ANISOTROPIC GOAL-ORIENTED MESH ADAPTATION IN FIREDRAKE

Joseph G. Wallwork[1] Nicolas Barral[2] David A. Ham[3] Matthew D. Piggott[4]

[1]*Imperial College London, London, U.K. j.wallwork16@imperial.ac.uk*
[2]*Imperial College London, London, U.K. n.barral@imperial.ac.uk*
[3]*Imperial College London, London, U.K. david.ham@imperial.ac.uk*
[4]*Imperial College London, London, U.K. m.d.piggott@imperial.ac.uk*

ABSTRACT

We consider metric-based mesh adaptation methods for steady-state partial differential equations (PDEs), solved using the finite element method in *Firedrake*. In this work, a number of mesh-adaptive methods are implemented within this framework, each enabling accurate approximation of a scalar *quantity of interest (QoI)*. Through the QoI we define adjoint equations, with which we may gain understanding of its sensitivities to aspects of the PDE solution. *Dual weighted residual* type error estimation techniques are utilised in order to enable a *goal-oriented* strategy. Isotropic and anisotropic approaches are considered, both of which are able to achieve the same relative error in approximating the QoI as with uniform refinement, but using fewer elements. For validation purposes, we compare QoI values resulting from these approaches against analytical values which may be extracted for a particular advection-diffusion based test case. Potential applications in desalination plant outfall modelling are discussed.

Keywords: anisotropy, mesh adaptation, error estimation, dual weighted residual, Firedrake

1. INTRODUCTION

In computational fluid dynamics (CFD), it is often the case that we are required to do more than just approximate the solution of a PDE. For many problems, what is most important is the accurate approximation of some functional, relating to a *quantity of interest (QoI)*. For example, the QoI could be the drag on an aeroplane wing [1], the profit or power output of a tidal turbine array [2], or the ocean surface elevation near to important coastal infrastructure in the approach of a tsunami [3]. One method for accurately approximating such a quantity is to base a mesh adaptation routine around achieving this *goal*, as investigated in this work.

Mesh adaptation relies upon an appropriate choice of error estimator. Commonly utilised strategies include gradient-based methods [4], Hessian-based methods [5] and 'explicit' estimators derived from a posteriori bounds [6]. For example, mesh adaptation under Hessian-based error estimation seeks to minimise the interpolation error with respect to some scalar field. The scalar field may be one which is related to the fluid flow, such as free surface height. We use the term *goal-oriented mesh adaptation* to refer to strategies which seek meshes permitting PDE solutions which minimise the error accrued in evaluating a QoI, whilst minimising the usage of computational resources.

The classical approach to goal-oriented error estimation uses the adjoint equation to compute sensitivities of the QoI to aspects of the PDE solution. In the pioneering works of [7, 8], the *dual weighted residual (DWR)* method was developed, providing linear and quadratic approximations to the QoI error, in terms of PDE residuals and the forward and adjoint solutions.

Dual weighted residual error estimation has been used to drive effective meshing strategies for a number of CFD applications (such as [9, 10]). The majority of

codes using DWR estimation do not incorporate directionality; an element is typically refined wherever the local DWR indicator breaches a pre-specified tolerance. This approach is particularly suited to quad-tree and oct-tree adaptive mesh refinement strategies, where there is an inherent hierarchical mesh structure (such as in [3]).

Metric-based mesh adaptation routines take a different approach, with meshes interpreted in a continuous sense using Riemannian metric fields [11]. An advantage of metric-based methods is that they can be used to produce anisotropic meshes, enabling the accurate resolution of strongly direction-dependent aspects of the fluid flow using relatively few elements. That is, the shape and orientation of mesh elements are taken into consideration, as well as their size. In its classical formulation, the DWR error estimator provides a scalar field, from which we are not able to immediately extract anisotropic information. However, a number of strategies have been proposed to achieve this (see, for example, [12, 13, 14]).

The work of [1] includes an alternative goal-oriented error analysis to the classical DWR formulation, providing an error estimator expressed in terms of an interpolation error, weighted by the adjoint solution. Within the metric-based framework, it is possible to produce anisotropic meshes from this formulation.

This paper includes a literature review of anisotropic approaches to goal-oriented mesh adaptation, including those referenced above. The main focus of this paper is a successful implementation of one isotropic and two anisotropic approaches to goal-oriented mesh adaptation within the Firedrake finite element solver framework. To the best of the authors' knowledge, this work provides the first implementation of these three approaches in the same code framework. Preliminary numerical experiments show that each approach is able to achieve a wide range of tolerated errors using fewer elements than would be required in a uniform mesh.

The organisation of this paper is arranged as follows. The metric-based mesh adaptation strategy is introduced in Section 2. The theory of goal-oriented error estimation is outlined in Section 3, along with a literature review of different strategies for using such estimators to construct anisotropic metric fields. Section 4 outlines the way in which a metric-based, goal-oriented mesh adaptation strategy has been implemented in this work, which is then used for numerical experiments in Section 5. These numerical experiments concern the TELEMAC-2D *'Point Discharge with Diffusion'* steady advection-diffusion test case, for which an analytical solution exists [15]. This enables us to validate the goal-oriented mesh adaptation strategies considered. Finally, conclusions are drawn and outlook discussed in Section 6.

2. METRIC-BASED MESH ADAPTATION

In this work we utilise a *metric-based* approach. This means that the mesh adaptation process is driven by a *Riemannian metric field* of dimension $n \times n$, where $\Omega \subset \mathbb{R}^n$ is the PDE domain.

A Riemannian metric field, or *metric*, denoted $\{\mathcal{M}(\mathbf{x})\}_{\mathbf{x}\in\Omega}$, is a collection of symmetric positive definite (SPD) linear forms defined pointwise, which contain local information on distances. Anisotropic edge lengths and element volumes can be derived from a metric. The main idea of metric-based mesh adaptation, introduced for the first time in [16], is to use a Riemannian metric space within the mesher to compute the necessary geometrical quantities, and to generate a unit mesh with respect to this Riemannian metric space.

A major advantage of the metric-based approach is that it enables control of mesh anisotropy, meaning that not only element size, but also shape and orientation may be dictated.

Given some error estimator η, our aim is to find a mesh $\mathcal{H}$ of the domain Ω such that the numerical solution of a PDE either: (a) achieves a certain level of error [12, 17]; or (b) minimises the interpolation error for a given number of mesh vertices [18, 1]. In this work we follow the former approach. For this, we need to establish the error estimator and express it as a metric.

Throughout this paper, we use the notation $\mathcal{H}$ when referring to meshes and K to denote elements thereof. Occasionally, the cell diameter function h is explicitly mentioned as a subscript, $\mathcal{H}_h$, if instructive. The edge set of element K is denoted ∂K, whose normal vectors are denoted $\widehat{\mathbf{n}}_K$. The indicator function which is unity on element K and zero elsewhere is denoted $\mathbb{1}_K$

2.1 Hessian-Based Metric

Suppose u is a (sufficiently smooth) scalar field of interest whose Hessian may be approximated in an element K as H. The interpolation error associated with a linear approximation $\mathcal{I}_h u$ is related to H through the truncated Taylor expansion of u by [19]

$$\|u - \mathcal{I}_h u\|_{\mathcal{L}_\infty(K)} \leq \gamma \max_{\mathbf{x}\in K} \max_{\mathbf{e}\in\partial K} \mathbf{e}^T |H(\mathbf{x})| \mathbf{e}, \qquad (1)$$

where $\gamma > 0$ is a constant related to the spatial dimension. Since H is symmetric, it has an orthogonal eigen-decomposition, $H = V\Lambda V^T$, meaning it makes sense to take the absolute value as

$$|H| = V|\Lambda|V^T. \qquad (2)$$

A metric tensor $\{\mathcal{M}(\mathbf{x})\}_{\mathbf{x}\in\Omega}$ may be defined as [17]

$$\mathcal{M}(\mathbf{x}) = \frac{\gamma}{\epsilon}|H(\mathbf{x})|, \qquad (3)$$

where $\epsilon > 0$ is the tolerated error level.

Since we take the absolute value for all eigenvalues, this ensures $\mathcal{M}(\mathbf{x})$ is SPD. Doing so is justified by the fact that we primarily care about the magnitude of errors and not their sign. The constant γ does not play an important role in the adaptation and is difficult to compute in practice. We follow [17] by setting $\gamma = 1$.

By specifying smaller values for the desired error ϵ in (3), we allow heightened mesh complexity in return for reduced interpolation error.

Since in practice the field u is only known in a discrete sense, its Hessian must be approximated using a recovery technique, which typically involves solving an auxiliary PDE. In this work, we use a *double* $\mathcal{L}_2$ *projection* approach (see [17] for details).

For further details on Hessian-based mesh adaptation, see [17, 19, 20].

2.2 Combining Metric Information

Suppose now that we have two Riemannian metrics, $\mathcal{M}_1$ and $\mathcal{M}_2$, each of which captures some aspect of the PDE solution, or an error estimate thereof. For instance, we may consider the Hessians of the solutions of both a PDE and its adjoint.

The most straightforward way to combine these metrics is to consider a convex combination

$$\mathcal{M} := \alpha\mathcal{M}_1 + (1-\alpha)\mathcal{M}_2, \tag{4}$$

for a parameter $\alpha \in (0,1)$. That (4) is indeed a metric follows from the definition of positive-definiteness:

$$\mathbf{y}^T\mathcal{M}(\mathbf{x})\mathbf{y} > 0,\ \forall \mathbf{y} \neq \mathbf{0}. \tag{5}$$

An advantage of using (4) is that the parameter α can be used to weight the resulting metric towards $\mathcal{M}_1$ or $\mathcal{M}_2$, as appropriate. In this work we consider exclusively the case where $\alpha = 0.5$, which we refer to as *metric averaging*. The geometrical meaning of the metric averaging is not intuitive. However, it is simple to implement and the numerical experiments in Section 5 illustrate that it can be an effective means of combining metric information in practice.

Alternatively, metrics can be combined using *superposition* (also known as *intersection*). This method of combining metric information will always yield meshes with more elements than those which would arise from the constituent metrics. Unlike with combination by (4), this approach is not weighted towards one metric, although the intersection order is in general non-commutative. For details on metric superposition, see pp.3778–3779 of [17].

An investigation into the properties of metric superposition and averaging was considered on pp.131-138 of [21]. In the context of unsteady adaptation applied to advection problems, metric superposition was found to deal better with shocks. However, metric averaging was found to be more effective at resolving sharp angles and small scale features.

2.3 Metric Normalisation

It is necessary to scale metrics before combining them using the methods outlined in Subsection 2.2. In the case where two metrics have been constructed from the Hessians of different fields, for example, there is no guarantee that these second derivative matrices are of the same order of magnitude. Normalising *before* combination means that the Hessians may be averaged or superimposed in a meaningful way.

For a Hessian H defined over a domain $\Omega \subset \mathbb{R}^n$ and fixed $p \in [1,\infty)$, $\mathcal{L}_p$ normalisation is defined by [22]

$$\mathcal{M}_{\mathcal{L}_p} = \frac{n}{\epsilon}\left(\int_\Omega \det(|H|)^{\frac{p}{2p+n}}\,\mathrm{d}\mathbf{x}\right)^{\frac{1}{p}} \det(|H|)^{\frac{-1}{2p+n}}|H|, \tag{6}$$

where ϵ is the tolerated error level. Taking the limit $p \to \infty$ yields the strategy referred to as $\mathcal{L}_\infty$ normalisation. This approach corresponds to the case where (6) reduces to (3), with constant $\gamma = n$. Whilst commonly used, the $\mathcal{L}_\infty$ normalisation strategy is not able to fully capture discontinuities and will in many cases use very high levels of mesh refinement surrounding such features. A low order $\mathcal{L}_p$ normalisation strategy, such as $p = 1$ or $p = 2$, is more suited to resolving such features and also permits multiscale mesh adaptation.

2.4 Metric Gradation

We use a metric gradation procedure to ensure that the sizes prescribed by the metric at neighbouring vertices do not vary by more than a specified threshold, taken here as $\beta = 1.4$. That is, the ratio of the prescribed sizes is bounded from above by β. This has the effect that meshes resulting from the metric-based mesh adaptation routine do not have sudden changes in resolution, which could act as artificial internal boundaries in the flow. For details on the metric gradation algorithm used, see [18].

3. GOAL-ORIENTED ERROR ESTIMATION

3.1 Forward and Adjoint Equations

Consider a PDE, referred to as the *forward equation*,

$$\Psi(u) = 0, \tag{7}$$

defined upon a domain Ω with piecewise smooth boundary $\partial\Omega$. For the purposes of this paper, suppose $\partial\Omega$ may be decomposed into the disjoint union of

open boundary, Dirichlet and Neumann components as $\partial\Omega = \partial\Omega_{\text{open}} \cup \partial\Omega_D \cup \partial\Omega_N$.

For a function space V containing the exact (forward) solution, u, for (7), consider a functional

$$J : V \to \mathbb{R}. \tag{8}$$

Here J describes an important scalar quantity related to the flow - the so-called *quantity of interest (QoI)*. Let $\langle \cdot, \cdot \rangle$ denote the usual $\mathcal{L}_2$ inner product on Ω and the $\mathcal{L}_2$ inner product over other spaces S be denoted $\langle \cdot, \cdot \rangle_S$. We restrict attention to quantities of interest which can be written as

$$J(v) = \langle g, v \rangle, \tag{9}$$

where g is some kernel function on Ω.

The variational formulation of (7) is given by

$$-\rho(u, v) = \langle \Psi(u), v \rangle = 0, \quad \forall v \in V. \tag{10}$$

We refer to $\rho(\cdot, v)$ as the *weak residual* of the PDE. Typically, integration by parts is applied in deriving this weak residual, so that Neumann boundary conditions may be applied and only first order derivatives appear. This also becomes important for the construction of goal-oriented error estimators in Subsection 5.2.

For a finite dimensional subspace $V_h \subset V$, we have the Galerkin approximation

$$-\rho(u_h, v_h) = \langle \Psi_h(u_h), v_h \rangle = 0, \quad \forall v_h \in V_h. \tag{11}$$

The (continuous) adjoint equation associated with (7) is given by

$$\left(\frac{\partial \Psi}{\partial u}(u)\right)^T u^* = \frac{\partial J}{\partial u}^T, \tag{12}$$

where u^* is the corresponding *adjoint solution*. The weak residual $\rho^*(\cdot, v)$ for the adjoint equation satisfies

$$\rho^*(u^*, v) = 0, \quad \forall v \in V, \tag{13}$$

where

$$\rho^*(u^*, v) = \left\langle \frac{\partial J}{\partial u}, v \right\rangle - \left\langle \left(\frac{\partial \Psi}{\partial u}(u)\right)^T u^*, v \right\rangle, \tag{14}$$

with appropriate application of integration by parts; similarly for the Galerkin approximation to (12).

Due to assumption (9) on the form of the QoI, we may rewrite the adjoint weak residual (14) as

$$\rho^*(u^*, v) = J(v) - \left\langle \left(\frac{\partial \Psi}{\partial u}(u)\right)^T u^*, v \right\rangle. \tag{15}$$

3.2 Estimate for Error in QoI

Due to the goal-oriented error analysis of [8], we have the classical result

$$J(u) - J(u_h) = \rho(u_h, u^* - u_h^*) + R, \tag{16}$$

where the remainder term R is quadratic in the forward and adjoint errors $e = u - u_h$ and $e^* = u^* - u_h^*$. A 'second-order' result is also derived, given by

$$J(u) - J(u_h) = \frac{1}{2}\rho(u_h, u^* - u_h^*) + \frac{1}{2}\rho^*(u_h^*, u - u_h) + \overline{R}, \tag{17}$$

where the remainder term $\overline{R}$ is cubic in the forward and adjoint errors. The remainder terms in (16) and (17) vanish if we have both a linear PDE and a quadratic QoI [8].

Goal-oriented error estimators may be derived from (16) and (17), as outlined in the following subsections.

Local error indicators η_K on each element K of a mesh $\mathcal{H}$ may be extracted from (16) by evaluating

$$\eta_K = |\rho(u_h, u^* - u_h^*)|_K| \,. \tag{18}$$

and from (17) by evaluating

$$\eta_K = \left| \frac{1}{2}\rho(u_h, u^* - u_h^*)|_K + \frac{1}{2}\rho^*(u_h^*, u - u_h)|_K \right|. \tag{19}$$

Summing over all elements yields a (global) error estimator

$$\eta := \sum_{K \in \mathcal{H}} \eta_K. \tag{20}$$

3.3 Isotropic Goal-Oriented Error Estimation

As mentioned in Subsection 3.1, integration by parts over Ω is typically applied when deriving the variational formulation (10) of (7). Evaluating error indicators of the form (18) typically involves integrating by parts again, *over each element, K*, yielding

$$\begin{aligned} \eta_K = \Big| & -\langle \Psi_h(u_h), u^* - u_h^* \rangle_K \\ & + \langle \psi_h^N(u_h), u^* - u_h^* \rangle_{\partial K \cap \partial\Omega_N} \\ & + \langle \psi_h^{\text{flux}}(u_h), u^* - u_h^* \rangle_{\partial K \setminus \partial\Omega} \Big|, \end{aligned} \tag{21}$$

where ψ_h^N is the residual of the Neumann boundary conditions and ψ_h^{flux} corresponds to flux terms between elements. For continuous Galerkin methods, the flux terms in ψ_h^{flux} arise purely due to the Neumann condition. For discontinuous FEM (such as *discontinuous Galerkin (DG)* methods [23]), there are also contributions due to the flux terms incorporated into the weak residual $\rho(\cdot, \cdot)$.

A similar expression to (21) results from applying integration by parts to (17).

As discussed in Section 1, classical DWR approaches are not usually used to give anisotropic meshes. In the case of metric-based mesh adaptation, the simplest way to create a metric $\{\mathcal{M}(\mathbf{x})\}_{\mathbf{x}\in\Omega}$ from (18) is to scale the identity matrix by an appropriate scalar field. However, one drawback of this approach is that (18) is piecewise constant and discontinuous across element interfaces ($\mathbb{P}0$), whereas we seek a piecewise linear metric which is continuous across element interfaces ($\mathbb{P}1$). Given an appropriate projection operator $\Pi_1 : \mathbb{P}0 \to \mathbb{P}1$, we may define an isotropic metric by

$$\mathcal{M} = \begin{bmatrix} \Pi_1\eta^{\mathbb{P}0} & \\ & \Pi_1\eta^{\mathbb{P}0} \end{bmatrix}, \quad \eta^{\mathbb{P}0} := \sum_{K\in\mathcal{H}} \eta_K \mathbb{1}_K \tag{22}$$

in the 2D case. One projection approach is to use a nodewise construction, taking averages of the values of η_K over adjacent cells, weighted by the area thereof [9]. Extension of (22) to the 3D case is trivial.

Metrics of the form (22) allow us to control the size of elements under mesh adaptation, but not their shape or orientation. The normalisation methods outlined in Subsection 2.3 should be applied to such isotropic metrics, as well as the anisotropic ones introduced in the following subsection.

3.4 Anisotropic Goal-Oriented Error Estimation

The literature contains a number of approaches for extending the goal-oriented error estimation framework to obtain anisotropic meshes. Some notable contributions are summarised in the following.

The approach advocated in [13] uses one mesh corresponding to each spatial dimension, which is refined uniformly in that direction alone. Local DWR indicators (18) are computed on each mesh and interpolated onto the base mesh, where a single anisotropic metric is formed. This approach essentially involves the computation of as many residuals as there are spatial dimensions, with the corresponding DWR estimators combined in an appropriate way so as to yield a single Riemannian metric field.

An alternative approach to DWR error estimation is considered in [1]. Therein, the result

$$J(u) - J(u_h) = \langle (\Psi_h - \Psi)(u), u^* \rangle + \widetilde{R} \tag{23}$$

is derived, where the remainder term $\widetilde{R}$ involves interpolation errors on V_h and the adjoint error, e^*. Here we have what may be interpreted as an interpolation error in the PDE, weighted by the adjoint solution.

If the PDE can be written in a conservative form,

$$\Psi(u) = \nabla \cdot \mathcal{F}(u) = 0, \tag{24}$$

then, by integration by parts in (23) and an application of the Cauchy-Schwarz inequality, [1]

$$\begin{aligned} |J(u) - J(u_h)| \leq \langle |(\mathcal{F} - \Pi_h\mathcal{F})(u)|, |\nabla u^*| \rangle_\Omega \\ + \langle \left|\widehat{n} \cdot (\overline{\mathcal{F}} - \Pi_h\overline{\mathcal{F}})(u)\right|, |u^*| \rangle_{\partial\Omega}, \end{aligned} \tag{25}$$

where $\overline{\mathcal{F}}$ embodies the boundary conditions and terms resulting from the integration by parts.

Since (25) involves interpolation errors in $\overline{\mathcal{F}}$ and $\mathcal{F}$, it is argued that we may use the interpolation error bound (1) to obtain an anisotropic metric by scaling components of the Hessians of these quantities by components of the adjoint and its gradient. That is, we define Riemannian metric fields

$$H^{\text{volume}} = \sum_{i=1}^{n}\sum_{j=1}^{m} |H(\mathcal{F}_i(u))| \left|\frac{\partial u_i^*}{\partial x_j}\right| \tag{26}$$

on the whole domain and

$$H^{\text{surface}} = \sum_{i=1}^{n} |u^*| \left| H\left(\sum_{j=1}^{m} \overline{\mathcal{F}}_j(u) \cdot n_j\right)\right| \tag{27}$$

on the domain boundary. Again, the modulus sign indicates that the Hessian eigenvalues are considered with a positive sign, to ensure positive-definiteness. The metrics resulting from (26) and (27) are superposed on the boundary in order to give a globally defined metric field. For further details, see [1], where this approach is shown to be effective in solving aerospace problems using the Euler equations. For the extension to unsteady problems, see [24, 25].

The approach of [12] also makes use of the Hessian to form an anisotropic metric. However, in this case, interpolation errors in the adjoint solution are considered, as opposed to interpolation errors in the PDE residual. We may interpret the error representation (16) as

$$J(u) - J(u_h) \approx \langle \Psi_h(u_h), u^* - \Pi_h u^* \rangle; \tag{28}$$

similarly

$$J(u) - J(u_h) \approx \langle \Psi_h^*(u_h^*), u - \Pi_h u \rangle \tag{29}$$

for the corresponding expression involving the adjoint residual.

An element-based anisotropic metric may then be constructed from (28) by weighting the Hessian of the adjoint solution with the residual in the forward equation:

$$\widetilde{\mathcal{M}}(K) = \|\Psi_h(u_h)\|_K \, |\overline{H}(u^*)|_K \tag{30}$$

and similarly, for (29):

$$\widetilde{\mathcal{M}^*}(K) = \|\Psi_h^*(u_h^*)\|_K \, |\overline{H}(u)|_K. \tag{31}$$

Here $\overline{H}$ indicates the average Hessian computed on a particular element.

As noted, (30) and (31) are defined elementwise, whereas we seek a (continuous) nodewise metric. In practice, we apply the same weighted-average projection operator as in (22) in order to yield a nodewise definition of the strong residual. Denoting these averaged quantities by $|\overline{\Psi_h}|$ and $|\overline{\Psi_h^*}|$, we may construct metrics $\{\mathcal{M}(\mathbf{x})\}_{\mathbf{x}\in\Omega}$ and $\{\mathcal{M}^*(\mathbf{x})\}_{\mathbf{x}\in\Omega}$ by

$$\begin{aligned}\mathcal{M}(\mathbf{x}) &= |\overline{\Psi_h}|_{\mathbf{x}}|H(u_h^*)|_{\mathbf{x}},\\ \mathcal{M}^*(\mathbf{x}) &= |\overline{\Psi_h^*}|_{\mathbf{x}}|H(u_h)|_{\mathbf{x}}.\end{aligned} \tag{32}$$

Note that in (32) Hessians in the forward and adjoint solutions are approximated by Hessians of the corresponding finite element solutions.

Another notable contribution in the field of anisotropic goal-oriented mesh adaptation is the work of [14]. The authors make use of upper bounds for interpolation errors of the forward and adjoint solutions (and gradients thereof) which are expressed in terms of the anisotropic quantities used by a metric-based mesh adaptation routine. As with the approaches of [1] and [12], metrics are defined using Hessians, due to their relation (1) with interpolation error.

Each of the approaches discussed above (along with many others which exist in the literature) have been used to provide logical, effective anisotropic extensions for goal-oriented mesh adaptation; these approaches are certainly worth further investigation. However, for simplicity of the presentation of this paper, we henceforth focus on the anisotropic approaches introduced in [12] and [1], along with the isotropic approach defined by (22).

Another reason for comparing the anisotropic approaches of [12] and [1] is that the former uses *a posteriori* error estimation, whilst the latter uses *a priori* error estimation. In this classification, we follow [1] in interpreting an a posteriori standpoint as assuming knowledge of errors accrued on an existing mesh and an a priori standpoint as assuming knowledge of the PDE solution. That is, the a posteriori metrics (30)–(31) involve residuals evaluated on the current mesh, whereas the a priori metrics (26)–(27) involve exact solutions for the forward and adjoint PDEs and make no reference to finite element solutions. In practice, these exact values are of course approximated by finite element solutions, given by replacing instances of u and u^* with u_h and u_h^*, respectively. However, the difference in classification has to do with the derivation of the methods, rather than implementation details.

For the reasons discussed above, we henceforth refer to meshes adapted using metrics (30)–(31) (and combinations thereof) as *a posteriori* and metrics derived from (26)–(27) as *a priori*. The isotropic approach is also a posteriori.

It is worth remarking that, whilst the anisotropic methods considered are inspired by the work of [12] and [1], there are some notable differences between the original implementations and those used in this work. Further details regarding these differences are discussed in Subsection 5.2.

4. IMPLEMENTATION

4.1 Error Estimate Evaluation

An important aspect of goal-oriented error estimate evaluation is the treatment of forward and adjoint error terms, $e = u - u_h$ and $e^* = u^* - u_h^*$, since these terms are clearly unknown. In the following, we assume the PDE is to be solved in $\mathbb{P}1$ space, although the same logic applies to other finite element spaces.

One approach for approximating e^* involves an auxiliary adjoint PDE. Given the current mesh $\mathcal{H}_h$, consider an *iso*-$\mathbb{P}2$ *refinement*, denoted $\mathcal{H}_{h/2}$. That is, we insert vertices wherever there would be quadrature nodes of a piecewise quadratic element, effectively quadrupling the number of elements. Henceforth, we refer to this as *uniform refinement*.

The adjoint PDE is solved on $\mathcal{H}_{h/2}$ in $\mathbb{P}2$ space, yielding an approximation $\widehat{u_{h/2}^*}$ to the adjoint solution u^*. On the same mesh, we obtain a linear approximation by projection, meaning the adjoint equation need only be solved once in order to approximate this error. We obtain an estimate for the adjoint error by projecting the difference back onto $\mathcal{H}_h$. In summary:

$$u^* - u_h^* \approx \widehat{u_{h/2}^*} - \Pi_1\widehat{u_{h/2}^*}, \tag{33}$$

where Π_1 is the projection operator into $\mathbb{P}1$ space.

This approach requires considerable additional computational resources, since the adjoint equation must be solved again on a fine mesh, in a higher order space. However, we illustrate in Section 5 that it provides proof of concept that the DWR error estimation strategy can be effectively deployed within our framework to yield high quality meshes. For nonlinear PDEs, one could argue that additional adjoint solves are less problematic than additional forward solves, since the adjoint PDE is always only linear.

An alternative approach performs local patch-based interpolation, using a supermesh of $\mathcal{H}_h$. The approximate adjoint solution u_h^* may then be interpolated onto the supermesh (see pp.870-871 of [14] for details).

Both of the strategies referred to above may also be applied to approximation of the primal error e, through additional solves of the forward PDE.

For the a posteriori anisotropic approach, we do not actually need to evaluate the adjoint error terms, since this is approximated using the Hessian. As such, the

above discussion is not of concern. In the a priori case, we follow the authors of [1] in approximating gradients in the exact adjoint solution by gradients in the numerical approximation thereof. As such, approximation of e and e^* is only required for isotropic metrics. In these cases, we make use of the space enrichment approach, providing a benchmark to which the anisotropic strategies may be compared.

4.2 Mesh Adaptation Approach

In this subsection the mesh adaptive solver routine is outlined for stationary PDEs.

Given an initial mesh $\mathcal{H}_0$ and a desired error level $\epsilon > 0$, we construct a sequence of meshes $\{\mathcal{H}_i\}_{i\in\mathbb{N}}$ by iteratively solving the PDE and its adjoint, evaluating an appropriate error estimator and extracting an associated metric field. See Algorithm 1 for a workflow representation.

```
Given desired error ε > 0;
Given initial mesh H_0;
Set i := 0;
while not converged do
    Solve PDE on H_i;
    Solve adjoint PDE on H_i;
    Evaluate error estimator and QoI;
    Extract metric M_i from error estimator for
      desired error ε;
    Apply metric gradation to M_i;
    Adapt mesh using M_i to obtain H_{i+1};
    if source term f is not analytic then
        Interpolate f onto H_{i+1};
    end
    Increment i;
end
```

Algorithm 1: Mesh adaptation routine.

Convergence is attained when either the change in QoI or the number of mesh elements falls below some relative tolerance, both taken here as 0.5%, as recommended in [12]. A maximum iteration count of 35 is imposed. In practice, assuming that $\mathcal{H}_0$ is relatively isotropic, we find that at least three mesh iterations are required in order to introduce anisotropy into meshes resulting from Algorithm 1. As such, we require that at least three iterations are performed.

Under the assumption that source terms are described analytically, there is no need to interpolate across meshes, since the source term can be defined in a mesh-independent way. This is the case considered in the numerical experiments of Section 5. If the source term is defined using data then interpolation will be required whenever a new mesh is considered.

In this work we restrict attention to steady problems, with unsteady problems to be considered in the future. In such problems, solutions must be transferred from one mesh to another between timesteps.

4.3 Software

Firedrake is a Python based finite element package with a high-level interface which is intuitive from the perspective of a mathematician [26]. PDEs in weak form are expressed using *Unified Form Language (UFL)* [27] in a very similar way as they would be written by hand. Firedrake uses sophisticated code generation technology to produce efficient low-level code. This automatically generated C code uses PETSc [28, 29] to solve the resulting linear and nonlinear systems.

Anisotropic mesh adaptation is achieved in Firedrake using the Pragmatic library [30]. This C++ library takes as input an arbitrary mesh and a metric field over it and returns a mesh that is adapted to the metric. The adapted mesh is derived from the input one through a series of mesh modifications (edge splits, collapses and swaps and vertex smoothing in the 2D case) that optimise the lengths of the edges as well as a certain quality functional. More detail can be found in [31, 32, 33]. Pragmatic and Firedrake are interfaced through PETSc [34, 35]. High-level error estimate information is passed to Pragmatic using Riemannian metric fields constructed in Firedrake using the FEM solutions of the PDE and its adjoint.

Conservative projection [36] between meshes in Firedrake is enabled by the libsupermesh library [37, 38]. Evaluating DWR error estimators using the higher order finite element solution (33) requires these projection operators to transfer information to and from the iso-$\mathbb{P}2$ refined space, upon which the approximation to $u^* - u_h^*$ is constructed.

For the PDE considered in the Section 5 of this paper, the continuous form adjoint is easily derived and may be solved in the same manner as the forward equation. For more complex PDEs or boundary conditions, Firedrake also supports the automatic generation and solution of the adjoint PDE in a discrete adjoint type approach, using pyadjoint [39, 40]. Note that stabilising the continuous adjoint equation using the same strategy as for the forward PDE is not in general equivalent to generating the discrete adjoint equation from the stabilised PDE. For example, these discretisations for the adjoint PDE are not equivalent for SUPG stabilisation applied to advection-diffusion problems, which we consider in Section 5.

The Firedrake and PETSc versions used in this work are archived using Zenodo as [41]. The simulation code used to generate results in Section 5 is archived as [42].

5. NUMERICAL EXPERIMENTATION

5.1 Forward and Adjoint Equations

For the purposes of numerical experimentation, consider steady advection-diffusion of a passive tracer ϕ immersed in a fluid with velocity $\mathbf{u}$ and diffusivity ν:

$$\begin{cases} \mathbf{u}\cdot\nabla\phi - \nabla\cdot(\nu\nabla\phi) &= f \quad \text{in } \Omega \\ \nu\widehat{\mathbf{n}}\cdot\nabla\phi &= g_N \quad \text{on } \partial\Omega_N \\ \phi &= g_D \quad \text{on } \partial\Omega_D, \end{cases} \tag{34}$$

where f is a source term. For a function space V, consider the weak formulation

$$a(\phi, v) = L(v), \quad \forall v \in V, \tag{35}$$

where

$$\begin{aligned} a(\phi, v) :=& \langle \mathbf{u}\cdot\nabla\phi, v\rangle + \langle \nu\nabla\phi, \nabla v\rangle \\ &- \langle \nu\widehat{\mathbf{n}}_K\cdot\nabla\phi, v\rangle_{\partial\Omega\setminus\partial\Omega_N}, \\ L(v) :=& \langle f, v\rangle + \langle g_N, v\rangle_{\partial\Omega_N}. \end{aligned} \tag{36}$$

Thus $\rho(\cdot, v) \equiv L(v) - a(\cdot, v)$. Stabilisation terms are usually added to (35) in order to control under- and over-shoots that are typical with advection-dominated problems. In this work we use (continuous) Lagrange ($\mathbb{P}1$) finite elements and *Streamline Upwind Petrov Galerkin (SUPG)* stabilisation, under which an additional term $\langle \Psi(\phi), \mathbf{u}\cdot\nabla v\rangle$ is used, where

$$\Psi(\phi) = f - \mathbf{u}\cdot\nabla\phi + \nabla\cdot(\nu\nabla\phi) \tag{37}$$

denotes the strong residual for (34). For further details on the stabilisation strategy, see [43].

The continuous adjoint equation associated with (34) may be derived as

$$\begin{cases} -\nabla\cdot(\mathbf{u}\phi^*) - \nabla\cdot(\nu\nabla\phi^*) &= \frac{\partial J}{\partial\phi} \quad \text{in } \Omega \\ \phi^* &= 0 \quad \text{on } \partial\Omega\setminus\partial\Omega_N \\ \nu\widehat{\mathbf{n}}\cdot\nabla\phi^* + \phi^*\mathbf{u}\cdot\widehat{\mathbf{n}} &= 0 \quad \text{on } \partial\Omega\setminus\partial\Omega_D. \end{cases} \tag{38}$$

Note that the derivative of the QoI with respect to the forward solution provides a source term for the adjoint equation. SUPG stabilisation applied to the adjoint equation requires the adjoint strong residual,

$$\Psi^*(\phi^*) = \frac{\partial J}{\partial\phi} + \nabla\cdot(\mathbf{u}\phi^*) + \nabla\cdot(\nu\nabla\phi^*). \tag{39}$$

5.2 Goal-Oriented Metrics

Formulating the DWR estimator amounts to applying integration by parts on each element. For the stabilised version of the CG formulation given in (35), this yields

$$\begin{aligned} &\rho_h(\phi_h, \phi^* - \phi_h^*)|_K \\ &\quad = \langle \Psi(\phi_h), \phi^* - \phi_h^*\rangle_K \\ &\qquad + \langle \nu\widehat{\mathbf{n}}_K\cdot\nabla\phi_h - g_N, \phi^* - \phi_h^*\rangle_{\partial K\cap\partial\Omega_N} \\ &\qquad + \langle \nu\widehat{\mathbf{n}}_K\cdot\nabla\phi_h, \phi^* - \phi_h^*\rangle_{\partial K\setminus\partial\Omega} \\ &\qquad + \langle \Psi(\phi_h), \mathbf{u}\cdot\nabla(\phi^* - \phi_h^*)\rangle_K, \end{aligned} \tag{40}$$

where the terms on the RHS correspond to those given in (21), along with the stabilisation error term. Observe that if ϕ_h is replaced with the (assumed smooth) exact solution of (34) then this estimator vanishes. Similarly, if $\phi^* - \phi_h^*$ is replaced with the adjoint finite element solution $\phi_h^* \in V_h$ then the estimator vanishes, because ϕ_h is a solution in the finite element space V_h. When summed over all elements of the mesh, the inner product over $\partial K\setminus\partial\Omega$ in (40) corresponds to a flux jump term.

For the stabilised adjoint equation, we have the DWR estimator

$$\begin{aligned} &\rho_h^*(\phi_h^*, \phi - \phi_h)|_K \\ &\quad = \langle \Psi^*(\phi_h^*), \phi - \phi_h\rangle_K \\ &\qquad + \langle \nu\widehat{\mathbf{n}}_K\cdot\nabla\phi_h^* + \phi_h^*\mathbf{u}\cdot\widehat{\mathbf{n}}_K, \phi - \phi_h\rangle_{\partial K\setminus(\partial\Omega\setminus\partial\Omega_D)} \\ &\qquad + \langle \Psi^*(\phi_h^*), \mathbf{u}\cdot\nabla(\phi - \phi_h)\rangle_K, \end{aligned} \tag{41}$$

where the boundary and flux terms have been combined due to their similarity.

Isotropic metrics may be constructed from (40) and (41) using (22). The strong residual in the forward PDE required for constructing the anisotropic metric (30) is given by (37), whilst the strong residual in the adjoint PDE required for (31) is given by (39).

For metrics constructed from (26)–(27) to make sense, we require the PDE (34) to be in conservative form. Due to the constant velocity field, it may be expressed in terms of a potential functional $\mathcal{F}$ as

$$\nabla\cdot\mathcal{F}(\phi) = f, \quad \mathcal{F}(\phi) = \mathbf{u}\phi - \nu\nabla\phi. \tag{42}$$

However, the source term f requires special treatment. In this work we neglect the boundary metric terms arising from (27) and construct a metric by summing (26) and the Hessian of the source term:

$$\begin{aligned} \mathcal{M} =& |H(\mathcal{F}_1(\phi))| \left|\frac{\partial\phi^*}{\partial x}\right| + |H(\mathcal{F}_2(\phi))| \left|\frac{\partial\phi^*}{\partial y}\right| \\ &+ |H(f)|\,|\phi^*|. \end{aligned} \tag{43}$$

A similar strategy is used for the corresponding adjoint metric. That is, for a potential functional $\mathcal{G}$ satisfying

$$\nabla\cdot\mathcal{G}(\phi^*) = g, \quad \mathcal{G}(\phi^*) = -\mathbf{u}\phi^* - \nu\nabla\phi^*, \tag{44}$$

we define an adjoint metric

$$\begin{aligned} \mathcal{M} =& |H(\mathcal{G}_1(\phi^*))| \left|\frac{\partial\phi}{\partial x}\right| + |H(\mathcal{G}_2(\phi^*))| \left|\frac{\partial\phi}{\partial y}\right| \\ &+ |H(g)|\,|\phi|, \end{aligned} \tag{45}$$

recalling that $g = \frac{\partial J}{\partial\phi}$. To the best of the authors' knowledge, no a priori error result for the adjoint equation equivalent to (23) exists in the literature.

Nonetheless, extending the a priori metric construction by combining the forward metric (43) with the adjoint metric (45) is shown in Section 5 to provide meshes upon which the QoI may be accurately approximated.

Following (6), we use $\mathcal{L}_1$ metric normalisation for all goal-oriented mesh adaptation strategies. Using $\mathcal{L}_1$ normalisation, as opposed to the more commonly applied $\mathcal{L}_\infty$ normalisation, allows us to better capture the discontinuities associated with the source terms in the forward and adjoint equations.

It is worth remarking that, whilst stabilisation errors have been accounted for in (40) and (41), they are not considered in either of the anisotropic metric techniques considered in this paper. An investigation of how best to integrate the stabilisation error into these anisotropic metrics is something to be considered in future work. However, stabilisation errors are accounted for in the anisotropic metrics constructed in [14].

5.3 Point Source Test Case

For this work we consider the *'Point Discharge with Diffusion'* test case from TELEMAC-2D validation document version 7.0 [15], whose source is a delta function at the point $\mathbf{x}_0 = (x_0, y_0) = (1, 5)$ in the domain $\Omega = [0, 50] \times [0, 10]$.

The fluid velocity $\mathbf{u} = (u_x, u_y) \equiv (1, 0)$ is entirely in the x-direction and the diffusivity $\nu \equiv 0.1$ is constant. An inflow condition of $\phi \equiv 0$ is imposed at the boundary where $x = 0$, along with free-slip conditions for $y \in \{0, 10\}$ and an open boundary at $x = 50$. The only difference in the test case considered here is that we consider the point source to be located at $\mathbf{x}_0 = (2, 5)$, so that it is not as close to the inflow boundary. We consider the same initial mesh as in [15], comprised of a uniform grid of 4,000 right-angled triangles.

The quantities of interest considered take the form

$$J_i(\phi) = \int_\Omega \mathbb{1}_{R_i} \phi \,\mathrm{d}x = \int_{R_i} \phi \,\mathrm{d}x, \tag{46}$$

where $R_i \subset \Omega$ are 'receiver' regions, for $i \in \{1, 2\}$. That is, each kernel $g = \mathbb{1}_{R_i}$ is an indicator function. Goal-oriented mesh adaptation seeks a mesh which permits an accurate approximation of the tracer concentration over the receiver region. Both (46) and (34) are linear, meaning that the error estimates provided by (16) and (17) are *exact* [8].

For a related application in environmental science, consider the inlet and outlet pipes of a desalination plant. The former pipe brings sea water into the plant, extracting from region R_i. Salt is extracted from this water and the remaining salt residues are released back into the ocean through the latter pipe, depositing at $\mathbf{x}_0$. An undesirable (but entirely possible) situation is the one in which a significant quantity of salt from the outlet pipe is later taken back into the plant at the inlet, making the task of desalination more difficult.

For this application we do not necessarily care about the wider dispersal of high salinity water, but rather achieving an efficient and accurate calculation of the salinity at the inlet pipe. If salt is interpreted as a passive tracer in the fluid flow, we can model its advection and diffusion using (34) and quantify its concentration at the inlet using (46). Through goal-oriented mesh adaptation, we are able to select a mesh which gives a good approximation to the salinity at the inlet, whilst retaining a relatively small number of elements.

As in the similar steady-state advection-diffusion test case examined in [12], we consider two scenarios: one where the receiver is directly downstream from the source and one where it is offset to one side of the channel. Given the definition of the disc

$$B_\epsilon(\mathbf{y}) = \{\mathbf{x} \in \Omega \mid \|\mathbf{x} - \mathbf{y}\|_2 \leq \epsilon\}, \tag{47}$$

we choose $R_1 = B_{\frac{1}{2}}((20, 5))$ in the former instance and $R_2 = B_{\frac{1}{2}}((20, 7.5))$ in the latter.

As documented in [15], the analytical solution for this problem is given by

$$\phi_{\text{exact}}(\mathbf{x}) = \frac{q}{2\pi\nu} \exp\left(\frac{u_x x}{2\nu}\right) K_0\left(\frac{u_x \|\mathbf{x} - \mathbf{x}_0\|}{2\nu}\right), \tag{48}$$

where q denotes the inflow discharge at the source and K_0 is the modified Bessel function of the second kind and order zero.

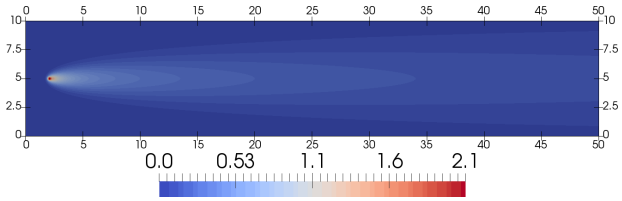
Evaluating the QoI amounts to integrating (48) over region R_1 or R_2, as appropriate. Due to the presence of the modified Bessel function, we are not able to perform this integration analytically and hence resort to applying numerical quadrature on a sequence of increasingly refined uniform meshes. For the cases of both centred and offset receiver regions, we attain convergence to five decimal places on a mesh with 1,024,000 elements, as illustrated in Table 1. The close agreement between QoI values under analytical and finite element solutions validates our $\mathbb{P}1$ SUPG FEM solution strategy.

Thus the benchmark values $J_1^{\text{exact}} = 0.16344$ and $J_2^{\text{exact}} = 0.06959$ are obtained using the analytical solution evaluated on this mesh. The analytical solution (48) is presented in Figure 1a on this mesh, along with finite element approximations to the forward and adjoint solutions. In each case, $\mathbb{P}1$ elements are used.

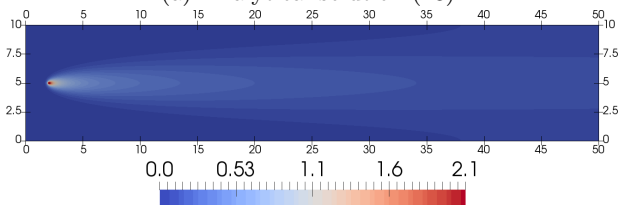
An initial qualitative observation to be made from Subfigures 1a–1b and Table 1 is that the piecewise linear finite element approximation on the fine uniform

Elements	$J_1(\phi)$	$J_1(\phi_h)$	$J_2(\phi)$	$J_2(\phi_h)$
4,000	0.20757	0.20547	0.08882	0.08901
16,000	0.16904	0.16873	0.07206	0.07205
64,000	0.16263	0.62590	0.06924	0.06922
256,000	0.16344	0.16343	0.06959	0.06958
1,024,000	0.16344	0.16345	0.06959	0.06958

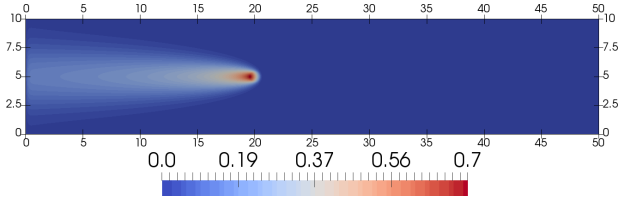
Table 1: Convergence of QoIs J_1 and J_2 under analytical and finite element solutions on a sequence of uniform meshes. Columns labelled $J_i(\phi)$ correspond to analytical solutions, whilst columns labelled $J_i(\phi_h)$ correspond to finite element solutions.



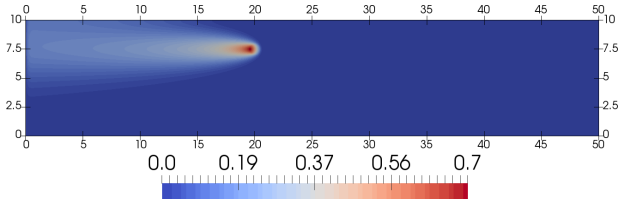
(a) Analytical solution (48).



(b) Finite element forward solution.



(c) Finite element adjoint solution corresponding to QoI J_1 and receiver R_1.



(d) Finite element adjoint solution corresponding to QoI J_2 and receiver R_2.

Figure 1: Analytical and finite element solutions for the TELEMAC-2D 'Point Discharge with Diffusion' validation experiment [15], along with adjoint solutions corresponding to QoIs J_1 and J_2. Each field is presented on a 1,024,000 element uniform mesh.

mesh provides an excellent approximation to the analytical solution. It appears from Subfigures 1a–1b that the only region where there is a noticeable difference between the analytical and finite element solutions is near the boundary for $x \in [30, 50]$. This is in agreement with what may be observed in the TELEMAC-2D solution given in [15].

Whilst (34) has a point source term at $\mathbf{x}_0 = (2, 5)$, the corresponding adjoint equations have source terms in regions R_1 and R_2, which are discs with radius 0.5. As such, we observe from Subfigures 1c–1d that the source regions are indeed larger than in the forward PDE. Whilst in the forward PDE information flows in the direction of positive x, we observe from Subfigures 1c–1d that information flows in the opposite direction for the adjoint PDE, as is to be expected.

The purpose of the numerical experimentation described in this subsection is to obtain meshes from goal-oriented error metrics, upon which J_1^{exact} and J_2^{exact} may be accurately approximated (for some desired error level ϵ), under the condition that the meshes have as few elements as possible. By fulfilling this purpose, we illustrate the successful implementation of goal-oriented mesh adaptation.

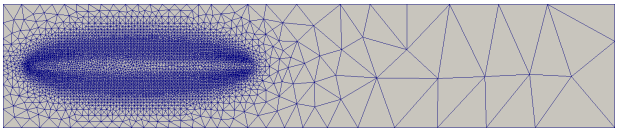
We do not intend to make a rigorous performance analysis, since the problem size considered is relatively small; larger scale problems are required for testing run-time. Instead, we capitalise on the existence of the analytical solution (48), using it to validate our approach and thereby imply that the mesh adaptation strategies considered here may be readily applied to more complicated PDE problems.

5.3.1 Centred Receiver Region

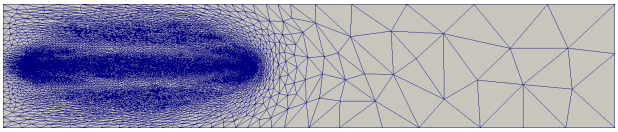
Firstly, we consider the case of R_1, wherein the receiver is directly downstream of the source. We consider three different goal-oriented approaches for residuals in the forward PDE: isotropic mesh adaptation based on (22); a posteriori anisotropic mesh adaptation based on (30); and a priori anisotropic mesh adaptation based on (43). Henceforth, we shall use these names when referring to the three approaches.

Figure 2 illustrates a selection of meshes adapted using these strategies. In each mesh, we observe that there is coarse resolution downstream of the receiver location. Given that this problem is advection-dominated, this is to be expected, as it indicates the insensitivity of the tracer concentration at the receiver to what happens downstream from there. Additionally, in each case there is a region of high resolution surrounding the point source, which enables us to accurately capture the dynamics thereof.

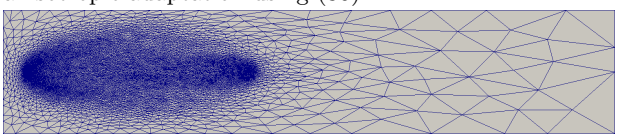
As expected, we observe elements with significantly more anisotropy in the meshes displayed in Subfigures 2b and 2c than in 2a. Note that a posteri-



(a) Mesh with 12,246 elements resulting from isotropic adaptation using (22).



(b) Mesh with 16,407 elements resulting from a posteriori anisotropic adaptation using (30).



(c) Mesh with 44,894 elements resulting from a priori anisotropic adaptation using (43).

Figure 2: Meshes generated using goal-oriented metrics as indicated, for receiver region R_1.

ori anisotropic metrics constructed using (30) contain the Hessian of the adjoint solution. The influence of this Hessian is visible as the three bands propagating against the flow in Subfigure 2b, relating to regions where the second derivatives are most significant.

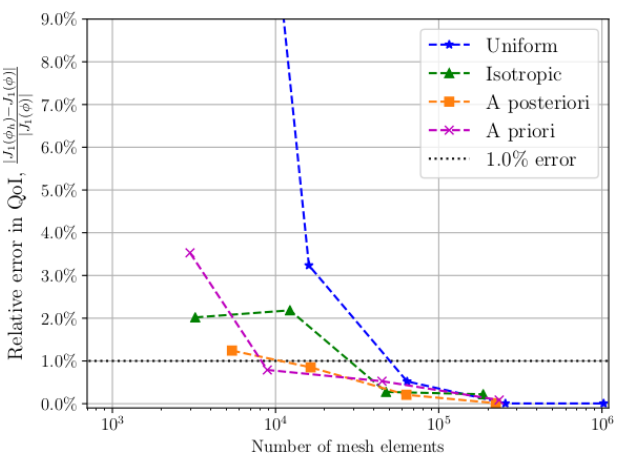


Figure 3: Relative error in J_1^{exact} for finite element solutions computed on meshes arising from: uniform refinement; isotropic adaptation; a posteriori anisotropic adaptation; and a priori anisotropic adaptation.

Figure 3 plots the relative error in approximating J_1^{exact} for a sequence of mesh adaptations. This sequence is generated using decreasing values of $\epsilon > 0$, to specify the desired error level. For large values of ϵ, the resulting goal-oriented meshes are often extremely coarse, with an insufficient number of elements to capture the QoI at all, leading to a zero value. These data are excluded from the figures displayed in this work.

Table 1 shows convergence of the QoI to its analytical value, J_1^{exact}, on a sequence of uniform meshes. Figure 3 illustrates that this also occurs under all three goal-oriented mesh adaptive approaches.

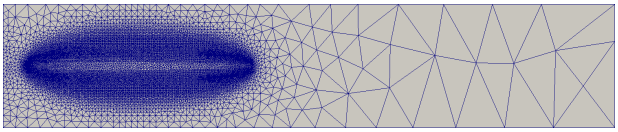
For a given accuracy level relative to the converged QoI value, all three goal-oriented meshing strategies are able to achieve this using fewer elements than required using uniform meshes. Further, the anisotropic approaches outperform the isotropic method in the sense that they require fewer elements to reach the 1% accuracy level. However, the isotropic approach attains 2% accuracy using as few as 3,000 elements and the three methods yield fairly similar results overall.

Whilst it appears from Figure 3 that the anisotropic metrics do not lead to significant gains over isotropic ones, it is important to recall that the latter approach requires an auxiliary FEM solve in a higher order, iso-$\mathbb{P}2$ refined space. In particular, the construction of the isotropic metric involves approximating the adjoint solution in an enriched space. This additional FEM solve is inherently more computationally intensive than solving the adjoint equation in the unenriched space, meaning a significant computational cost is accrued. This motivates using anisotropic methods, for which no such auxiliary FEM solves are required.

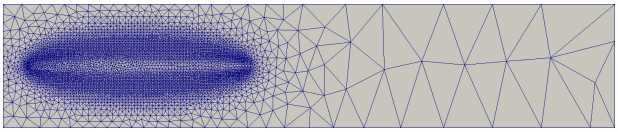
Next we consider combining the metrics used above with the corresponding metrics associated with the adjoint equation. For this, we use the metric averaging and superposition operators defined in Subsection 2.2. For the isotropic approach, we combine the forward metric used above with an isotropic metric constructed using (41) and (22). The a posteriori anisotropic approach combines (30) and (31), whilst the a priori approach combines (43) and (45).

In each case illustrated in Figure 4, we observe a degree of symmetry in the region $x \in [2, 20]$ about the line $x = 11$, which marks halfway between the source and receiver. This symmetry follows from the similarity of the forward and adjoint solutions referred to in Subfigures 1b and 1c. One noticeable break in this symmetry is exhibited in Subfigures 4e and 4f, where the different forms of the forward and adjoint source terms (point source and disc) are clearly visible. The high element count in Subfigure 4e is due to many elements being deployed in order resolve the discontinuous source terms in the forward and adjoint equations - a feature which is also visible in Subfigure 4f.

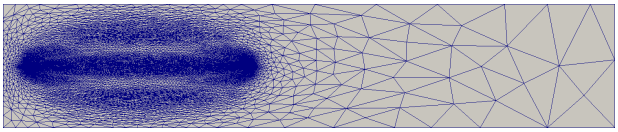
As observed in Figure 2, the goal-oriented strategies advocate low resolution downstream of the receiver, since the QoI is insensitive to what happens there.



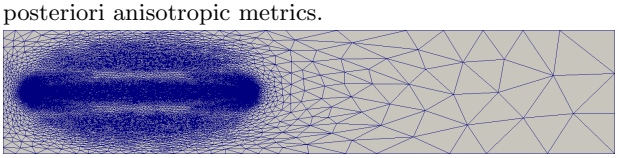
(a) Mesh with 26,792 elements resulting from averaging isotropic metrics.



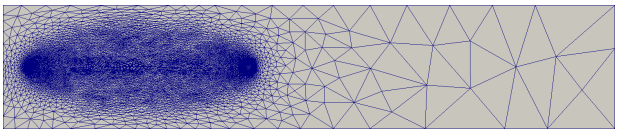
(b) Mesh with 12,300 elements resulting from superposing isotropic metrics.



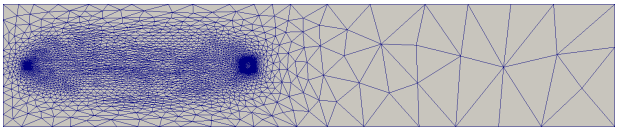
(c) Mesh with 16,538 elements resulting from averaging a posteriori anisotropic metrics.



(d) Mesh with 26,116 elements resulting from superposing a posteriori anisotropic metrics.



(e) Mesh with 45,108 elements resulting from averaging a priori anisotropic metrics.



(f) Mesh with 18,383 elements resulting from superposing a priori anisotropic metrics.

Figure 4: Meshes generated using combined goal-oriented metrics as indicated, for receiver region R_1.

A first observation to be made from the relative error plot in Figure 5 is that all goal-oriented meshing strategies considered succeed in yielding solutions with the same level of error as is attained on a uniform mesh, but with fewer elements.

Compared with the results shown in Figure 3, we observe in Figure 5 that both superposing and averaging isotropic metrics provide meshes upon which the 1% error level may be attained using fewer elements. We deduce that the isotropic adjoint error metric contains information which improves the error estimation.

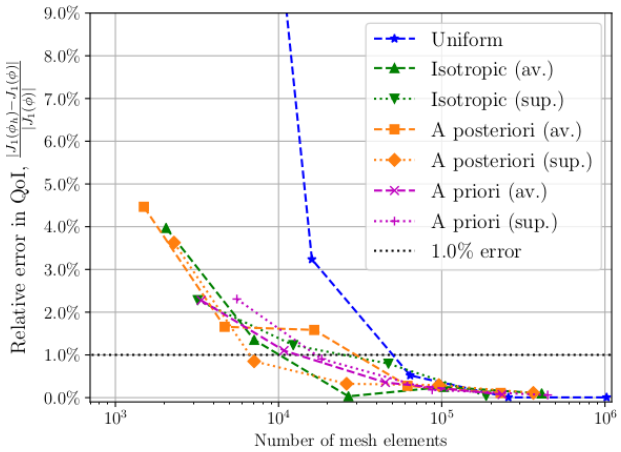


Figure 5: Relative error in J_1^{exact} resulting from mesh adaptation routines which combine forward and adjoint error metrics. Here *av.* and *sup.* denote 'average' and 'superposition', respectively.

For the anisotropic approaches, there does not appear to be any clear improvement attained under metric combination for the centred receiver case. One possible explanation for this is that the forward anisotropic metrics already recommend relatively high resolution in the directly downstream of the source, as is visible in Subfigures 2b and 2c. As such, the information associated with the adjoint metric is not so important in the case of a centred receiver region.

However, the superposition and averaging of metrics are shown to yield meshes with 2,000-3,000 elements, upon which the QoI error is less than 5%. This is in contrast with the initial 4,000 element uniform mesh, upon which the corresponding error is 25.7%.

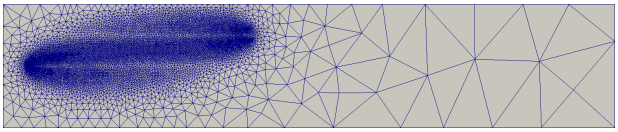
In conclusion, the above experiments validate the three goal-oriented mesh adaptation implementations in the case where the receiver is directly downstream from the source.

5.3.2 Offset Receiver Region

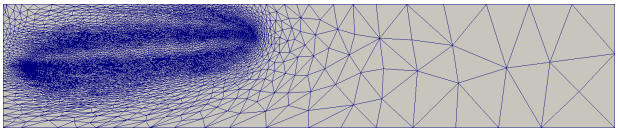
Secondly, we consider the case with an offset receiver region, indicated by R_2.

Again, all three meshes shown in Figure 6 use coarse resolution downstream of the receiver region. Additionally, relatively coarse resolution is used in the lower part of the domain. This is to be expected, since the sensitivity of the QoI to what happens in this region is negligible, as indicated in Subfigure 1d.

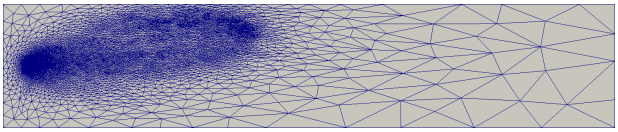
Whilst the isotropic metric recommends heightened mesh resolution in a band spanning the region between source and receiver (as seems reasonable for a goal-oriented strategy), both anisotropic strategies



(a) Mesh with 19,399 elements resulting from isotropic adaptation using (22).



(b) Mesh with 9,868 elements resulting from anisotropic adaptation using (30).



(c) Mesh with 29,143 elements resulting from anisotropic adaptation using (43).

Figure 6: Meshes generated using goal-oriented metrics as indicated, for receiver region R_2.

prefer to use high resolution surrounding the source and receiver. In the case of the a priori anisotropic metric, this is likely due to the Hessians of the source terms. In the forward equation, the source term is a delta function, whilst in the adjoint equation it is an indicator function. As such, both of these source terms are discontinuous (and hence non-differentiable) on the edges of the source regions. However, our application of $\mathcal{L}_1$ normalisation means that the resulting goal-oriented meshes are able to capture these discontinuous features. In the context of QoI error estimation, we accept additional mesh resolution surrounding region R_2, since this ultimately permits a more accurate approximation.

As with the centred receiver, Figure 7 illustrates all three goal-oriented approaches eventually converging to J_2^{exact}. Additionally, both a posteriori goal-oriented approaches require fewer elements to attain the 1% relative error level than on a uniform mesh. The anisotropic a posteriori approach requires particularly few elements to attain this threshold - around 10,000.

The a priori approach offers a reduction in error over uniform meshing if 10,000 elements or fewer are to be used. On the other hand, this strategy appears to offer no significant improvement over using uniform meshes if more than 10,000 elements are to be used. It is likely that the a priori anisotropic approach suffers from over-resolving the source region at the expense of under-resolving the receiver, for the reasons discussed above. Examining Subfigure 6c, it appears that mesh resolution is more focused on the source than the receiver, supporting this argument. That the a priori approach uses high resolution around the (discontinuous) source term is due to the inclusion of its Hessian.

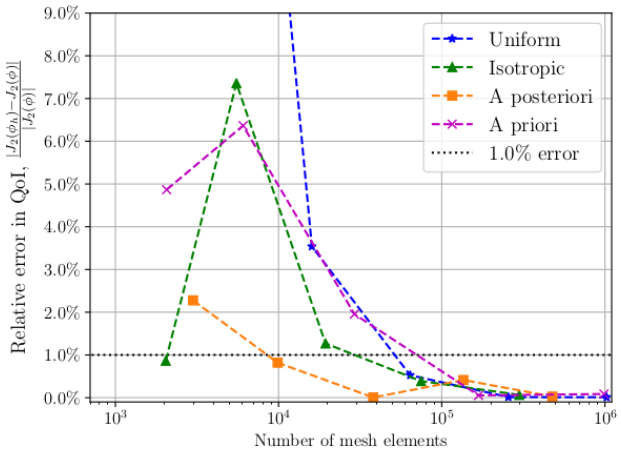


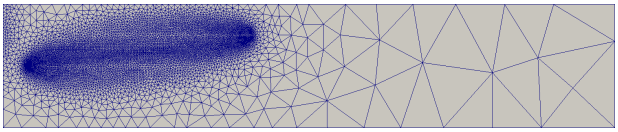
Figure 7: Relative error in J_2^{exact} and error estimator (16) evaluated on meshes arising from: uniform refinement; isotropic adaptation; a posteriori anisotropic adaptation; and a priori anisotropic adaptation.

The leftmost data point for the isotropic approach indicates an error less than 1% using even fewer elements: 2,035. However, the next data point suggests that the first is in fact a QoI error which is anomalously small. On closer examination of the 2,035 element mesh, the receiver is covered by only very few elements, suggesting that the QoI value may only be coincidentally close to the analytical value. Indeed, this potential anomaly warrants further investigation in future work.

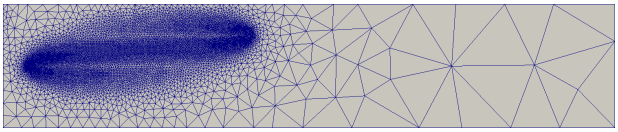
Finally, we consider combined forward and adjoint metrics related to the case of a receiver which is offset from the centre of the channel.

Figure 8 illustrates many of the phenomena which have already been discussed, including the lack of unnecessary downstream resolution, as well as large numbers of elements used near to source and receiver regions in Subfigures 8e and 8f.

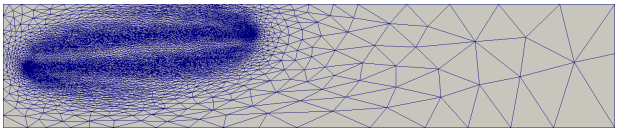
As in the previously considered cases, Figure 9 shows all goal-oriented approaches achieving relative errors below the 1% level using fewer elements than required on uniform meshes. Further, all three goal-oriented approaches seem to be improved by inclusion of adjoint error estimator information, because smaller errors may be attained using the same number of elements as previously. These improvements are in contrast to Figure 5, where the additional information from the adjoint metric does not significantly improve the QoI approximation. For receiver regions which are



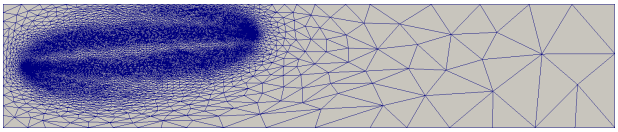
(a) Mesh with 13,980 elements resulting from averaging isotropic metrics.



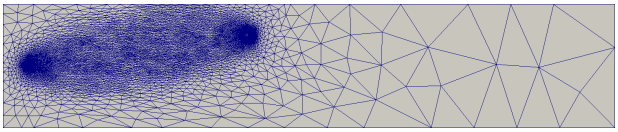
(b) Mesh with 19,588 elements resulting from superposing isotropic metrics.



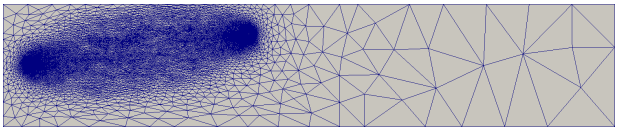
(c) Mesh with 9,289 elements resulting from averaging a posteriori anisotropic metrics.



(d) Mesh with 14,470 elements resulting from superposing a posteriori anisotropic metrics.



(e) Mesh with 25,204 elements resulting from averaging a priori anisotropic metrics.



(f) Mesh with 49,793 elements resulting from superposing a priori anisotropic metrics.

Figure 8: Meshes generated using combined goal-oriented metrics as indicated, for receiver region R_2.

offset from the centre of the flow, it appears that the adjoint metric provides useful information for obtaining meshes which yield accurate QoI estimates.

The superposition of a posteriori anisotropic metrics results in small errors using particularly few elements, as does the averaging of isotropic metrics. The anomaly observed in Figure 7 for the isotropic metric seems to also appear here, although to a lesser extent.

In Subsection 5.2 we remark that it is not known whether the adjoint metric associated to the a priori approach relates to a rigorous error estimate. However, the fact that the error may be reduced by inclusion of this information indicates that it certainly contains practical value.

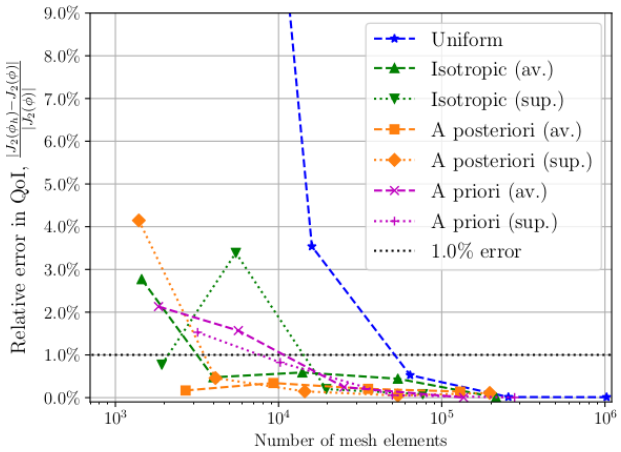


Figure 9: Relative error in J_2^{exact} resulting from mesh adaptation routines which combine forward and adjoint error metrics. Notation as in Figure 5.

That the relative error is so consistently small for each anisotropic method - using only very few elements - illustrates the great potential of anisotropic goal-oriented mesh adaptation.

For the a posteriori anisotropic and averaged isotropic methods, we observe that the QoI approximation accuracy 'saturates' at around 0.5%, for meshes with over 4,000 elements. Uniform meshing requires around 64,000 elements in order to achieve this error threshold. As such, the saturation of the errors resulting from these goal-oriented approaches indicates that the high resolution required by a 64,000 element mesh is being achieved in the important regions under these strategies. In particular, sufficient resolution is used in the region of interest. However, lower resolution is used elsewhere, where it is unnecessary.

The major success is that all goal-oriented metrics illustrate convergence to J_2^{exact} using (often significantly) fewer elements than required on uniform meshes. That is, we demonstrate the capability of our implementation of goal-oriented mesh adaptation to accurately approximate QoIs of the form of J_1 and J_2 for model steady-state advection-diffusion problems.

Having validated the three methods for the centred and offset receiver cases, we argue that they may be utilised in more complex velocity fields in future work.

5.4 Three Dimensional Problems

Whilst serving their purpose as validation experiments, two dimensional approximations are insuffi-

cient for many realistic CFD applications, such as the desalination outfall problem discussed in Subsection 5.3. Figure 10 illustrates the extension of the goal-oriented methodology to such problems.

The offset receiver tracer transport test case examined in Subsection 5.3 is extended to three dimensions in the natural way by considering source location $\mathbf{x}_0 = (2, 5, 5)$, region of interest centred at $(20, 7.5, 7.5)$ and fluid velocity $\mathbf{u} = (1, 0, 0)$.

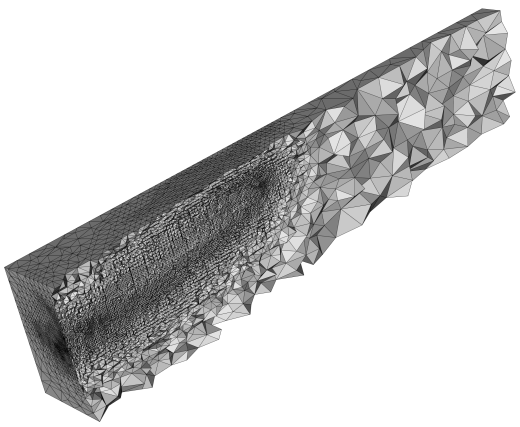


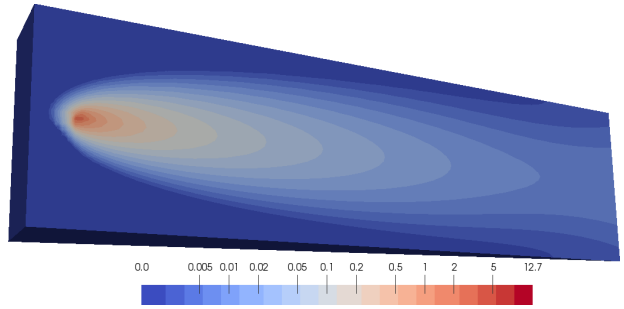
Figure 10: Clip of a mesh with 1,776,396 elements resulting from goal-oriented mesh adaptation for a 3D extension of the TELEMAC-2D test case discussed in Subsection 5.3. The metric is obtained by averaging forward and adjoint a posteriori anisotropic metrics.

The 3D mesh plot shown in Figure 10 exhibits a wide range of scales, from high resolution near to the source and receiver to very coarse resolution downstream of the receiver region. The cell volumes range from 1.25×10^{-10} cubic units to 4.81 cubic units, covering more than *ten cubic orders of magnitude*. That is, there are variations of a factor of as much as 1,000 in each direction. Clearly, using a uniform mesh with cells of volume 1.25×10^{-10} is completely prohibitive.

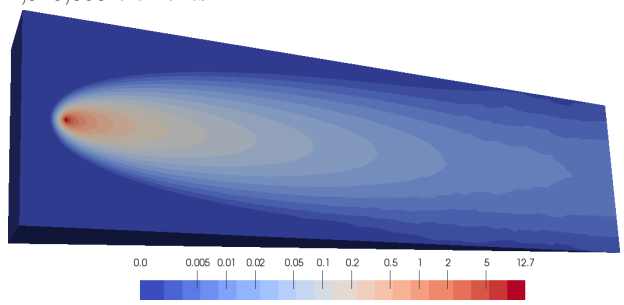
That the goal-oriented adaptation strategy exhibited in Figure 10 is capable of producing multi-scale meshes is essential when moving to large-scale realistic 3D applications, where the region of interest is tiny in comparison to the domain size.

It is also worth noting that, for PDE problems with quantities of interest of the form used in this work, the advantages associated with goal-oriented mesh adaptation improve with spatial dimension. The ratio between the volume/area of the region of interest to the volume/area of the domain decreases when we generalise the 2D problem as described in Subsection 5.3 to the 3D problem considered here.

Consider the slice plots shown in Figure 11, which



(a) Finite element solution on a uniform mesh with 1,920,000 elements.



(b) Finite element solution on a 1,766,396 element mesh which was obtained by adaptation with respect to averaged forward and adjoint a posteriori metrics.

Figure 11: Slices of the solution to the 3D extension of the TELEMAC-2D test case discussed in Subsection 5.3, presented on meshes as indicated. Slices were taken on the plane intersecting source and receiver which is orthogonal to the z-axis.

compare isosurfaces of finite element solutions computed on a uniform mesh and the goal-oriented adaptive mesh shown in Figure 10. The impact of the low downstream resolution is visible in Subfigure 11b, where grid-scale features are apparent in the isosurface contours. In contrast, the contours of the uniform mesh isosurfaces shown in Subfigure 11a are relatively smooth across the entire domain. Given the location of the receiver, the downstream resolution applied in the uniform mesh case is unnecessary, as observed in Subsection 5.3.

Another clear advantage of the goal-oriented mesh is that it is able to accurately capture the point source. This cannot be said for the uniform mesh case, where the area surrounding the source appears to have suffered from numerical diffusion effects. Accurately approximating the source term is essential for capturing the resultant downstream dynamics.

6. CONCLUSION

The main achievement of this work is the successful implementation of three approaches to goal-oriented

mesh adaptation in the finite element package Firedrake. To the best of the authors' knowledge, this is the first comparison of these rather different approaches within a single framework.

An established advection-diffusion test case with a known analytical solution is considered in Section 5. The fact that an analytical solution exists means that the implementation may be validated by showing that a quantity of interest converges to its exact value under each goal-oriented approach. The experiments in Section 5 illustrate that all three goal-oriented strategies enable convergence of two different quantities of interest to their analytical values using (often significantly) fewer elements than would be required for uniform meshes. Whilst no clear conclusions may be drawn concerning the optimality of one particular goal-oriented approach, this work motivates further investigation of each of these strategies within the Firedrake framework.

In a number of instances, it is shown that combining forward and adjoint goal-oriented error metrics, by superposition or averaging, can lead to smaller errors in QoI estimation, for the same number of elements. In particular, this work contains a novel anisotropic metric construction strategy, based on the extension of a known a priori metric to an equivalent metric corresponding to the adjoint equation. To the best of the authors' knowledge, it has not yet been shown that this metric relates to a rigorous error estimator. However, the numerical experiments in Section 5 show that combining this metric with the established a priori metric can result in meshes with relatively few elements upon which the QoI may be accurately approximated. Future work will seek a rigorous error estimate associated with this metric.

The numerical experiments performed in Subsection 5.3 may be considered as simplified interpretations of a desalination outfall modelling scenario, with idealised domains, fluid velocities and diffusivity coefficients. Such idealised problems are useful for validation purposes, due to the existence of analytical solutions.

Moving to more realistic applications comes with a number of additional challenges. Namely, realistic desalination outfall problems are inherently three dimensional, tide-dependent and multi-scale, with the diameters of the inlet and outlet pipes, the scales of key mixing processes, and the size of the ocean domain varying across multiple orders of magnitude. Future work will extend the goal-oriented adaptation approach discussed in this paper to unsteady problems with realistic domains and tidal forcing within the Thetis coastal ocean modelling framework [44] which is built on top of Firedrake. That this work extends to three dimensions is illustrated in Subsection 5.4.

Due to the relatively small sizes of the experiments considered in Subsection 5.3, performance analyses were not considered in this paper. Larger scale test cases should be investigated in future work, wherein such experiments can be made rigorous.

Additionally, we intend to investigate goal-oriented mesh adaptation in the context of more complicated finite element discretisations. One increasingly popular discretisation is the discontinuous Galerkin (DG) space family [23], within which additional inter-element flux terms arise in the formulation of dual weighted residual error estimators.

Acknowledgements

Many thanks to members of Imperial College London's Applied Modelling and Computation Group (AMCG) and to the developers of the Firedrake and Thetis projects for their useful recommendations regarding this work. Further thanks to staff and students of the Mathematics of Planet Earth Centre for Doctoral Training (MPE CDT) for their ongoing support and advice. This work was funded by the Engineering and Physical Sciences Research Council (EPSRC) under grants EP/L016613/1 and EP/R029423/1.

References

[1] Loseille A., Dervieux A., Alauzet F. "Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations." *Journal of computational physics*, vol. 229, no. 8, 2866–2897, 2010

[2] Funke S.W., Kramer S.C., Piggott M.D. "Design optimisation and resource assessment for tidal-stream renewable energy farms using a new continuous turbine approach." *Renewable energy*, vol. 99, 1046–1061, 2016

[3] Davis B.N., LeVeque R.J. "Adjoint methods for guiding adaptive mesh refinement in tsunami modeling." *Global Tsunami Science: Past and Future, Volume I*, pp. 4055–4074. Springer, 2016

[4] Zienkiewicz O.C., Zhu J.Z. "A simple error estimator and adaptive procedure for practical engineering analysis." *International journal for numerical methods in engineering*, vol. 24, no. 2, 337–357, 1987

[5] Loseille A., Alauzet F. "Optimal 3D highly anisotropic mesh adaptation based on the continuous mesh framework." *Proceedings of the 18th International Meshing Roundtable*, pp. 575–594. Springer, 2009

[6] Ainsworth M., Oden J.T. "A posteriori error estimation in finite element analysis." *Computer*

methods in applied mechanics and engineering, vol. 142, no. 1-2, 1–88, 1997

[7] Becker R., Rannacher R. *A feed-back approach to error control in finite element methods: basic analysis and examples.* IWR, 1996

[8] Becker R., Rannacher R. "An optimal control approach to a posteriori error estimation in finite element methods." *Acta numerica*, vol. 10, 1–102, 2001

[9] Venditti D.A., Darmofal D.L. "Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows." *Journal of Computational Physics*, vol. 187, no. 1, 22–46, 2003

[10] Wang Y., Ragusa J.C. "Standard and goal-oriented adaptive mesh refinement applied to radiation transport on 2D unstructured triangular meshes." *Journal of Computational Physics*, vol. 230, no. 3, 763–788, 2011

[11] Loseille A., Alauzet F. "Continuous mesh framework part I: well-posed continuous interpolation error." *SIAM Journal on Numerical Analysis*, vol. 49, no. 1, 38–60, 2011

[12] Power P., Pain C.C., Piggott M., Fang F., Gorman G.J., Umpleby A., Goddard A.J., Navon I. "Adjoint a posteriori error measures for anisotropic mesh optimisation." *Computers & Mathematics with Applications*, vol. 52, no. 8-9, 1213–1242, 2006

[13] Rogé G., Martin L. "Goal-oriented anisotropic grid adaptation." *Comptes Rendus Mathématique*, vol. 346, no. 19-20, 1109–1112, 2008

[14] Carpio J., Prieto J.L., Bermejo R. "Anisotropic goal-oriented mesh adaptivity for elliptic problems." *SIAM Journal on Scientific Computing*, vol. 35, no. 2, A861–A885, 2013

[15] Riadh A., Cedric G., Jean M. "TELEMAC modeling system: 2D hydrodynamics TELEMAC-2D software release 7.0 user manual." *Paris: R&D, Electricite de France*, p. 134, 2014

[16] George P., Hecht F., Vallet M. "Creation of internal points in Voronoi's type method. Control and adaptation." *Adv. Eng. Software*, vol. 13, no. 5-6, 303–312, 1991

[17] Pain C., Umpleby A., De Oliveira C., Goddard A. "Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations." *Computer Methods in Applied Mechanics and Engineering*, vol. 190, no. 29-30, 3771–3796, 2001

[18] Alauzet F. "Size gradation control of anisotropic meshes." *Finite Elements in Analysis and Design*, vol. 46, no. 1-2, 181–202, 2010

[19] Frey P.J., Alauzet F. "Anisotropic mesh adaptation for CFD computations." *Computer methods in applied mechanics and engineering*, vol. 194, no. 48-49, 5068–5082, 2005

[20] Alauzet F., Loseille A. "A decade of progress on anisotropic mesh adaptation for computational fluid dynamics." *Computer-Aided Design*, vol. 72, 13–39, 2016

[21] Barral N. *Time-accurate anisotropic mesh adaptation for three-dimensional moving mesh problems.* Ph.D. thesis, Université Pierre et Marie Curie, 2015

[22] Alauzet F., Olivier G. "An Lp-L∞ space-time anisotropic mesh adaptation strategy for time dependent problems." *Proceedings of ECCOMAS CFD*. 2010

[23] Reed W.H., Hill T. "Triangular mesh methods for the neutron transport equation." Tech. rep., Los Alamos Scientific Lab., N. Mex.(USA), 1973

[24] Alauzet F., Belme A., Dervieux A. "Anisotropic goal-oriented mesh adaptation for time dependent problems." *Proceedings of the 20th international meshing roundtable*, pp. 99–121. Springer, 2011

[25] Belme A., Dervieux A., Alauzet F. "Time accurate anisotropic goal-oriented mesh adaptation for unsteady flows." *Journal of Computational Physics*, vol. 231, no. 19, 6323–6348, 2012

[26] Rathgeber F., Ham D.A., Mitchell L., Lange M., Luporini F., McRae A.T., Bercea G.T., Markall G.R., Kelly P.H. "Firedrake: automating the finite element method by composing abstractions." *ACM Transactions on Mathematical Software (TOMS)*, vol. 43, no. 3, 24, 2016. URL `http://arxiv.org/abs/1411.2940`

[27] Alnæs M.S., Logg A., Ølgaard K.B., Rognes M.E., Wells G.N. "Unified form language: A domain-specific language for weak formulations of partial differential equations." *ACM Transactions on Mathematical Software (TOMS)*, vol. 40, no. 2, 9, 2014

[28] Balay S., Abhyankar S., Adams M.F., Brown J., Brune P., Buschelman K., Dalcin L., Dener A., Eijkhout V., Gropp W.D., Kaushik D., Knepley M.G., May D.A., McInnes L.C., Mills R.T., Munson T., Rupp K., Sanan P., Smith B.F., Zampini

S., Zhang H., Zhang H. "PETSc Users Manual." Tech. Rep. ANL-95/11 - Revision 3.11, Argonne National Laboratory, 2019. URL http://www.mcs.anl.gov/petsc

[29] Balay S., Gropp W.D., McInnes L.C., Smith B.F. "Efficient Management of Parallelism in Object Oriented Numerical Software Libraries." E. Arge, A.M. Bruaset, H.P. Langtangen, editors, *Modern Software Tools in Scientific Computing*, pp. 163–202. Birkhäuser Press, 1997

[30] Rokos G., Gorman G. "PRAgMaTIc–parallel anisotropic adaptive mesh toolkit." *Facing the Multicore-Challenge III*, pp. 143–144. Springer, 2013

[31] Ibanez D., Barral N., Krakos J., Loseille A., Michal T., Park M. "First Benchmark of the Unstructured Grid Adaptation Working Group." *Procedia engineering*, vol. 203, 154–166, 2017

[32] Park M.A., Barral N., Ibanez D., Kamenetskiy D.S., Krakos J.A., Michal T.R., Loseille A. "Unstructured Grid Adaptation and Solver Technology for Turbulent Flows." *2018 AIAA Aerospace Sciences Meeting*, p. 1103. 2018

[33] Park M.A., Balan A., Anderson W.K., Galbraith M.C., Caplan P., Carson H.A., Michal T.R., Krakos J.A., Kamenetskiy D.S., Loseille A., et al. "Verification of Unstructured Grid Adaptation Components." *AIAA Scitech 2019 Forum*, p. 1723. 2019

[34] Barral N., Knepley M.G., Lange M., Piggott M.D., Gorman G.J. "Anisotropic mesh adaptation in Firedrake with PETSc DMPlex." *arXiv preprint arXiv:1610.09874*, 2016

[35] Dalcin L.D., Paz R.R., Kler P.A., Cosimo A. "Parallel distributed computing using Python." *Advances in Water Resources*, vol. 34, no. 9, 1124–1139, 2011. New Computational Methods and Software Tools

[36] Farrell P., Piggott M., Pain C., Gorman G., Wilson C. "Conservative interpolation between unstructured meshes via supermesh construction." *Computer Methods in Applied Mechanics and Engineering*, vol. 198, no. 33, 2632 – 2642, 2009

[37] Farrell P.E., Maddison J.R. "Conservative interpolation between volume meshes by local Galerkin projection." *Computer Methods in Applied Mechanics and Engineering*, vol. 200, no. 1-4, 89–100, 2011

[38] Maddison J.R., Farrell P.E., Panourglas I.P. "Parallel supermeshing for multimesh modelling." Tech. Rep. eCSE03-08, ARCHER, May 2016. URL https://www.archer.co.uk/community/eCSE/eCSE03-08/eCSE03-08-TechnicalReport.pdf

[39] Farrell P.E., Ham D.A., Funke S.W., Rognes M.E. "Automated derivation of the adjoint of high-level transient finite element programs." *SIAM Journal on Scientific Computing*, vol. 35, no. 4, C369–C393, 2013

[40] Mitusch S.K. *An Algorithmic Differentiation Tool for FEniCS*. Master's thesis, University of Oslo, 2018

[41] "Software used in 'Anisotropic Goal Oriented Mesh Adaptation in Firedrake'.", June 2019. URL https://doi.org/10.5281/zenodo.3250888

[42] "Simulation code used in 'Anisotropic Goal Oriented Mesh Adaptation in Firedrake'.", Aug. 2019. URL https://doi.org/10.5281/zenodo.3358565

[43] Donea J., Huerta A. *Finite element methods for flow problems*. John Wiley & Sons, 2003

[44] Kärnä T., Kramer S.C., Mitchell L., Ham D.A., Piggott M.D., Baptista A.M. "Thetis coastal ocean model: discontinuous Galerkin discretization for the three-dimensional hydrostatic equations." *Geoscientific Model Development*, vol. 11, no. 11, 4359–4382, 2018

[45] Hendrickson B., Leland R. "A multilevel algorithm for partitioning graphs." *Supercomputing '95: Proceedings of the 1995 ACM/IEEE Conference on Supercomputing (CDROM)*, p. 28. ACM Press, New York, 1995

[46] Amestoy P.R., Duff I.S., L'Excellent J.Y., Koster J. "A fully asynchronous multifrontal solver using distributed dynamic scheduling." *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 1, 15–41, 2001

[47] Amestoy P.R., Guermouche A., L'Excellent J.Y., Pralet S. "Hybrid scheduling for the parallel solution of linear systems." *Parallel Computing*, vol. 32, no. 2, 136–156, 2006

[48] Mitchell L., Müller E.H. "High level implementation of geometric multigrid solvers for finite element problems: applications in atmospheric modelling." *Journal of Computational Physics*, vol. 327, 1–18, 2016. URL http://arxiv.org/abs/1605.00492