setting. To this end, we first introduce the dual problem: find $z \in H^1(\Omega)$, such that

$$-\Delta z - k^2 z = u - u_{hp} \quad \text{in } \Omega ,$$

$$\frac{\partial z}{\partial \boldsymbol{n}} + ik\vartheta z = 0 \qquad \text{on } \Gamma_R ,$$

$$z = 0 \qquad \text{on } \Gamma_D .$$

Noting that $z \in H^{2/3+s}(\Omega)$, $0 < s \leq 1/2$, cf. [18], we recall the following (second) *a posteriori* error bound from [23].

Theorem 3.1. *Assume that the mesh $\mathcal{T}_h$ is shape-regular, locally quasi-uniform, in the sense that, for two elements K and K' which share a face $f \subset \mathcal{F}_h^I$, there is a constant τ, independent of h, such that*

$$\tau^{-1} \leq h_K / h_{K'} \leq \tau$$

for all choices of K and K', and that $\mathcal{T}_h$ is quasi-uniform in the vicinity of Γ_R, i.e., for all $K \in \mathcal{T}_h$ which lie on the boundary Γ_R, i.e., so that $\partial K \cap \Gamma_R \neq \emptyset$, there exists τ_R such that

$$h / h_K \leq \tau_R.$$

Then, for $g_D \equiv 0$ and fixed p_K, $K \in \mathcal{T}_h$, the following a posteriori *bound holds:*

$$\|u - u_{hp}\|_{L^2(\Omega)} \leq \mathfrak{E}(u_{hp}, h) \equiv C \left(\sum_{K \in \mathcal{T}_h} \eta_K^2 \right)^{1/2} ,$$

where

$$\eta_K^2 = \left\| \alpha^{1/2} h_K^s [\![u_{hp}]\!] \right\|_{L^2(\partial K \backslash \partial \Omega)}^2 + k^{-2} \left\| \beta^{1/2} h_K^s [\![\nabla u_{hp}]\!] \right\|_{L^2(\partial K \backslash \partial \Omega)}^2$$
$$+ k^{-2} \left\| \delta^{1/2} h_K^s (g_R - \nabla u_{hp} \cdot \boldsymbol{n}_K + iku_{hp}) \right\|_{L^2(\partial K \cap \Gamma_R)}^2 + \left\| \alpha^{1/2} h_K^s u_{hp} \right\|_{L^2(\partial K \cap \Gamma_D)}^2 , \tag{6}$$

where C is a positive constant, which is independent of h.

We stress that the *a posteriori* error bound stated in Theorem 3.1 depends on the regularity index s; thereby, *a priori* knowledge of s is required in order to yield a fully computable bound. From the proof presented in [23], it is clear that the constant C in the reliability estimate reported in Theorem 3.1 is independent of $\boldsymbol{p}$, or equivalently $\boldsymbol{q}$. However, we wish to determine the scaling for each of the terms present in (6) in order to obtain efficiency also in $\boldsymbol{p}$. Thereby, we propose the following *empirical* error estimator, where for simplicity of notation, we also denote it by $\mathfrak{E}$, for the hp–version TDG method:

$$\mathfrak{E}(u_h, h, \boldsymbol{p}) = \left(\sum_{K \in \mathcal{T}_h} \eta_K^2 \right)^2 , \tag{7}$$

where

$$\eta_K^2 = \left\| \alpha^{1/2} h_K^{1/2} q_K^{-1/2} [\![u_{hp}]\!] \right\|_{L^2(\partial K \backslash \partial \Omega)}^2 + \left\| \beta^{1/2} h_K^{3/2} q_K^{-3/2} [\![\nabla u_{hp}]\!] \right\|_{L^2(\partial K \backslash \partial \Omega)}^2$$
$$+ \left\| \delta^{1/2} h_K^{3/2} q_K^{-3/2} (g_R - \nabla u_{hp} \cdot \boldsymbol{n}_F + iku_{hp}) \right\|_{L^2(\partial K \cap \Gamma_R)}^2 + \left\| \alpha^{1/2} h_K^{1/2} q_K^{-1/2} (g_D - u_{hp}) \right\|_{L^2(\partial K \cap \Gamma_D)}^2 . \tag{8}$$

We stress that the choice of the exponents of h_K and q_K have been selected on the basis of numerical experimentation on a problem with a smooth analytical solution; for details, see Section 4.2 below. Compared to the error indicator (6) from [23], we note that we have a factor of h instead of k^{-1} in the terms with $[\![\nabla u_{hp}]\!]$ and in the Robin boundary terms. These are both dimensionally correct, but reproducing the numerical experiments conducted in Section 4.2 with the dependency on k^{-1} results in different effectivities for different wavenumbers k.

3.2 Plane wave directional adaptivity

In this section, we discuss the design of a practical algorithm for determining the direction vectors $\boldsymbol{d}_{K,\ell}$, $\ell = 0, \ldots, p_K - 1$, used to define the plane wave basis within each element K in the computational mesh $\mathcal{T}_h$. The key observation is that, many wave propagation problems typically exhibit a dominant direction of propagation of the underlying wave within each element in $\mathcal{T}_h$. Thereby, by aligning the plane wave basis in an appropriate fashion, we expect to attain a significant reduction of the error in the computed TDG solution. Indeed, in the simple case when the analytical solution is a plane wave, then if the direction for one of the plane wave basis functions is selected such that it is aligned with this plane wave direction, then the TDG method will exactly recover the analytical solution, subject to rounding errors.

The essential idea here is to simply rotate the element basis according to the predicted elementwise dominant direction. For simplicity of presentation, let us consider the two-dimensional case, i.e., $d = 2$; we note that $d = 3$ follows in an analogous manner, cf. Remarks 3.2 & 3.3 below. In two-dimensions, the standard plane wave directions are generally selected to be evenly spaced, with the first direction $\boldsymbol{d}_{K,0} = (1,0)^\top$ always pointing along the x-axis, cf. (5) (in the three-dimensional setting, the first direction vector typically points along the z-axis). Alternatively, assuming that a dominant elementwise direction, denoted by $\boldsymbol{\mathfrak{d}}_K$, can be determined within each $K \in \mathcal{T}_h$, then the direction vectors for the plane wave basis functions in K are chosen such that the first plane wave direction is aligned with $\boldsymbol{\mathfrak{d}}_K$, i.e., (5) is replaced by

$$\boldsymbol{d}_{K,\ell} = \left(\cos(2\pi\ell/p_K + \theta_K), \sin(2\pi\ell/p_K + \theta_K)\right)^\top, \tag{9}$$

$\ell = 0, \ldots, p_K - 1$, where θ_K is the angle between $\boldsymbol{\mathfrak{d}}_K$ and the x-axis.

Clearly, in general, the dominant elementwise direction $\boldsymbol{\mathfrak{d}}_K$, $K \in \mathcal{T}_h$, cannot be determined *a priori*, but instead must be numerically estimated as part of the solution process. In this regard, a number of algorithms have been proposed within the literature; here we mention the ray-tracing approach developed in [6, 7], though this includes terms involving integrals over the elements within the underlying TDG formulation. In [2], the optimal angle of rotation was numerically estimated based on adding an extra unknown into the problem; however, this leads to a system of nonlinear equations to be computed. The selection of the plane wave directions by solving an optimal control problem was proposed in [1], while [15] uses an approximation of

$$\frac{\nabla e(\boldsymbol{x}_0)}{ike(\boldsymbol{x}_0)},$$

at a given point $\boldsymbol{x}_0 \in K$, $K \in \mathcal{T}_h$, where e denotes the error.

Stimulated by the work undertaken in [12, 13], cf. also [14, 17], on the design of anisotropically refined computational meshes, in this section we compute an estimate of $\boldsymbol{\mathfrak{d}}_K$, $K \in \mathcal{T}_h$, based on the properties of the Hessian of the TDG solution u_{hp}. Indeed, we note that the principal eigenvector, i.e., the eigenvector corresponding to the largest eigenvalue in absolute value, of the Hessian of a given function indicates the direction of most rapid variation, and thereby, in our context, the dominant direction of wave propagation. With this in mind, writing $\boldsymbol{\mathcal{H}}(\varphi, \boldsymbol{x}_0)$ to denote the Hessian matrix of a given function φ, evaluated at the point $\boldsymbol{x}_0 \in \mathbb{R}^d$, in Algorithm 1 we outline the steps involved in computing a *potential* dominant plane wave direction $\hat{\boldsymbol{\mathfrak{d}}}_K$ for a given element $K \in \mathcal{T}_h$. Table 1 summarizes how this potential first plane wave direction $\hat{\boldsymbol{\mathfrak{d}}}_K$ is selected; for the numerical experiments presented in Section 4, we set $\Lambda = 2$. We note that in the case when no primary propagation direction is determined, then we leave the first plane wave direction unchanged.

Algorithm 1 Computation of the potential first plane wave direction $\hat{\mathfrak{d}}_K$ for element K.

1: Input: the TDG solution u_{hp} of the discrete problem (2) and the parameter $\Lambda > 1$.
2: Writing $\boldsymbol{x}_K$ to denote the centroid of K, $K \in \mathcal{T}_h$, evaluate the eigenpairs $(\lambda_1, \boldsymbol{v}_1), (\lambda_2, \boldsymbol{v}_2)$ of $\boldsymbol{\mathcal{H}}(\mathrm{Re}(u_{hp}|_K), \boldsymbol{x}_K)$, and $(\mu_1, \boldsymbol{w}_1), (\mu_2, \boldsymbol{w}_2)$ of $\boldsymbol{\mathcal{H}}(\mathrm{Im}(u_{hp}|_K), \boldsymbol{x}_K)$, such that $|\lambda_1| \geq |\lambda_2|$ and $|\mu_1| \geq |\mu_2|$.
3: **if** $|\lambda_1| \geq \Lambda|\lambda_2|$ **then**
4:     **if** $|\mu_1| \geq \Lambda|\mu_2|$ **then**
5:         **if** $|\lambda_1| \geq \Lambda|\mu_1|$ **then**
6:             $\hat{\mathfrak{d}}_K \leftarrow \boldsymbol{v}_1$
7:         **else if** $|\mu_1| \geq \Lambda|\lambda_1|$ **then**
8:             $\hat{\mathfrak{d}}_K \leftarrow \boldsymbol{w}_1$
9:         **else**
10:             $\hat{\mathfrak{d}}_K \leftarrow \frac{\boldsymbol{v}_1 + \boldsymbol{w}_1}{\|\boldsymbol{v}_1 + \boldsymbol{w}_1\|}$
11:         **end if**
12:     **else**
13:         **if** $|\lambda_1| \geq \Lambda|\mu_1|$ **then**
14:             $\hat{\mathfrak{d}}_K \leftarrow \boldsymbol{v}_1$
15:         **else**
16:             No primary propagation direction
17:         **end if**
18:     **end if**
19: **else**
20:     **if** $|\mu_1| \geq \Lambda|\mu_2|$ **then**
21:         **if** $|\mu_1| \geq \Lambda|\lambda_1|$ **then**
22:             $\hat{\mathfrak{d}}_K \leftarrow \boldsymbol{w}_1$
23:         **else**
24:             No primary propagation direction
25:         **end if**
26:     **else**
27:         No primary propagation direction
28:     **end if**
29: **end if**

$|\lambda_1| \geq \Lambda|\lambda_2|$	$|\mu_1| \geq \Lambda|\mu_2|$	$|\lambda_1| \geq \Lambda|\mu_1|$	$|\mu_1| \geq \Lambda|\lambda_1|$	First Plane Wave $\hat{\mathfrak{d}}_K$
✓	✓	✓	✗	$\boldsymbol{v}_1$
✓	✓	✗	✓	$\boldsymbol{w}_1$
✓	✓	✗	✗	$\frac{(\boldsymbol{v}_1 + \boldsymbol{w}_1)}{\|\boldsymbol{v}_1 + \boldsymbol{w}_1\|}$
✓	✗	✓	✗	$\boldsymbol{v}_1$
✓	✗	✗	—	—
✗	✓	✗	✓	$\boldsymbol{w}_1$
✗	✓	—	✗	—
✗	✗	—	—	—

Table 1: Summary of selection of first plane wave direction $\hat{\mathfrak{d}}_K$ using Algorithm 1.

Remark 3.2. We note that in the case when $d = 3$, $\boldsymbol{\mathcal{H}}(\mathrm{Re}(u_{hp}|_K), \boldsymbol{x}_K)$ and $\boldsymbol{\mathcal{H}}(\mathrm{Im}(u_{hp}|_K), \boldsymbol{x}_K)$ each have a third eigenpair, $(\lambda_3, \boldsymbol{v}_3)$ and $(\mu_3, \boldsymbol{w}_3)$, respectively. However, if the eigenpairs are sorted such that $|\lambda_1| \geq |\lambda_2| \geq |\lambda_3|$ and $|\mu_1| \geq |\mu_2| \geq |\mu_3|$, the third eigenpairs *never* represent a dominant direction, and thereby Algorithm 1 can be used to identify $\hat{\mathfrak{d}}_K$, $K \in \mathcal{T}_h$, without modification.

Noting that eigenvectors are only unique up to scalar multiples, the vector $\hat{\mathfrak{d}}_K$, $K \in \mathcal{T}_h$, evaluated according to Algorithm 1 may be pointing in precisely the opposite direction to the

---

Algorithm 2 Evaluation of the first plane wave direction $\boldsymbol{\mathfrak{d}}_K$ for element K.

1: Input: the TDG solution u_{hp} of the discrete problem (2), the parameter $0 \le \delta \to 0$, and $\hat{\boldsymbol{\mathfrak{d}}}_K$ computed by Algorithm 1.

2: The first plane wave direction $\boldsymbol{\mathfrak{d}}_K$ on element K, $K \in \mathcal{T}_h$, is given by

$$
\boldsymbol{\mathfrak{d}}_K = \begin{cases} -\hat{\boldsymbol{\mathfrak{d}}}_K, & \text{if } \operatorname{Re}\left(\frac{\nabla u_{hp}(\boldsymbol{x}_K + \delta\hat{\boldsymbol{\mathfrak{d}}}_K) \cdot \hat{\boldsymbol{\mathfrak{d}}}_K + iku_{hp}(\boldsymbol{x}_K + \delta\hat{\boldsymbol{\mathfrak{d}}}_K)}{ik} \right) < \mathrm{e}^{ik\delta}, \\ \hat{\boldsymbol{\mathfrak{d}}}_K, & \text{if } \operatorname{Re}\left(\frac{\nabla u_{hp}(\boldsymbol{x}_K + \delta\hat{\boldsymbol{\mathfrak{d}}}_K) \cdot \hat{\boldsymbol{\mathfrak{d}}}_K + iku_{hp}(\boldsymbol{x}_K + \delta\hat{\boldsymbol{\mathfrak{d}}}_K)}{ik} \right) \ge \mathrm{e}^{ik\delta}. \end{cases}
$$

---

primary wave propagation direction. Thereby, to ensure that $\hat{\boldsymbol{\mathfrak{d}}}_K$, $K \in \mathcal{T}_h$, is correctly oriented, we study the impedance trace on the boundary of a ball $B_\delta(\boldsymbol{x}_K)$ of radius δ, centered at $\boldsymbol{x}_K$, of both the numerical solution and a plane wave with (the desired) propagation direction $\boldsymbol{\mathfrak{d}}_K$. As we let $\delta \to 0$, we expect that the numerical solution should be closely approximated by the plane wave in the primary propagation direction.

More precisely, given $K \in \mathcal{T}_h$, the impedance trace of the plane wave

$$
\tilde{u}_K(\boldsymbol{x}) = \mathrm{e}^{ik\boldsymbol{\mathfrak{d}}_K \cdot (\boldsymbol{x} - \boldsymbol{x}_K)}
$$

on $\partial B_\delta(\boldsymbol{x}_K)$ is given by

$$
(\nabla \tilde{u}_K(\boldsymbol{x}) \cdot \boldsymbol{n}_{B_\delta} + ik\tilde{u}_K(\boldsymbol{x}))|_{\partial B_\delta(\boldsymbol{x}_K)} = (ik(\boldsymbol{\mathfrak{d}}_K \cdot \boldsymbol{n}_{B_\delta} + 1)\mathrm{e}^{ik\boldsymbol{\mathfrak{d}}_K \cdot (\boldsymbol{x} - \boldsymbol{x}_K)})|_{\partial B_\delta(\boldsymbol{x}_K)}, \quad (10)
$$

where $\boldsymbol{n}_{B_\delta}$ denotes the unit outward normal vector on $\partial B_\delta(\boldsymbol{x}_K)$. Setting $\boldsymbol{x} = \boldsymbol{x}_K + \delta\hat{\boldsymbol{\mathfrak{d}}}_K$ in (10) and noting that, at this point of evaluation, $\boldsymbol{n}_{B_\delta} = \hat{\boldsymbol{\mathfrak{d}}}_K$, we deduce that

$$
\frac{\nabla \tilde{u}_K(\boldsymbol{x}_K + \delta\hat{\boldsymbol{\mathfrak{d}}}_K) \cdot \boldsymbol{n}_{B_\delta} + ik\tilde{u}_K(\boldsymbol{x}_K + \delta\hat{\boldsymbol{\mathfrak{d}}}_K)}{ik} = \begin{cases} 2\mathrm{e}^{ik\delta}, & \text{if } \hat{\boldsymbol{\mathfrak{d}}}_K = \boldsymbol{\mathfrak{d}}_K, \\ 0, & \text{if } \hat{\boldsymbol{\mathfrak{d}}}_K = -\boldsymbol{\mathfrak{d}}_K. \end{cases}
$$

Thereby, the (potential) dominate direction of propagation $\hat{\boldsymbol{\mathfrak{d}}}_K$, $K \in \mathcal{T}_h$, predicted according to Algorithm 1 may be corrected to yield the dominant direction $\boldsymbol{\mathfrak{d}}_K$ on the basis of Algorithm 2; this direction will then be selected as the first plane wave direction on element K, $K \in \mathcal{T}_h$. For simplicity, throughout this article we set $\delta = 0$.

Remark 3.3. In the three–dimensional setting, once the selection of the primary wave propagation direction $\boldsymbol{\mathfrak{d}}_K$ has been computed on the basis of Algorithms 1 & 2, we then select the remaining wave directions, $\boldsymbol{d}_{K,\ell}$, $\ell = 1, \ldots, p_K - 1$, by applying a transformation matrix $T \in \mathbb{R}^{3 \times 3}$ to the original 'reference' directions $\tilde{\boldsymbol{d}}_{K,\ell}$, $\ell = 1, \ldots, p_K - 1$, respectively, where $\tilde{\boldsymbol{d}}_{K,0}$ points along the z-axis, cf. above. Thereby,

$$
\boldsymbol{d}_{K,\ell} = T\tilde{\boldsymbol{d}}_{K,\ell},
$$

$\ell = 1, \ldots, p_K - 1$, where T is selected such that

$$
\boldsymbol{\mathfrak{d}}_K \equiv \boldsymbol{d}_{K,0} = T \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \equiv T\tilde{\boldsymbol{d}}_{K,0}.
$$

We note that the selection of T is not unique; writing $\boldsymbol{\mathfrak{d}}_K = (d_x, d_y, d_z)^\top$, we define T to be the identity matrix if $d_x = d_y = 0$; otherwise, we set

$$
T = \begin{pmatrix} \frac{d_x d_z}{\sqrt{d_x^2 + d_y^2}} & \frac{d_y}{\sqrt{d_x^2 + d_y^2}} & d_x \\ \frac{d_y d_z}{\sqrt{d_x^2 + d_y^2}} & -\frac{d_x}{\sqrt{d_x^2 + d_y^2}} & d_y \\ -\sqrt{d_x^2 + d_y^2} & 0 & d_z \end{pmatrix}.
$$

Algorithm 3 hp–Adaptive refinement algorithm.
___

1: Input the control parameters γ_h, γ_p, and γ_n.
2: Choose a coarse initial mesh $\mathcal{T}_{h,0}$ of Ω and a corresponding low-order starting (effective) polynomial degree vector $\boldsymbol{q}_0$, together with the total dimension vector $\boldsymbol{p}_0$ defined as in (3).
3: Set the initial predicted error indicator $\eta_{K,0}^{\mathrm{pred}} = \infty$ for all $K \in \mathcal{T}_{h,0}$.
4: **for** $i = 0, 1, \ldots$, until sufficiently many iterations have been performed. **do**
5:      Solve (2) for $u_{hp} \in V_{\boldsymbol{p}_i}(\mathcal{T}_{h,i})$.
6:      Compute the *a posteriori* error indicators $\eta_{K,i} \equiv \eta_K$, $K \in \mathcal{T}_{h,i}$, and mark elements for refinement based on their relative magnitude.
7:      **for** $K \in \mathcal{T}_{h,i}$ **do**
8:          **if** K is marked for refinement **then**
9:              **if** $\eta_{K,i} > \eta_{K,i}^{\mathrm{pred}}$ **then**
10:                 Perform h–refinement: Subdivide K into N children $K_s, s = 1, \ldots, N$, and set
11:                 $(\eta_{K_s,i+1}^{\mathrm{pred}})^2 \leftarrow \frac{1}{N}\gamma_h \left(\frac{1}{2}\right)^{2q_K} \eta_{K,i}^2$, $1 \le s \le N$.
12:              **else**
13:                 Perform p–refinement: $q_K \leftarrow q_K + 1$
14:                 $(\eta_{K,i+1}^{\mathrm{pred}})^2 \leftarrow \gamma_p \eta_{K,i}^2$
15:              **end if**
16:          **else**
17:              $(\eta_{K,i+1}^{\mathrm{pred}})^2 \leftarrow \gamma_n (\eta_{K,i}^{\mathrm{pred}})^2$
18:          **end if**
19:      **end for**
20:      Construct the new mesh $\mathcal{T}_{h,i+1}$ and corresponding Trefftz space $V_{\boldsymbol{p}_{i+1}}(\mathcal{T}_{h,i+1})$.
21: **end for**
___

3.3 hp–Adaptive mesh refinement

In this section we discuss the design of an automatic algorithm for generating sequences of hp–adaptively refined TDG finite element spaces in an efficient manner. This topic has been extensively studied within the finite element literature in the case when the local element spaces consist of polynomial functions; for a comprehensive review, we refer to [26, 27]. In general, the key underlying principle of most hp–refinement strategies is to employ local mesh subdivision (h–refinement) in regions where the solution is not smooth, while local enrichment of the finite element space (p–refinement) is undertaken elsewhere. Given that such regularity information is generally unknown *a priori*, several strategies have been developed to *a posteriori* estimate the local smoothness of the analytical solution, based on its numerical approximation; cf. [21], for example. However, in the context of TDG schemes for the numerical approximation of high-frequency time-harmonic wave problems, the extraction of such regularity information is expected to be unreliable due to the oscillatory nature of the computed numerical solution.

Thereby, as an alternative to directly estimating local smoothness of the solution, we employ the *a posteriori* error indicator (8) to select the type of refinement to be undertaken on the basis of the refinement history of the current element, cf. [25]. More precisely, following [25] refinements are selected based on checking if the local error estimate has decayed according to the expected rate of convergence based on the last type of refinement employed. If the expected rate of convergence is achieved, then p–refinement is performed; otherwise, h–refinement is undertaken. The variant of [25, Algorithm 4.4] we employ here is summarized in Algorithm 3. Here, we note that γ_h, γ_p, and γ_n are control parameters; for the purposes of this article, we select $\gamma_h = 4$, $\gamma_p = 0.4$, and $\gamma_n = 1$. Furthermore, the number of child elements, N, cf. step 10. in Algorithm 3, is dependent on the type of subdivision, i.e., isotropic/anisotropic, undertaken, as well as the element shape; for isotropic refinement of tensor-product elements, we have that $N = 2^d$.

Remark 3.4. In lines 11 and 14 of Algorithm 3 we use the same predictions as in a polynomial finite element method. This prediction is based on the *a priori* convergence rates, which in the

case of the TDG method with p plane waves, where $p = 2q + 1$ in 2D or $p = (q + 1)^2$ in 3D, are the same as in the case of a DG method with polynomials of degree q. For h-refinement, i.e., line 11, we write e_K to denote the $L^2(K)$-norm of the error in the computed numerical solution on element K, of size h_K, and $\widetilde{e}_K$ to be the square-root of sum of the squares of the L^2-norm errors computed in each of the N children of K, each of size $h_K/2$, namely $\widetilde{e}_K^2 = \sum_{s=1}^{N} e_{K_s}^2$. Then, we have the predicted ratio $\frac{\widetilde{e}_K}{e_K} = \left(\frac{1}{2}\right)^{q+1}$. Assuming the square of the error is equally distributed among the N children, we deduce the coefficient of $\frac{1}{N}\gamma_h \left(\frac{1}{2}\right)^{2q}$ in Line 11. For line 14, we assume exponential convergence of the error under p-refinement; hence, we expect the predicted error to reduce by a constant multiple.

Remark 3.5. We note that in [25] the initial values of the predicted error indicator $\eta_{K,0}^{\mathrm{pred}}$, $K \in \mathcal{T}_{h,0}$, are set to zero; thereby, this ensures that h–refinement is undertaken the first time an element is refined. In contrast, in Algorithm 3 we set $\eta_{K,0}^{\mathrm{pred}} = \infty$ for all $K \in \mathcal{T}_{h,0}$ which instead leads to p–enrichment being undertaken as the first refinement of a given element, since the TDG method for the numerical approximation of the Helmholtz equation is intrinsically a high-order method.

Remark 3.6. Plane wave directional adaptivity can be performed at different stages within Algorithm 3; for example, the following options are available:

- undertake directional adaptivity only on elements marked for p–refinement,

- undertake directional adaptivity on all elements marked for refinement, with h–refinement performed after plane wave direction adaptivity, or

- undertake directional adaptivity on every element $K \in \mathcal{T}_h$, even if the element K has not been marked for refinement.

In Section 4 we shall numerically investigate each of these approaches in order to assess their relative computational performance in terms of error reduction.

Remark 3.7. As a final remark, we note that within Algorithm 3 we employ the fixed fraction refinement strategy to select elements for refinement, cf. step 6; throughout this article we set the refinement fraction equal to 25%.

4   Numerical experiments

In this section, we present a series of numerical experiments to highlight the practical performance of the hp–refinement algorithm, with directional adaptivity, proposed in Algorithm 3. Throughout this section we shall compare the performance of the proposed hp–adaptive refinement strategy with the corresponding algorithm based on exploiting only local mesh subdivision, i.e., h–refinement. The numerical experiments presented within this section have been undertaken using the AptoFEM software package [20].

4.1   Plane wave direction adaptivity

In this first example, we study the effect of adjusting the plane wave directions while employing a fixed computational mesh with uniform p–refinement. To this end, we consider problem (1) with $\Omega = (0, 1)^2$, $\Gamma_R = \partial\Omega$, and $\Gamma_D \equiv \emptyset$; furthermore, the Robin boundary condition g_R is selected such that the analytical solution u of (1) is given by

$$u(x, y) = \mathcal{H}_0^{(1)}\left(k\sqrt{(x + 1/4)^2 + y^2}\right), \tag{11}$$

where $\mathcal{H}_0^{(1)}$ denotes the Hankel function of the first kind of order 0. Throughout this section, we set $k = 20$; note that for this problem the analytical solution u is smooth in Ω.

Here, the underlying computational mesh consists of 16 uniform square elements; on each element we initially select the effective polynomial degree $q = 2$, i.e., $p = 5$. In Table 2 we compare

| q | No of Dofs | Relative $L^2(\Omega)$-Error | | % Reduction |
		Standard TDG	Direction Adaptivity	
3	112	2.015×10^{0}	1.959×10^{0}	2.7%
4	144	5.027×10^{-1}	3.194×10^{-1}	36.5%
5	176	7.414×10^{-2}	2.658×10^{-2}	64.1%
6	208	1.616×10^{-2}	6.320×10^{-3}	60.9%
7	240	3.420×10^{-3}	1.435×10^{-3}	58.0%
8	272	5.154×10^{-4}	3.011×10^{-4}	41.6%
9	304	8.928×10^{-5}	6.908×10^{-5}	22.6%

Table 2: Plane Wave Refinement: Comparison of the relative L^2-error for uniform p–refinement (without direction adaptivity), and p–refinement with direction adaptivity (Algorithm 2).

(a)

(b)

(c)

(d)
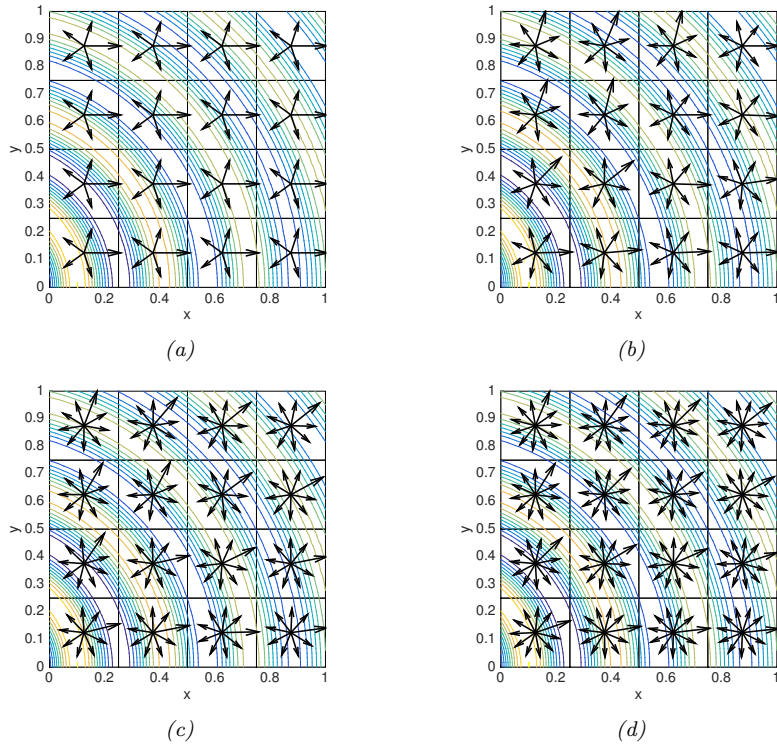
Figure 1: Plane Wave Refinement: Plane wave directions of (a) initial mesh and after (b) 1, (c) 2 and (d) 3 p–refinements with plane wave refinement (Algorithm 2)

		Relative $L^2(\Omega)$-Error		
q	No of Dofs	Initial	One Direction Adapt.	Two Direction Adapts.
3	112	2.015×10^0	8.755×10^{-1}	5.856×10^{-1}
4	144	5.027×10^{-1}	1.267×10^{-1}	1.149×10^{-1}
5	176	7.414×10^{-2}	2.614×10^{-2}	2.584×10^{-2}
6	208	1.616×10^{-2}	6.330×10^{-3}	6.327×10^{-3}
7	240	3.420×10^{-3}	1.435×10^{-3}	1.435×10^{-3}
8	272	5.154×10^{-4}	3.011×10^{-4}	3.011×10^{-4}

Table 3: Plane Wave Refinement: Comparison of the relative L^2-error with fixed effective polynomial degree, $q = 3,\ldots,8$, and direction adaptivity (Algorithm 2).

the computed relative L^2-error based on employing uniform p–refinement of the underlying TDG space $V_{\boldsymbol{p}}(\mathcal{T}_h)$ in the two cases when the standard TDG scheme is employed, i.e., when the local plane wave directions are kept fixed, and when plane wave directional adaptivity is utilized, based on exploiting Algorithm 2 (direction adaptivity). We note that, since uniform p–refinement is employed in both cases, then at each step of the refinement, both schemes possess the same number of degrees of freedom. At each step of the refinement algorithm, we observe that the exploitation of directional adaptivity leads to roughly 50% reduction in the relative L^2-error when compared to the corresponding quantity computed for the standard TDG method (without direction adaptivity). For $q = 3$, however, we note that the relative L^2-norm of the error is only reduced by a small amount; this behavior is due to the fact that the underlying computed numerical solution with $q = 2$ is too inaccurate to provide a reliable prediction of the dominant wave direction. Furthermore, we also note that, as the number of plane waves increases, the improvement in the relative L^2-error decreases; this is caused by the fact that, as the number of plane waves increases for the standard TDG scheme, one of the directions will get closer to the actual dominant direction.

In Figure 1 we plot, for each element, the initial plane wave directions and the plane wave directions computed after 1, 2, and 3 uniform p–refinements employing directional adaptivity. We emphasize the first plane wave direction with a larger arrow, i.e., the dominant wave direction as determined by Algorithm 2. Moreover, we overlay the directions on top of a contour plot showing the real part of the analytical solution (11). From Figure 1, we can clearly observe that the directional adaptivity algorithm is able to accurately determine the dominant wave direction after a few refinements.

Finally, in this section we consider performing more than one directional adaptivity step after each uniform p–refinement. To this end, we compute, for the case when $q = 3,\ldots,8$, the numerical solution with the initial directions, perform direction adaptivity using this solution, and then perform a second direction adaptation using the numerical solution computed with the directions from the first direction adaptation. We show, in Table 3, the relative L^2-norm of the error for the initial directions, as well as after the first and second application of the direction adaptivity algorithm. Here, we observe that additional application of the direction adaptivity algorithm does not lead to a significant reduction in the relative L^2-norm error; indeed, most of the reduction, when compared to the standard TDG scheme, without directional adaptivity, is attained after one step of Algorithm 2. Moreover, we emphasize that this first step may be undertaken in a very computationally cheap manner.

4.2   Efficiency of the *a posteriori* error indicator

The selection of the exponents of h_K and q_K in the weights present in (8), together with the independence on the wavenumber k, have been determined by numerical experimentation. To this end, we consider the example presented in the previous section, cf. (11), whereby the numerical approximation is computed on a series of uniform computational meshes, with uniform effective polynomial degrees q, for a range of wave numbers k. In each case, we compute the effectivity index of each constituent term arising in $\mathfrak{E}(u_h, h, \boldsymbol{p})$, whereby the dependency on h_K, q_K, and k
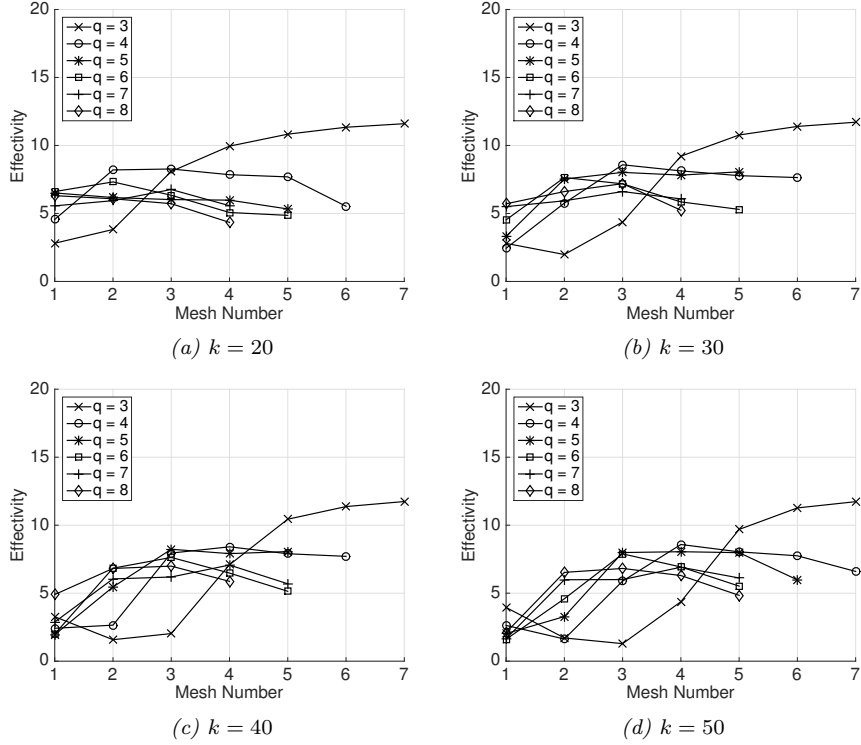
Figure 2: Effectivities for h–refinement with fixed effective polynomial degree of the smooth analytical Hankel solution with different wavenumbers.

was eliminated; note that with the removal of h_K, q_K, and k, the effectivity index of each term is computed by dividing by $\|u - u_{hp}\|_{L^2(\Omega)}$. More precisely, effectivity indices were computed for $q = 3, \ldots, 8$, and $k = 20, 30, 40, 50$, based on starting from a uniform 4×4 mesh consisting of square elements. On the basis of these results, the dependence of each term on h_K, q_K, and k was established. The final effectivity indices for the correctly scaled empirical *a posteriori* error indicator $\mathfrak{E}(u_h, h, \boldsymbol{p})$, i.e., $\mathfrak{E}(u_h, h, \boldsymbol{p})/\|u - u_{hp}\|_{L^2(\Omega)}$ are presented in Figure 2. Here, we observe that that the effectivity indices have roughly the same values for all the selected values of h, q, and k; however, at higher wave numbers, pre-asymptotic behavior leads to some increase in the effectivity indices as the mesh is refined, due to the fact that the mesh size is too large for the wavelength. We note that this behavior is more noticeable in the case when $q = 3$.

Finally, we compute the effectivity index for each individual term arising in the definition of the error indicator $\mathfrak{E}(u_h, h, \boldsymbol{p})$, cf. (8). More precisely, we define

$$\mathcal{E}_{\llbracket u_{hp} \rrbracket} := \frac{\left(\sum_{K \in \mathcal{T}_h} \left\| \alpha^{1/2} h_K^{1/2} q_F^{-1/2} \llbracket u_{hp} \rrbracket \right\|^2_{L^2(\partial K \backslash \partial \Omega)} \right)^{1/2}}{\|u - u_{hp}\|_{L^2(\Omega)}},$$

$$\mathcal{E}_{\llbracket \nabla u_{hp} \rrbracket} := \frac{\left(\sum_{K \in \mathcal{T}_h} \left\| \beta^{1/2} h_K^{3/2} q_K^{-3/2} \llbracket \nabla u_{hp} \rrbracket \right\|^2_{L^2(\partial K \backslash \partial \Omega)} \right)^{1/2}}{\|u - u_{hp}\|_{L^2(\Omega)}},$$

$$\mathcal{E}_R := \frac{\left(\sum_{K \in \mathcal{T}_h} \left\| \delta^{1/2} h_K^{3/2} q_K^{-3/2} (g_R - \nabla u_{hp} \cdot \boldsymbol{n}_F + ik u_{hp}) \right\|^2_{L^2(\partial K \cap \Gamma_R)} \right)^{1/2}}{\|u - u_{hp}\|_{L^2(\Omega)}};$$

the results for the case when $k = 20, 30, 40, 50$ are depicted in Figure 3. Here, we observe that each

13
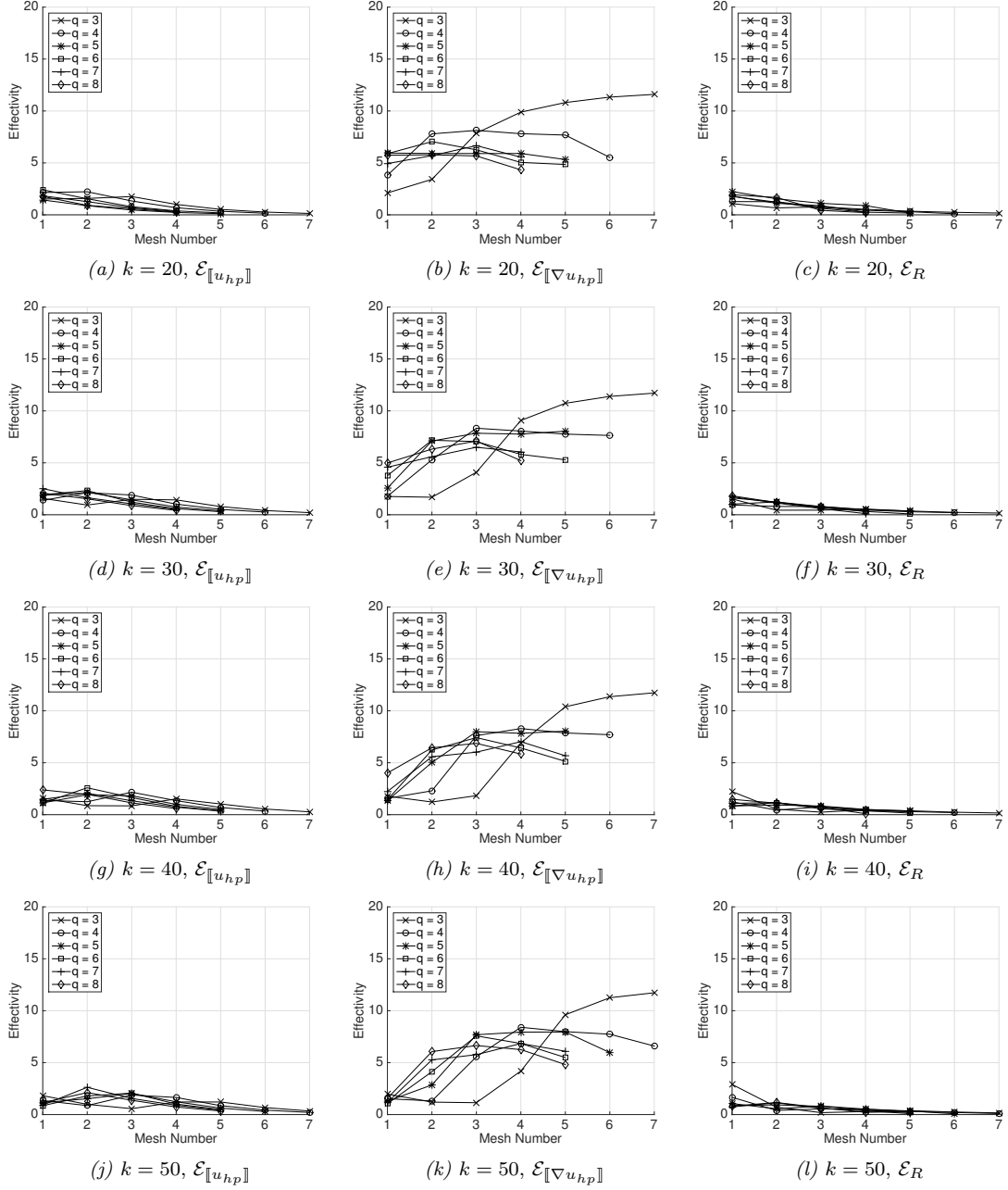
Figure 3: Effectivities of individual components of the error indicators for h–refinement with fixed effective polynomial degree of the smooth analytical Hankel solution with different wavenumbers.

14

individual effectivity index is roughly constant for all the selected values of h, q, and k, except within the pre-asymptotic region. For this smooth problem, we clearly observe that the dominant part of the error indicator involves the jump in the gradient of the numerical solution.

Remark 4.1. We note there that we have only computed the weightings for the interior and Robin faces. In the case of Dirichlet boundary conditions we assume that the weighting scales the same as the term involving $[\![u_{hp}]\!]$.

4.3 hp–Adaptive refinement

In this section we consider the computational performance of the proposed hp–adaptive refinement algorithm, with directional adaptivity, for a range of test problems in both two- and three-dimensions. To this end, employ the fixed fraction refinement strategy to mark elements for refinement; throughout this section, we set the refinement fraction equal to 25% of the elements with the largest contribution to the error bound. Furthermore, we allow the meshes $\mathcal{T}_h$ to be '1-irregular', i.e., each face of any element $K \in \mathcal{T}_h$ contains at most one hanging node (which, for simplicity, we assume to be at the barycenter of the corresponding face) and each edge of each face contains at most one hanging node (yet again assumed to be at the barycenter of the edge). We also only allow the effective polynomial degree q_K to vary by one between neighboring elements, as is commonly the case in hp-adaptivity employing standard polynomial bases. Note that this ensures that the local quasi-uniformity of the effective polynomial degree, required in the *a priori* analysis of [18], is satisfied.

For each test problem, we compare the performance of employing hp–adaptive refinement with h–adaptivity. In the latter case, we consider a standard h–adaptive algorithm, i.e., adaptive mesh refinement without directional adaptivity, as well as an h–adaptive strategy which incorporates directional adaptivity; here, we shall consider the two cases when directional adaptivity is either undertaken only on the elements marked for refinement, as well as the case when it is performed on all elements in the computational mesh. In the hp–setting, similar comparisons will be made, in addition to studying the case when directional adaptivity is only performed on elements marked for p–refinement.

We note that when hp–refinement is exploited we often reach a point where the L^2–norm of the error and *a posteriori* error bound stagnates, in the sense that both quantities no longer tend to zero, and indeed may start to oscillate, as further refinement is undertaken. This is caused by the fact that as the relative magnitude of q_K, with respect to $h_K k$, becomes large, the local plane wave bases are very ill-conditioned. In this situation, we simply stop the numerical experiments and discard further results. We refer to [10] for a numerical study of the conditioning of the plane wave basis for $d = 2$. In this case it is observed that the condition number of the underlying matrix behaves like $2.34^{p \ln p}(hk)^{1-p}$; hence, for any type of h-/p-/hp-refinement strategy, we expect to encounter issues with matrix conditioning. In the case of hp-refinement strategies, possible improvements based on ensuring q_K is well behaved with respect to $h_K k$ could be implemented; cf. [9, 22, 24] for details. We expect that the use of directional adaptivity should help with these issues, in the sense that the accuracy of the underlying numerical method can be improved by rotating the local plane-wave directions, instead of increasing q_K, or undertaking local mesh refinement.

4.3.1   Example 1 — Smooth solution (Hankel function)

In this section, we again consider the problem outlined in Section 4.1. Furthermore, we select the initial mesh to consist of 8×8 uniform square elements and set $q_K = 3$ on each $K \in \mathcal{T}_h$. Firstly, in Figures 4a and 4e we compare the relative error in the L^2-norm to the number of degrees of freedom in the TDG space $V_{\boldsymbol{p}}(\mathcal{T}_h)$, when h–refinement is employed, with wavenumbers $k = 20$ and $k = 50$, respectively. In each case, we consider the performance of the underlying adaptive algorithm when both the standard TDG scheme (without direction adaptivity) is employed, as well as the corresponding method with directional adaptivity; in this latter setting, we consider the cases when either directional adaptivity is undertaken on only the elements marked for refinement,
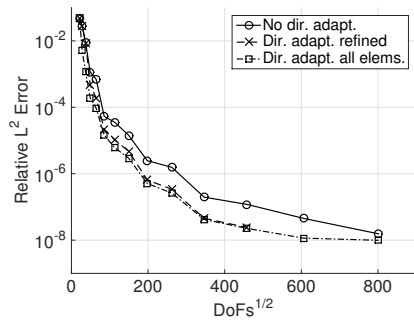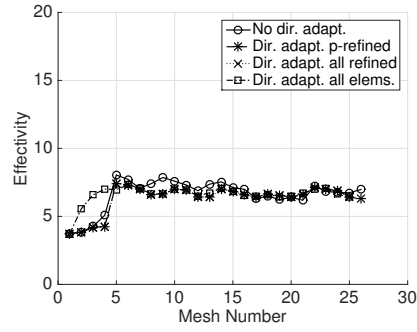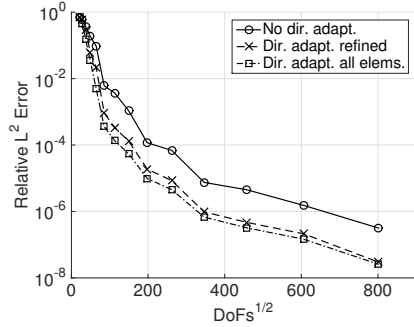
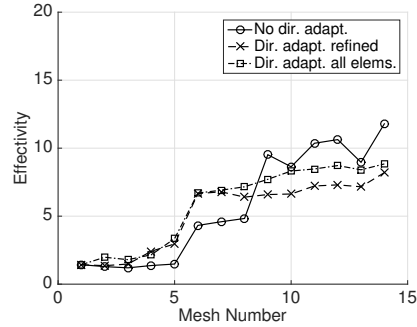*(a) $k = 20$; h–refinement*
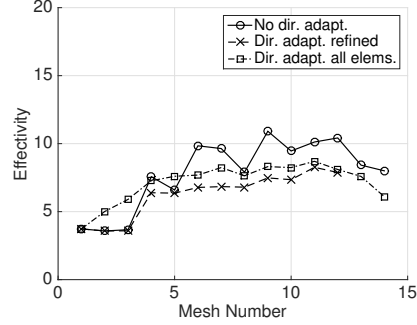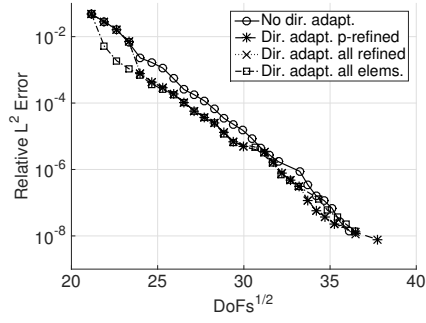
(b) $k = 20$; h–refinement
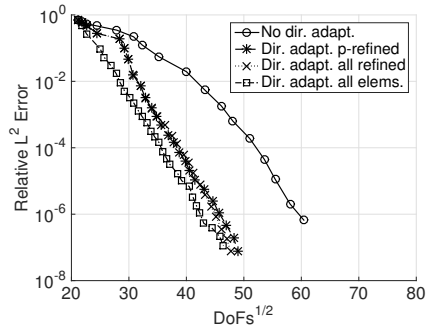

(c) $k = 20$; hp–refinement

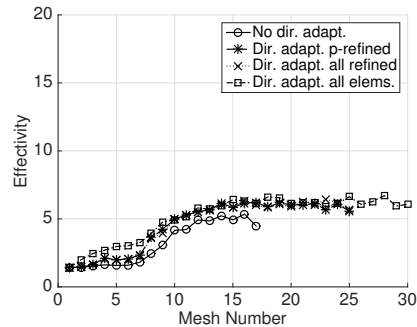*(d) $k = 20$; hp–refinement*

(e) $k = 50$; h–refinement

(f) $k = 50$; h–refinement

(g) $k = 50$; hp–refinement

(h) $k = 50$; hp–refinement

Figure 4: Example 1: (a) L^2-error and (b) Effectivity index for h–refinement with wavenumber $k = 20$; (c) L^2-e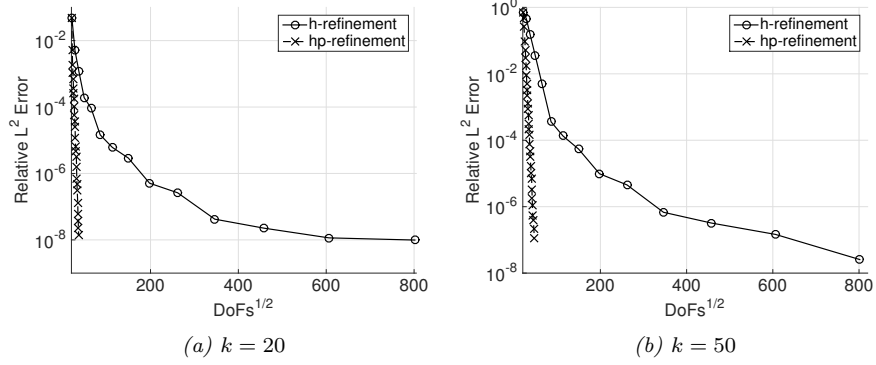rror and (d) Effectivity index for hp–refinement with $k = 20$; (e) L^2-error and (f) Effectivity index for h–refinement with $k = 50$; (g) L^2-error and (h) Effectivity index for hp–refinement with $k = 50$.

(a) k = 20

(b) k = 50

Figure 5: Example 1: Comparison of relative L^2-error for h– and hp–refinement, with direction adaptivity on all elements, for wavenumbers (a) $k = 20$ and (b) $k = 50$.

(a) k = 20

(b) k = 20

(c) k = 50

(d) k = 50
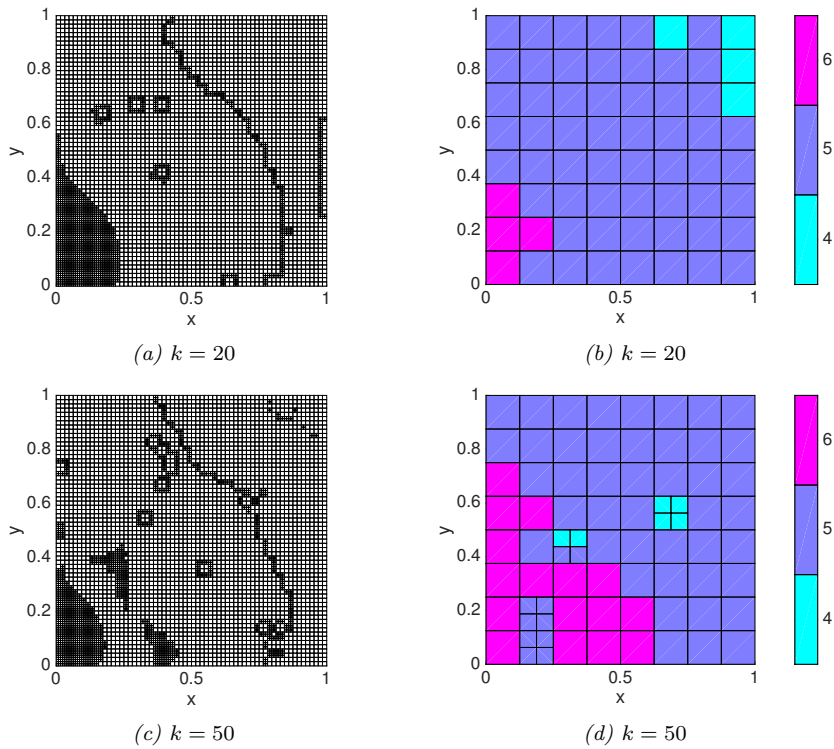
Figure 6: Example 1: Meshes after 8 (a) h– and (b) hp–refinements for wavenumber $k = 20$; meshes after 8 (c) h– and (d) hp–refinements for wavenumber $k = 50$.

as well as when it is exploited on every element in the computational mesh $\mathcal{T}_h$. Analogous results are presented in Figures 4c and 4g in the hp–setting, respectively; here, we compare standard hp–refinement, with hp–adaptivity incorporating directional adaptivity. In the latter case, different directional adaptivity strategies are considered: firstly, directional adaptivity is performed only on elements marked for p–refinement; secondly, directional adaptivity is undertaken on all elements marked for refinement; finally, directional adaptivity is applied to every element in $\mathcal{T}_h$. In the hp–setting we observe exponential convergence of the error as the finite element space is adaptively enriched: on a linear-log scale, the convergence lines are roughly straight. Thereby, it is clear that the exploitation of the proposed hp–refinement algorithm, with directional adaptivity, leads to a significant reduction in the relative L^2-norm of the error, for a given number of degrees of freedom, when compared to the same quantity computed with h–refinement alone; cf. Figure 5.

In both the h– and hp–refinement cases, we generally observe that the error is decreased when directional refinement is employed. Moreover, it is evident in the hp–setting that selecting more elements for directional refinement generally leads to a smaller error, for a given number of degrees of freedom; this is particularly noticeable in the case when $k = 50$. In Figures 4b, 4d, 4f, and 4h we plot the effectivity indices for each of the above refinement strategies for the case when $k = 20, 50$; here, we observe that they remain roughly constant during adaptive h–/hp–mesh refinement, and are roughly the same for the two different wavenumbers, with the notable exception of the pre-asymptotic region for $k = 50$.

Finally, in Figures 6a–6d, we show the meshes after 8 h– and hp–refinements, with directional adaptivity employed on all elements, for both $k = 20$ and $k = 50$; here, the hp–meshes show the effective polynomial degree q_K for each element. Given the smoothness of the analytical solution on Ω, we observe that the resulting computational meshes are almost uniform; indeed, in the hp–setting almost uniform p–refinement has been undertaken.

4.3.2   Example 2 — Singular solution

In this second example, we consider problem (1) posed on the L-shaped domain $\Omega = (-1, 1)^2 \setminus (0, 1) \times (-1, 0)$, $\Gamma_R = \partial\Omega$, and $\Gamma_D \equiv \emptyset$, with Robin boundary condition g_R selected so that the analytical solution is given, in polar coordinates (r, θ), by

$$u(r, \theta) = \mathcal{J}_{2/3}(kr) \sin(2\theta/3),$$

where $\mathcal{J}_{2/3}$ denotes the Bessel function of the first kind with order $2/3$; we note that the gradient of u has a singularity at the origin.

As in the previous example, we again compare the performance of the h– and hp–adaptive refinement algorithms, both in the standard setting, as well as when directional adaptivity is employed; here, we again consider the analogous directional refinement strategies employed in Section 4.3.1. To this end, in Figures 7a and 7e we compare the relative error in the L^2-norm with the number of degrees of freedom in the TDG space $V_{\boldsymbol{p}}(\mathcal{T}_h)$ when h–refinement is employed for $k = 20$ and $k = 50$, respectively; the respective convergence plots in the hp–setting are given in Figures 7c and 7g. Here, we observe that although exploiting hp–refinement leads to exponential convergence of the relative L^2-norm of the error as $V_{\boldsymbol{p}}(\mathcal{T}_h)$ is enriched, in both the h– and hp–settings, we observe that the magnitude of the error, computed both with and without directional refinement, is roughly identical; i.e., directional refinement does not lead to any reduction in the computed TDG solution when either h–/hp–refinement is employed. We note that, for this particular problem, this behavior is expected, since the error in the computed TDG solution is dominated by the influence of the singularity at the origin, rather than local wave propagation.

In Figures 7b, 7d, 7f, and 7h we plot the effectivity indices when both h– and hp–refinement is employed for the case when $k = 20, 50$. In all cases, we observe that the effectivity indices are roughly constant for this singular problem, though when h–refinement is employed, on highly refined meshes, we see a slight drop in the computed effectivity indices. Finally, in Figures 8a–8d, we show the meshes after 8 h– and hp–refinements, with direction adaptivity employed on all elements, for both $k = 20$ and $k = 50$. As we would expect, in both the h– and hp–settings, mesh subdivision is concentrated in the vicinity of the singularity located at the origin; away
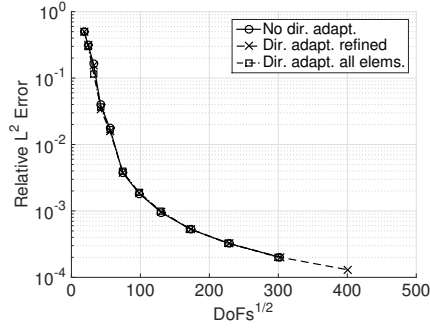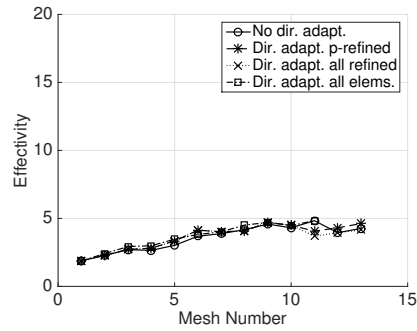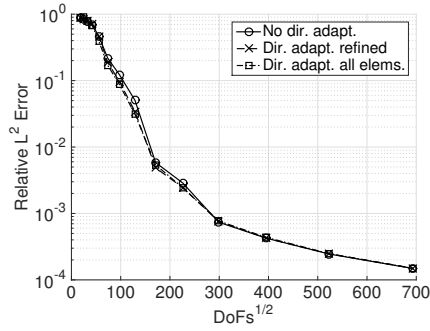
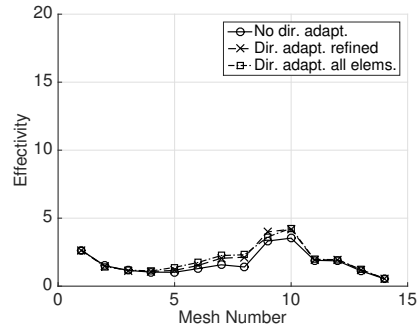*(a) $k = 20$; h–refinement*
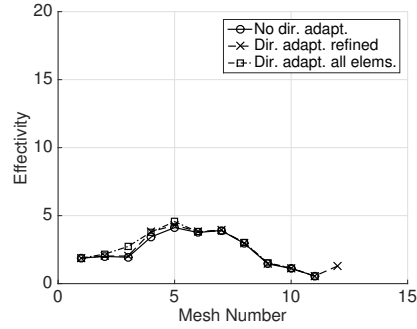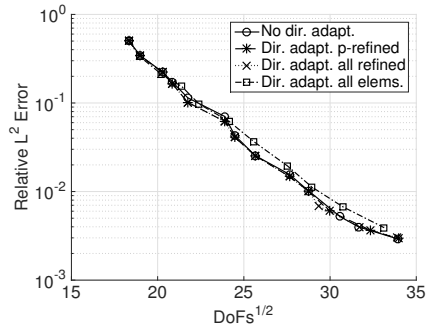
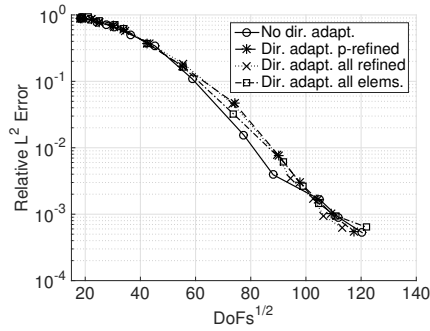*(b) $k = 20$; h–refinement*

(c) $k = 20$; hp–refinement

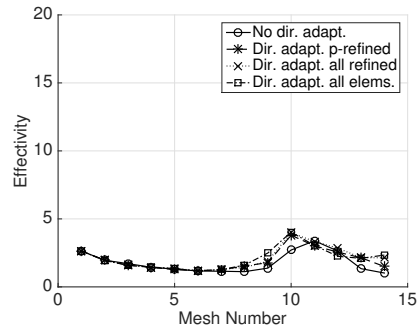*(d) $k = 20$; hp–refinement*

(e) $k = 50$; h–refinement

(f) $k = 50$; h–refinement

(g) $k = 50$; hp–refinement

(h) $k = 50$; hp–refinement

Figure 7: Example 2: (a) L^2-error and (b) Effectivity index for h–refinement with wavenumber $k = 20$; (c) L^2-error and (d) Effectivity index for hp–refinement with $k = 20$; (e) L^2-error and (f) Effectivity index for h–refinement with $k = 50$; (g) L^2-error and (h) Effectivity index for hp–refinement with $k = 50$.

19

(a) k = 20

(b) k = 20

(c) k = 50

(d) k = 50

Figure 8: Example 2: Meshes after 8 (a) h– and (b) hp–refinements for wavenumber $k = 20$; meshes after 8 (c) h– and (d) hp–refinements for wavenumber $k = 50$.

from this region, the h–refinement algorithm employs almost uniform mesh subdivision, while the hp–refinement strategy employs the necessary combination of local mesh refinement and local polynomial enrichment, as required, to reduce the error in the computed TDG solution.

4.3.3 Example 3 — Transmission/internal reflection

We now consider the case of transmission and internal reflection of a plane wave across a fluid-fluid interface in the domain $\Omega = (-1, 1)^2$, $\Gamma_R \equiv \emptyset$, and $\Gamma_D = \partial\Omega$, with two different refractive indices, cf. [23, Section 6.3]. The interface between the two regions is located at $y = 0$; in this setting the wavenumber k is given by the piecewise constant function

$$k(x,y) = \begin{cases} k_1 := \omega n_1 & \text{if } y \leq 0, \\ k_2 := \omega n_2 & \text{if } y > 0, \end{cases}$$

where, we select $\omega = 11$, $n_1 = 2$, and $n_2 = 1$. Throughout this section we impose an appropriate inhomogeneous Dirichlet boundary condition, so that the analytical solution u to (1) is given, for a constant $0 \leq \theta_i \leq \pi/2$, by

$$u(x,y) = \begin{cases} Te^{i(K_1 x + K_2 y)} & \text{if } y > 0, \\ e^{ik_1(x \cos(\theta_i) + y \sin(\theta_i))} + Re^{ik_1(x \cos(\theta_i) - y \sin(\theta_i))} & \text{if } y < 0, \end{cases}$$

where $K_1 = k_1 \cos(\theta_i)$, $K_2 = \sqrt{k_2^2 - k_1^2 \cos^2(\theta_i)}$,

$$R = -\frac{K_2 - k_1 \sin(\theta_i)}{K_2 + k_1 \sin(\theta_i)},$$

and $T = 1 + R$. We note that there exists a critical angle θ_{crit}, such that when $\theta_i > \theta_{crit}$ the wave is refracted, while $\theta_i < \theta_{crit}$ results in internal reflection, cf. [23, Section 6.3]. As in [23] we

20
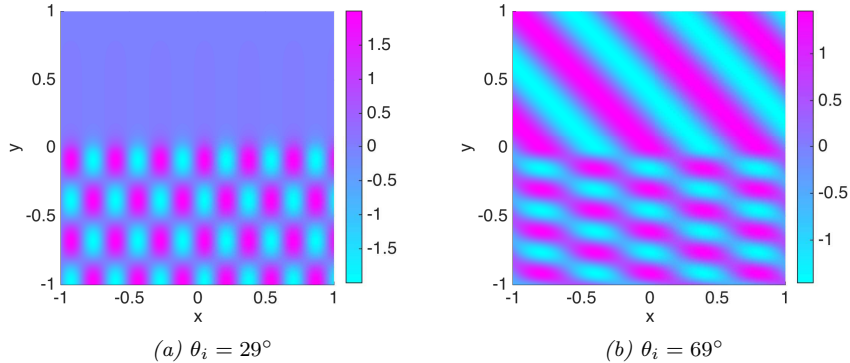
Figure 9: Example 3: Analytical solutions (real part) when (a) $\theta_i = 29°$ resulting in internal reflection, and (b) $\theta_i = 69°$ resulting in refraction.

perform numerical experiments for both internal reflection ($\theta_i = 29°$) and refraction ($\theta_i = 69°$). To highlight the reflection and refraction behavior, in Figures 9a and 9b we show the analytical solution when $\theta_i = 29°$ and $\theta_i = 69°$, respectively.

To account for the jump in the wavenumber k, the value of k present in the integrals along the interface $y = 0$ in the TDG scheme (2) is replaced by ω. We select the initial mesh to consist of 8×8 uniform square elements, so that the interface between the two materials is captured by the mesh; thereby, the wavenumber is constant in every element, and hence the TDG space (4) and error indicators (8) can be easily modified to treat this example by setting the wavenumber for each element equal to the wavenumber of the material within which the element is contained. Firstly, we consider the case when there is an internal reflection, i.e., when $\theta_i = 29°$; to this end, in Figures 10a and 10c we plot the relative error in the L^2-norm against the number of degrees of freedom in $V_{\boldsymbol{p}}(\mathcal{T}_h)$ using both h– and hp–refinement, respectively. As for the previous numerical experiments, here we again observe exponential convergence of the error when hp–refinement is employed. Furthermore, in the h–version setting, we observe that employing directional adaptivity does not improve the magnitude of the error; when hp–refinement is exploited, initially the standard refinement approach is superior, though as $V_{\boldsymbol{p}}(\mathcal{T}_h)$ is enriched, we again observe the benefits of employing directional adaptivity. This behavior is perhaps expected, since for the internal reflection case, no waves are present above the $y = 0$ line and moreover it does not possess a dominant wave propagation direction below the $y = 0$ line due to the reflected waves, cf. Figure 9a. In Figures 10b and 10d, we plot the effectivity indices for both refinement strategies, respectively; here we observe that, apart from an initial pre-asymptotic region, the effectivity indices are roughly constant.

The corresponding convergence plots for the refraction case, i.e., when $\theta_i = 69°$, are presented in Figures 10e and 10g when both h– and hp–refinement are employed, respectively; in the latter setting, we again observe exponential convergence of the computed relative L^2-norm of the error. Moreover, in contrast to the case when there is an internal reflection, here we observe the computational benefits of employing directional adaptivity, in the sense that this typically leads to a reduction in the error, for a given fixed number of degrees of freedom, when compared to the standard refinement strategy; this is particularly evidenced in the hp–setting. Indeed, in this case there is a dominant propagation direction throughout the domain, cf. Figure 9b. Figures 10f and 10h show the effectivity indices computed using both h– and hp–refinement, respectively; analogous behavior is observed as for the internal reflection case, i.e., the effectivity indices become roughly constant, after an initial pre-asymptotic region.

Finally, in Figures 11a & 11b we show the meshes after 7 h– and hp– adaptive mesh refinements have been performed, respectively, in the case of an internal reflection, i.e., $\theta_i = 29°$. Here, the h–refinement strategy concentrates most of the elements in the $y < 0$ region; although, there is some refinement above $y = 0$ to resolve the exponentially decaying solutions present there. Additional
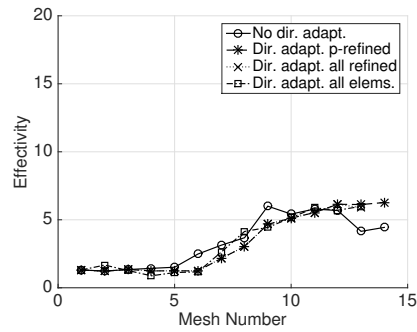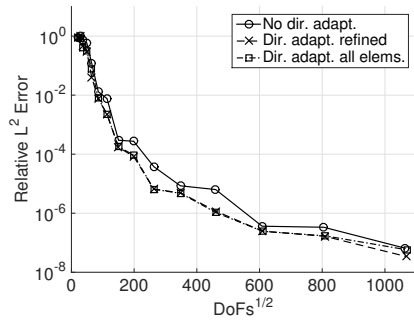
21
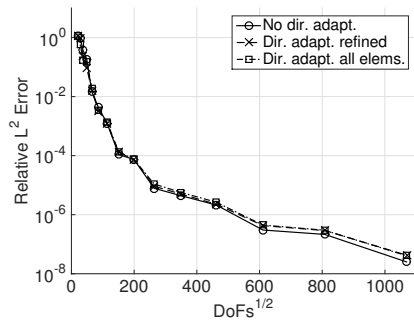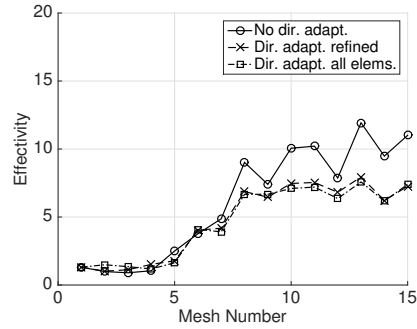
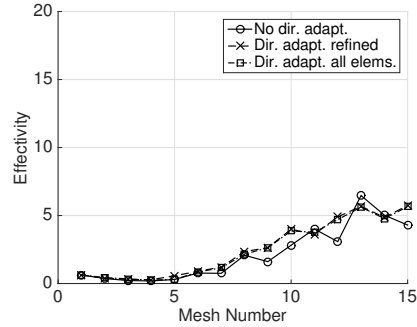*(a) $\theta_i = 29°$; h–refinement* *(b) $\theta_i = 29°$; h–refinement*
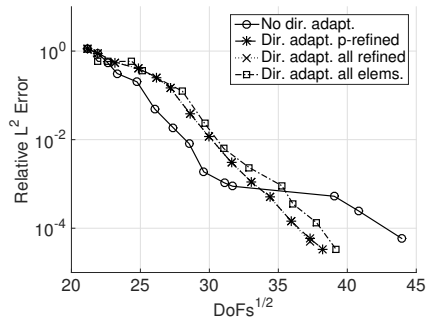
(c) $\theta_i = 29°$; hp–refinement *(d) $\theta_i = 29°$; hp–refinement*
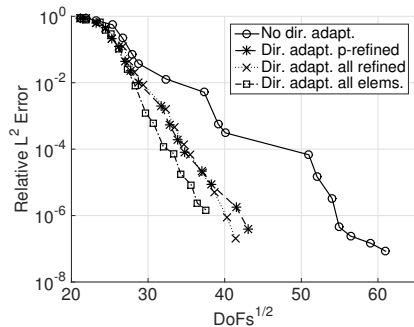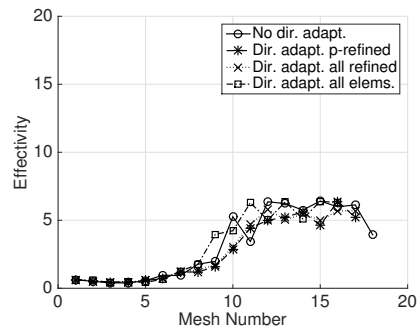
(e) $\theta_i = 69°$; h–refinement *(f) $\theta_i = 69°$; h–refinement*

(g) $\theta_i = 69°$; hp–refinement *(h) $\theta_i = 69°$; hp–refinement*

Figure 10: Example 3: (a) L^2-error and (b) Effectivity index for h–refinement with reflection ($\theta_i = 29°$); (c) L^2-error and (d) Effectivity index for hp–refinement with reflection ($\theta_i = 29°$); (e) L^2-error and (f) Effectivity index for h–refinement with refraction ($\theta_i = 69°$); (g) L^2-error and (h) Effectivity index for hp–refinement with refraction ($\theta_i = 69°$).

(a) $\theta_i = 29°$ *(b) $\theta_i = 29°$*

(c) $\theta_i = 69°$ *(d) $\theta_i = 69°$*
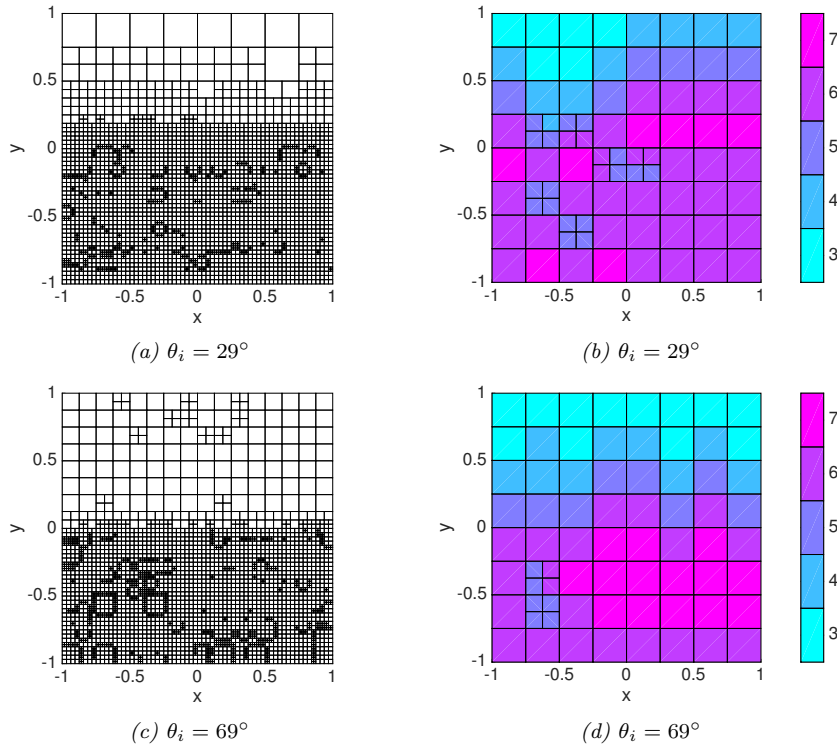
Figure 11: Example 3: Meshes after 7 (a) h– and (b) hp–refinements for reflection ($\theta_i = 29°$); meshes after 7 (c) h– and (d) hp–refinements for refraction ($\theta_i = 69°$).

mesh smoothing has also been undertaken here to ensure that there is only one hanging node per face, cf. [23]. The hp–refinement algorithm also performs some h–refinement below the $y = 0$ line, though this region is largely p–refined; however, most of the refinement occurs around the $y = 0$ line to resolve the exponentially decaying solutions. Some p–refinement occurs in the rest of the $y > 0$ region, which is caused by enforcing the condition that the effective polynomial degree may only vary by one between neighboring elements. In the refraction case, i.e., $\theta_i = 69°$, cf. Figures 11c & 11d, we note a sharp boundary at the $y = 0$, with more refinement undertaken in the $y < 0$ region than the region $y > 0$.

4.3.4 Example 4 — Scattering

We now consider sound-soft scattering around an obstacle. To this end, we consider a concave quadrilateral (kite) obstacle Ω_D defined by the vertices $(\pi + 1, \pi)^\top$, $(\pi, \pi + 1)^\top$, $(\pi - 1, \pi)^\top$, and $(\pi, \pi + 0.25)^\top$ within the domain $\Omega_R = (0, 2\pi)^2$; i.e., $\Omega = \Omega_R \setminus \Omega_D$. We define homogeneous Dirichlet boundary conditions $g_D = 0$ on the boundary of the obstacle $\Gamma_D = \partial\Omega_D$ and Robin boundary conditions

$$g_R(x) = \mathrm{e}^{ik\boldsymbol{d}\cdot\boldsymbol{x}},$$

with $k = 20$ and $\boldsymbol{d} = -(\cos(^{6\pi}/_{13}), \sin(^{6\pi}/_{13}))^\top$ on the remainder of the boundary $\Gamma_R = \partial\Omega_R$. Thereby, this simulates an incident plane wave with wavenumber k traveling along the direction $(\cos(^{6\pi}/_{13}), \sin(^{6\pi}/_{13}))^\top$.

We again compare the performance of the h– and hp–adaptive refinement algorithms, both in the standard setting, as well as when directional adaptivity is employed; here, we consider the analogous directional refinement strategies employed in Section 4.3.1. We note that for this problem we do not have a known analytical solution; therefore, in Figures 12a and 12b we compare the error estimator $\mathfrak{E}(u_h, h, \boldsymbol{p})$ from (8) with the number of degrees of freedom in the TDG space
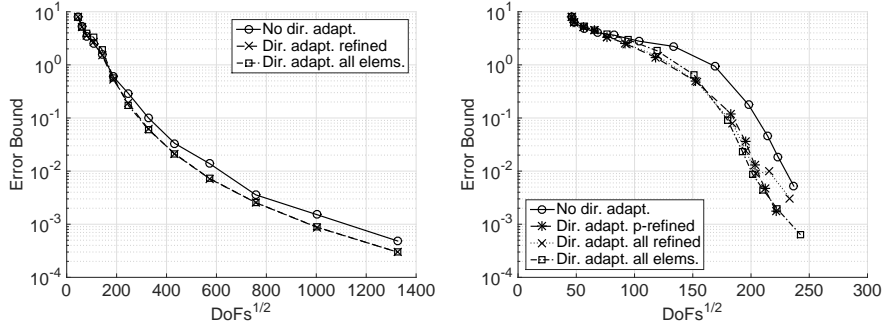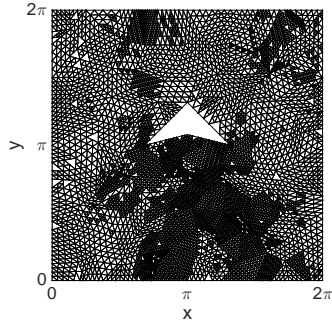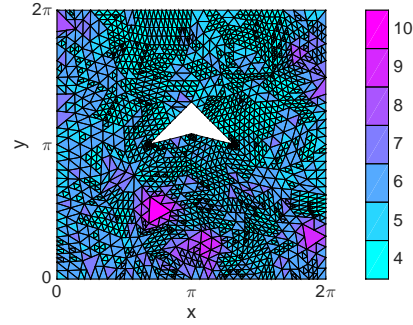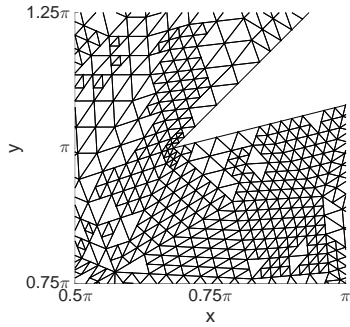
(a) h–refinement
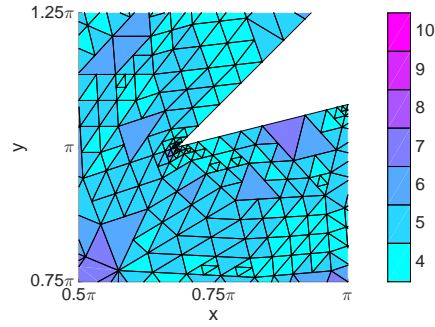
(b) hp–refinement
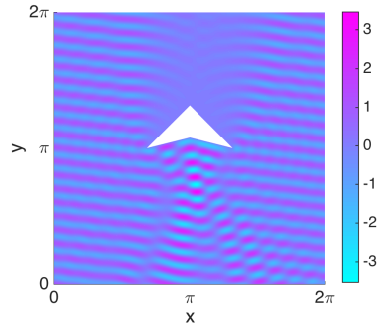
(c) Numerical Solution

(d) h-refinement Mesh

(e) hp-refinement Mesh

(f) h-refinement Mesh; zoomed

(g) hp-refinement Mesh; zoomed

Figure 12: Example 4: *A posteriori* error bound for (a) *h*–refinement with wavenumber and (b) *hp*–refinement with $k = 20$; (c) $\text{Re}(u_{hp})$ solution after 9-hp refinements; Meshes after (d) 6 *h*– and (e) 9 *hp*–refinements; Mesh detail around the corner at $(\pi - 1, \pi)$ after (f) 6 *h*– and (g) 9 *hp*–refinements.

$V_{\boldsymbol{p}}(\mathcal{T}_h)$ when h–refinement and hp-refinement, respectively, are employed. In the h-refinement case we observe that direction adaptation only slightly improves the error, which would be expected, since in the region directly below the obstacle, the reflected waves will result in multiple dominant wave directions. In the hp-refinement case we actually observe an improvement when directional adaptivity is exploited; however, we note that this appears to be due to the fact that initially the method does not converge until the mesh is fine enough to resolve the incidence waves and that the refinement strategies with directional adaptivity appears to resolve these waves with fewer degrees of freedom.

In Figures 12d & 12e we show the meshes after 6 h– and 9 hp–adaptive mesh refinements have been performed, respectively. We note that although the refinement is fairly uniform in the majority of the domain, there is more refinement in the area below the kite where the reflected waves are present; cf., Figure 12c, which shows the real part of the numerical solution after 9 hp-refinements. We also note that, in both h– and hp–settings, mesh refinement is undertaken in the vicinity of the two acute corners of the kite located at $(\pi - 1, \pi)^{\top}$ and $(\pi + 1, \pi)^{\top}$; cf., Figures 12f & 12g for an enlargement of the meshes around the point $(\pi - 1, \pi)^{\top}$.

4.3.5   Example 5 — 3D smooth solution (plane wave)

In this final example, we consider problem (1) posed on the domain $\Omega = (0, 1)^3$, $\Gamma_R = \partial\Omega$, and $\Gamma_D \equiv \emptyset$, with Robin boundary condition g_R selected so that the analytical solution u to (1) is given by

$$u(\boldsymbol{x}) = \mathrm{e}^{ik\boldsymbol{d}\cdot\boldsymbol{x}},$$

where $\boldsymbol{d}_j = 1/\sqrt{3}$ for $j = 1, 2, 3$.

In Figures 13a and 13e we present the performance of the proposed directional adaptivity algorithm employing h–refinement with wavenumbers $k = 20$ and $k = 50$, respectively; the analogous results for hp–refinement are given in Figures 13c and 13g, respectively. As in the two–dimensional setting, we observe that selecting more elements for directional adaptivity at each step of the proposed refinement strategy, leads to a greater reduction in the relative L^2-norm of the error, for a fixed number of degrees of freedom, when compared to the standard case when directional adaptivity is not employed. Of course, given the simple nature of the analytical solution for this problem, we clearly expect directional adaptivity to be advantageous. In the case when the wavenumber $k = 50$ we note that both h– and hp–refinement strategies are essentially in the pre-asymptotic region; however, performing directional adaptivity ensures that the method leaves this pre-asymptotic region after only a few mesh refinements. Finally, in Figures 13b, 13d, 13f, and 13h we plot the effectivity indices of both the h– and hp–refinement algorithms for the case when $k = 20, 50$. We note, especially in the hp–refinement case, that the effectivity indices are roughly constant but do slightly rise after the pre-asymptotic region.

5   Concluding remarks

In this article we have developed an automatic hp–adaptive refinement algorithm for the TDG approximation of the homogeneous Helmholtz equation. In addition to employing both local mesh subdivision and local basis enrichment, we also locally rotate the underlying plane wave basis in such a manner so that the first basis function is aligned with the dominant wave direction. The choice to h– or p–refine an element is based on a prediction of how much reduction we expect to observe in the elementwise error indicator, when a particular refinement is performed. The alignment of the local basis with the dominant wave direction is undertaken on the basis of an eigenvalue analysis of the Hessian of the numerical solution, together with a correction computed from an impedance condition. The computational efficiency of the proposed adaptive strategy has been studied through a series of numerical examples; indeed, the application of hp–refinement, with directional adaptivity, leads to a significant reduction in the computed error compared to standard refinement strategies. We also note that performing directional adaptivity on all elements generally leads to a greater reduction in the error than the corresponding case when only elements
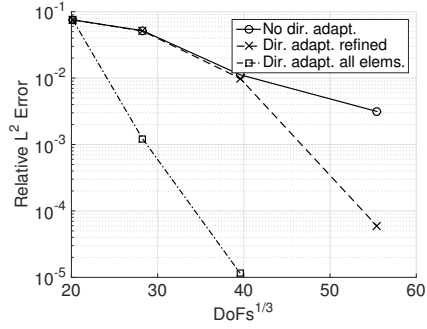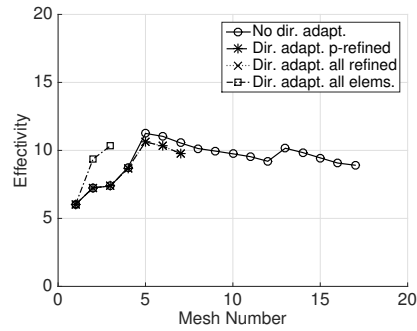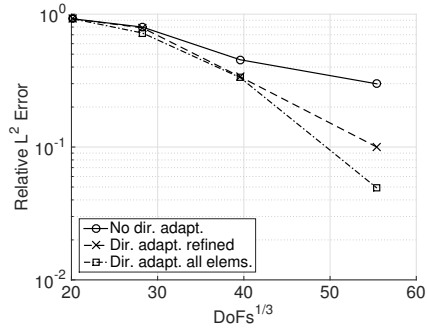
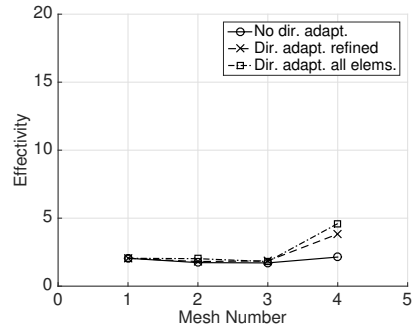*(a) $k = 20$; h–refinement*
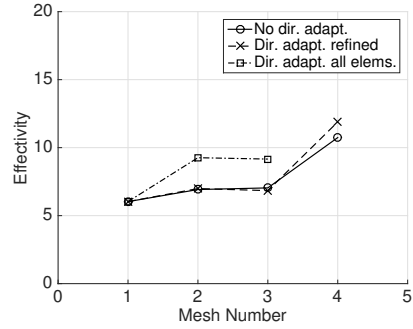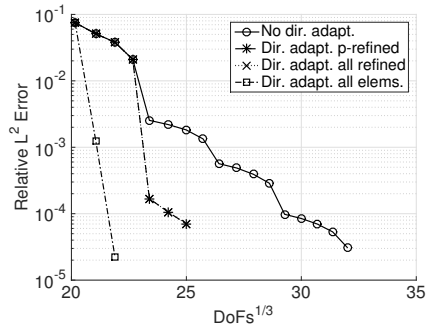
(b) $k = 20$; h–refinement
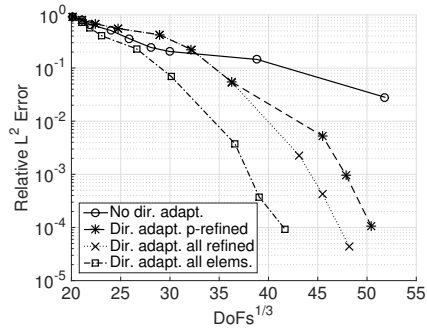

(c) $k = 20$; hp–refinement

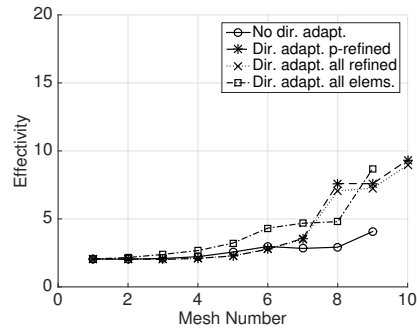*(d) $k = 20$; hp–refinement*

(e) $k = 50$; h–refinement

(f) $k = 50$; h–refinement

(g) $k = 50$; hp–refinement

(h) $k = 50$; hp–refinement

Figure 13: Example 5: (a) L^2-error and (b) Effectivity index for h–refinement with wavenumber $k = 20$; (c) L^2-error and (d) Effectivity index for hp–refinement with $k = 20$; (e) L^2-error and (f) Effectivity index for h–refinement with $k = 50$; (g) L^2-error and (h) Effectivity index for hp–refinement with $k = 50$.

marked for refinement are directionally adapted; clearly, this error reduction is attained while keeping the number of degrees of freedom in the underlying TDG space fixed. In this article, the proposed Hessian-based directional adaptivity algorithm can only take into account a single dominant direction; in general cases, for example, in scattering problems where several principle wave directions may be present, this strategy may not be sufficient to improve the accuracy in the underlying numerical solution. As part of our programme of future research, we will consider the extension to this setting; furthermore, work will also be devoted to the derivation of robust hp–version *a posteriori* error bounds, as well as the application to problems of engineering interest.

Acknowledgement

References

[1] A. Agrawal and R. H. W. Hoppe. Optimization of plane wave directions in plane wave discontinuous Galerkin methods for the Helmholtz equation. *Port. Math.*, 74:69–89, 2017.

[2] M. Amara, S. Chaudhry, J. Diaz, R. Djellouli, and S. L. Fiedler. A local wave tracking strategy for efficiently solving mid- and high-frequency Helmholtz problems. *Comput. Methods Appl. Mech. and Engrg.*, 276:473–508, 2014.

[3] M. Amara, R. Djellouli, and C. Farhat. Convergence analysis of a discontinuous Galerkin method with plane waves and Lagrange multipliers for the solution of Helmholtz problems. *SIAM J. Numer. Anal.*, 47(2):1038–1066, 2009.

[4] F. Babuška, F. Ihlenburg, T. Strouboulis, and S. K. Gangaraj. A posteriori error estimation for finite element solutions of Helmholtz' equation I. The quality of local indicators and estimators. *Internat. J. Numer. Methods Engrg.*, 40(18):3443–3462, 1997.

[5] F. Babuška, F. Ihlenburg, T. Strouboulis, and S. K. Gangaraj. A posteriori error estimation for finite element solutions of Helmholtz' equation II. Estimation of the pollution error. *Internat. J. Numer. Methods Engrg.*, 40(21):3883–3900, 1997.

[6] T. Betcke and J. Phillips. Adaptive plane wave discontinuous Galerkin methods for Helmholtz problems. In *Proceedings of the 10th International Conference on the Mathematical and Numerical Aspects of Waves*, pages 261–264, 2011.

[7] T. Betcke and J. Phillips. Approximation by dominant wave directions in plane wave methods. Technical report, UCL, 2012. Available at `http://discovery.ucl.ac.uk/1342769/`.

[8] D. Braess. *Finite elements. Theory, fast solvers, and applications in solid mechanics.* Cambridge University Press, Cambridge, second edition, 2001.

[9] O. Cessenat and B. Després. Application of an Ultra Weak Variational Formulation of elliptic PDEs to the two-dimensional Helmholtz problem. *SIAM J. Numer. Anal.*, 35(1):255–299, 1998.

[10] S. Congreve, J. Gedicke, and I. Perugia. Numerical investigation of the conditioning for plane wave discontinuous Galerkin methods. In *Numerical Mathematics and Advanced Applications. Proceedings of ENUMATH 2017, the 12th European Conference on Numerical Mathematics and Advanced Applications*, Voss, Norway, 2018. Springer.

[11] W. Dörfler and S. Sauter. A posteriori error estimation for highly indefinite Helmholtz problems. *Comp. Meth. Appl. Math.*, 13(3):333–347, 2013.

[12] L. Formaggia and S. Perotto. New anisotropic a priori error estimates. *Numer. Math.*, 89(4):641–667, 2001.

[13] L. Formaggia and S. Perotto. Anisotropic error estimates for elliptic problems. *Numer. Math.*, 94(1):67–92, 2003.

[14] E. Georgoulis, E. Hall, and P. Houston. Discontinuous Galerkin methods for advection–diffusion–reaction problems on anisotropically refined meshes. *SIAM J. Sci. Comput.*, 30(1):246–271, 2007.

[15] C. J. Gittelson. Plane wave discontinuous Galerkin methods. Master's thesis, ETH Zurich, 2008. `http://www.sam.math.ethz.ch/~hiptmair/StudentProjects/Gittelson/thesis.pdf`.

[16] C. J. Gittelson, R. Hiptmair, and I. Perugia. Plane wave discontinuous Galerkin methods: Analysis of the *h*-version. *ESAIM Math. Model. Numer. Anal.*, 43(2):297–331, 2009.

[17] E. J. C. Hall. *Anisotropic Adaptive Refinement For Discontinuous Galerkin Methods*. PhD thesis, Department of Mathematics, University of Leicester, 2007.

[18] R. Hiptmair, A. Moiola, and I. Perugia. Trefftz discontinuous Galerkin methods for acoustic scattering on locally refined meshes. *Appl. Numer. Math.*, 79:79–91, 2014.

[19] R. Hiptmair, A. Moiola, and I. Perugia. A survey of Trefftz methods for the Helmholtz equation. In G. R. Barrenechea, F. Brezzi, A. Cangiani, and E. H. Georgoulis, editors, *Building Bridges: Connections and Challenges in Modern Approaches to Numerical Partial Differential Equations*, pages 237–279. Springer, Cham, 2016.

[20] P. Houston. AptoFEM finite element analysis software, 2017. `http://www.aptofem.com` [Online].

[21] 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.

[22] T. Huttunen, P. Monk, and J. P. Kaipio. Computational aspects of the ultra-weak variational formulation. *J. Comput. Phys.*, 182(1):27–46, 2002.

[23] S. Kapita, P. Monk, and T. Warburton. Residual-based adaptivity and PWDG methods for the Helmholtz equation. *SIAM J. Sci. Comput.*, 37(3), 2015.

[24] T. Luostari, T. Huttunen, and P. Monk. Improvements for the ultra weak variational formulation. *Internat. J. Numer. Methods Engrg.*, 94(6):598–624, 2013.

[25] J. M. Melenk and B. I. Wohlmuth. On residual-based a posteriori error estimation in *hp*-FEM. *Adv. Comp. Math.*, 15(1–4):311–331, 2001.

[26] W. F. Mitchell and M. A. McClain. A comparison of *hp*-adaptive strategies for elliptic partial differential equations. Technical Report NISTIR 7824, National Institute of Standards and Technology, 2011.

[27] W. F. Mitchell and M. A. McClain. A comparison of hp-adaptive strategies for elliptic partial differential equations. *ACM Trans. Math. Softw.*, 41(1):2:1–2:39, 2014.

[28] A. Moiola, R. Hiptmair, and I. Perugia. Plane wave approximation of homogeneous Helmholtz solutions. *Z. Angew. Math. Phys.*, 62(5):809–837, 2011.

[29] S. Sauter and J. Zech. A posteriori error estimation of *hp*-dG finite element methods for highly indefinite Helmholtz problems. *SIAM J. Numer. Anal.*, 53:2414–2440, 2015.

[30] I. H. Sloan and R. S. Womersley. Extremal systems of points and numerical integration on the sphere. *Adv. Comput. Math.*, 21(1):107–125, 2004.

[31] R. S. Womersley. Extremal (maximum determinant) points on the sphere S^2, 2007. `http://web.maths.unsw.edu.au/~rsw/Sphere/Extremal/New/index.html` [Online].

[32] J. Zech. A posteriori error estimation of hp-DG finite element methods for highly indefinite Helmholtz problems. Master's thesis, Universität Zürich, 2014. `http://www.math.uzh.ch/compmath/index.php?id=dipl`.