Forward-in-Time Goal-Oriented Adaptivity

Judit Muñoz-Matute[a], David Pardo[a,b,c], Victor M. Calo[d,e], Elisabete Alberdi[a]

[a] *University of the Basque Country (UPV/EHU), Leioa, Spain*
[b] *BCAM-Basque Center for Applied Mathematics, Bilbao, Spain*
[c] *IKERBASQUE, Basque Foundation for Science, Bilbao, Spain*
[d] *Applied Geology, Western Australian School of Mines, Faculty of Science and Engineering, Curtin University, Perth, WA, Australia 6845*
[e] *Mineral Resources, Commonwealth Scientific and Industrial Research Organisation (CSIRO), Kensington, WA, Australia 6152*

Abstract

In goal-oriented adaptive algorithms for partial differential equations, we adapt the finite element mesh in order to reduce the error of the solution in some quantity of interest. In time-dependent problems, this adaptive algorithm involves solving a dual problem that runs backward in time. This process is, in general, computationally expensive in terms of memory storage. In this work, we define a pseudo-dual problem that runs forward in time. We also describe a forward-in-time adaptive algorithm that works for some specific problems. Although it is not possible to define a general dual problem running forwards in time that provides information about future states, we provide numerical evidence via one-dimensional problems in space to illustrate the efficiency of our algorithm as well as its limitations. Finally, we propose a hybrid algorithm that employs the classical backward-in-time dual problem once and then performs the adaptive process forwards in time.

Keywords: linear advection-diffusion equation, goal-oriented adaptivity, pseudo-dual problem, error representation, Finite Element Method

1. Introduction

Goal-oriented adaptive algorithms are widely employed in engineering [15–17, 20, 24]. This technology seeks to accurately approximate relevant solution features of partial differential equations (PDEs), called quantities of interest. We often quantify these features as functionals of the solution. The main objective of goal-oriented adaptivity is to reduce the error in the quantity of interest [1, 14, 18]. The general process is as follows: we define an auxiliary dual problem, with the output functional as the source. Then, we represent the error in the quantity of interest as an integral over the whole domain employing the errors of both primal and dual problems. Finally, we obtain an upper bound of the error in the quantity of interest in terms of local element contributions that drive the adaptive process.

To perform goal-oriented adaptivity in time, we need a full space-time variational formulation of the problem [9]. From this formulation, we can represent the error in the quantity of interest as an integral over the whole space-time domain employing the errors of the primal and dual problems. In this case, the dual problem is also a time-dependent PDE but running backwards in time. Most authors [5, 10, 22] select discontinuous-in-time test functions to discretize both primal and dual problems. This selection decouples the resulting systems and we can solve them as time-marching schemes [2, 21, 25]. However, as the dual problem runs backward in time, the goal-oriented adaptive process involves solving two problems running in opposite directions in time. This fact implies that we need to store all the information coming from the primal and dual problems in order to estimate the error contributions and perform the adaptivity.

We can find different strategies for time-domain goal-oriented adaptivity in the literature: in [3] for the wave equation, in [9, 22] for parabolic problems and in [23, 26] for nonlinear problems. The aforementioned work employs implicit time marching-schemes to solve the primal and dual problems as it is known that they have variational structure. Recently, the authors in [6] derived a goal-oriented a posteriori error estimation for Implicit-Explicit schemes employing quadrature rules. Finally, in [13] we proved that explicit Runge-Kutta methods can be reinterpreted as Galerkin methods and from this work, we derived in [12] an explicit-in-time goal-oriented adaptive algorithm for linear parabolic problems.

In this work, following the ideas presented in [7, 8, 11], we define a pseudo-dual problem that, as the primal problem, runs forward in time. Then, we define a forward-in-time goal-oriented adaptive process. In the proposed algorithm, we solve the primal and dual problems at a fixed time step to adapt the corresponding spatial mesh before we move to the next time step. This is possible because both primal and dual problems run in the same direction in time. This approach overcomes the memory storage issue of the classical algorithm. We can perform the adaptivity as we calculate the solution, thus we do not need to store the solution at every time step for each iteration. However, this algorithm only works in some specific problem configurations. For example, in diffusion problems where the quantity of interest is localized in some area of the spatial domain and has support in the whole time interval. For advection-diffusion problems and for problems where the quantity of interest is placed in a reduced late time interval, the pseudo-dual problem we define is not generally well defined. In general, it is not possible to define a dual problem running forward in time that captures the necessary information from the future time steps. Nevertheless, the method works properly in several relevant instances and could be useful in some engineering problems when some features of the solution are known beforehand. As an alternative for advection-diffusion problems, we also propose a hybrid algorithm that solves the classical dual problem backwards in time once. Then, we set the initial condition of the pseudo-dual problem as the initial condition of the classical dual problem. Finally, the adaptive process is performed forwards in time. In both algorithms proposed herein, we only need to store the solutions of the primal and dual problems at one time step to perform the adaptivity.

The paper is organized as follows: Section 2 shows the strong and weak formulations and the discretization of the linear advection-diffusion equation. Section 3 introduces the classical dual problem and the classical error representation. Section 4 defines the proposed pseudo-dual problem. Section 5 explains the classical goal-oriented adaptive algorithm in space for a fixed time grid, the proposed forward-in-time goal-oriented adaptive process, and the hybrid algorithm. Section 6 presents the numerical results showing in which situations the algorithm performs properly and also when performs poorly. Finally, Section 7 summarizes the conclusions.

2. Primal problem

In this section we introduce the strong and weak formulations, and the discretization of the primal problem.

2.1. Strong formulation

Let Ω an open bounded subset of $\mathbb{R}^d$ with $d = \{1, 2, 3\}$ and $I = (0, T] \subset \mathbb{R}$, we consider the *linear advection-diffusion equation*

$$\begin{cases} u_t - \nabla \cdot (\kappa \nabla u) + a \cdot \nabla u = f & \text{in } \Omega \times I, \\ u = 0 & \text{in } \partial\Omega \times I, \\ u(0) = u_0 & \text{in } \Omega, \end{cases} \tag{1}$$

where $\partial\Omega$ denotes the boundary of Ω and the solution $u(\mathbf{x}, t)$ represents the temperature distribution in a body.

The source term $f(\mathbf{x}, t)$, the initial condition $u_0(\mathbf{x})$, the diffusivity tensor $\kappa(\mathbf{x})$ and the velocity field $a(\mathbf{x})$ are known data. We assume that $a(\mathbf{x})$ is a bounded divergence-free vector field, $\nabla \cdot a = 0$, and $\kappa(\mathbf{x})$ is a bounded above and strictly positive symmetric, second order tensor.

2.2. Weak formulation

We derive a space-time formulation of problem (1) [3, 22]. We consider the Hilbert space

$$V := H_0^1(\Omega) = \{u \in H^1(\Omega) \mid u = 0 \text{ on } \partial\Omega\},$$

and its dual $V' = H^{-1}(\Omega)$. We introduce the test space $\mathcal{V} := L^2(I; V)$, which is the Bochner space of all integrable functions in time taking values in V, and we denote its dual space as $\mathcal{V}' := L^2(I; V')$.

We need for the solution to be in $\mathcal{V}$ and also $u_t \in \mathcal{V}'$, so we consider the trial space

$$\mathcal{U} := \{u \in \mathcal{V} \mid u_t \in \mathcal{V}'\},$$

which is continuously embedded in $C(\bar{I}; L^2(\Omega))$, therefore, all functions in $\mathcal{U}$ are globally continuous in time.

Now, we multiply the equation in (1) by the test functions $v \in \mathcal{V}$, we integrate over the space-time domain $\Omega \times I$ and we impose the initial condition in weak form. Then, the weak formulation of (1) is:

Find $u \in \mathcal{U}$ such that

$$\begin{aligned} \int_I \langle u_t, v\rangle \, dt + \int_I (\kappa \nabla u, \nabla v) \, dt + \int_I (a \cdot \nabla u, v) \, dt = \int_I \langle f, v\rangle \, dt, \ \forall v \in \mathcal{V}, \\ (u(0), w) = (u_0, w), \ \forall w \in L^2(\Omega), \end{aligned} \tag{2}$$

where we assume that $f \in \mathcal{V}'$ and $u_0 \in L^2(\Omega)$.

Here, $u(0) := u(\mathbf{x}, 0)$, $\langle \cdot, \cdot \rangle$ denotes the duality pairing between the spaces V and V', $(\cdot, \cdot)$ is the inner product in $L^2(\Omega)$ and we

$$B(u, v) := \int_I \langle u_t, v\rangle \, dt + \int_I (\kappa \nabla u, \nabla v) \, dt + \int_I (a \cdot \nabla u, v) \, dt. \tag{3}$$

2.3. Discretization

For the discretization of problem (2), we employ a *discontinuous Galerkin (DG) method* in time and a *finite element (FE) method* in space.

First, we define a partition of the time interval $\bar{I} = [0, T]$ as

$$0 = t_0 < t_1 < \ldots < t_{m-1} < t_m = T, \tag{4}$$

and we denote by $I_k = (t_{k-1}, t_k]$, $\tau_k := t_k - t_{k-1}$, $\forall k = 1, \ldots, m$ and $\tau := \max_{1 \le k \le m} \tau_k$.

In order to fully discretize problem (2), we select a finite element space $V_h^k \subset V$ associated to each time step t_k, $\forall k = 0, \ldots, m$ and we select the following discrete space

$$\mathcal{V}_{\tau h} := \{v_{\tau h} \in L^2(I;V) \mid v_{\tau h_{|I_k}} \in P_r(I_k; V_h^k),\ \forall k = 1, \ldots, m,\ v_{\tau h}(0) \in V_h^0\} \subset \mathcal{V}.$$

where $P_r(I_k; V_h^k)$ is the space of all polynomials with degree less or equal than r on the interval I_k taking values in V_h^k.

The functions in $\mathcal{V}_{\tau h}$ could be discontinuous at each time step t_k, so we define the jump of the function v at t_k as $[[v]]^k := v(t_k^+) - v(t_k^-)$, where

$$v(t_k^+) := \lim_{s \longrightarrow 0^+} v(t_k + s),\ \ v(t_k^-) := \lim_{s \longrightarrow 0^+} v(t_k - s).$$

The authors in, [19, 22], state that a Discontinuous Galerkin formulation of (2) is

$$\left\{ \begin{array}{l} \text{Find } u_{\tau h} \in \mathcal{V}_{\tau h} \text{ such that} \\ B_{DG}(u_{\tau h}, v_{\tau h}) = F(v_{\tau h}) \ \ \forall v_{\tau h} \in \mathcal{V}_{\tau h}, \end{array} \right. \tag{5}$$

where

$$\begin{aligned} B_{DG}(u,v) :&= \sum_{k=1}^{m} \int_{I_k} \Big(\langle u_t, v \rangle + (\kappa \nabla u, \nabla v) + (a \cdot \nabla u, v) \Big) dt \\ &+ \sum_{k=1}^{m} \Big([[u]]^{k-1}, v(t_{k-1}^+) \Big) + (u(0^-), v(0^-)), \\ F(v) :&= \sum_{k=1}^{m} \int_{I_k} \langle f, v \rangle \, dt + (u_0, v(0^-)). \end{aligned}$$

We can express

$$B_{DG}(u,v) = \sum_{k=1}^{m} B^k(u,v) + \sum_{k=1}^{m} \Big([[u]]^{k-1}, v(t_{k-1}^+) \Big) + (u(0^-), v(0^-)), \tag{6}$$

where $B^k(\cdot,\cdot)$ is the restriction of the bilinear form $B(\cdot,\cdot)$ to I_k.

As the test functions in (5) are discontinuous-in-time, we can split the statement into m local-in-time problems and solve it using a time-marching scheme. From [3] we know that when $r = 0$, the resulting scheme obtained from (5) is algebraically equivalent to the backward Euler method in time.

3. Dual problem and error representation

In this section, we define the dual problem we employ to derive the classical error representation.

3.1. Output functional and dual problem

We consider a quantity of interest given by a linear functional $L : \mathcal{U} \subset \mathcal{V} \longrightarrow \mathbb{R}$, the *output functional* which quantifies a relevant feature of the solution.

In this paper, we consider linear output functionals of the form

$$L(v) = \int_I \langle g, v \rangle \, dt + (z_T, v(T)),$$

where $g \in \mathcal{V}'$ and $z_T \in L^2(\Omega)$ are given functions.

In order to represent the error in the quantity of interest, we introduce the following *dual problem*

$$\begin{aligned} -\int_I \langle z_t, v \rangle \, dt + \int_I (\kappa \nabla z, \nabla v) \, dt - \int_I (a \cdot \nabla z, v) \, dt = \int_I \langle g, v \rangle \, dt, \ \forall v \in \mathcal{V}, \\ (z(T), w) = (z_T, w), \ \forall w \in L^2(\Omega). \end{aligned} \tag{7}$$

where $B^*(\cdot, \cdot)$ is the adjoint operator of the form $B(\cdot, \cdot)$

$$B^*(z, v) := -\int_I \langle z_t, v \rangle \, dt + \int_I (\kappa \nabla z, \nabla v) dt - \int_I (a \cdot \nabla z, v) dt.$$

For this particular case, the corresponding strong formulation of (7) is

$$\begin{cases} -z_t - \nabla \cdot (\kappa \nabla z) - a \cdot \nabla z = g & \text{in} \quad \Omega \times I, \\ z = 0 & \text{in} \quad \partial\Omega \times I, \\ z(T) = z_T & \text{in} \quad \Omega, \end{cases} \tag{8}$$

and we conclude that the dual problem runs backwards in time.

3.2. Discretization of the dual problem

The discontinuous Galerkin formulation of problem (7) is

$$\begin{cases} \text{Find } z_{\tau h} \in \mathcal{V}_{\tau h} \text{ such that} \\ B^*_{DG}(z_{\tau h}, v_{\tau h}) = L(v_{\tau h}) \ \ \forall v_{\tau h} \in \mathcal{V}_{\tau h}, \end{cases} \tag{9}$$

where $B^*_{DG}(\cdot,\cdot)$ is the resulting bilinear form after integrating by parts in time and the advection term in space the form $B_{DG}(\cdot,\cdot)$

$$B^*_{DG}(z,v) := \sum_{k=1}^{m} \int_{I_k} \Big(-\langle z_t, v\rangle + (\kappa\nabla z, \nabla v) - (a\cdot\nabla z, v)\Big) dt \\ - \sum_{k=0}^{m-1} \Big([[z]]^k, v(t_k^-)\Big) + (z(T^-), v(T^-)).$$

In the same way as the primal problem, we solve the dual problem (7) employing time-marching scheme but running backwards in time.

3.3. Error representation

First, as $\mathcal{V}_{\tau h} \not\subset \mathcal{U}$ because functions in $\mathcal{V}_{\tau h}$ are discontinuous in time and functions in $\mathcal{U}$ are globally continuous, we define the following space

$$\mathcal{V}_\tau = \{v \in \mathcal{V} \mid v \in L^2(I_k; V),\ v_t \in L^2(I_k; V'),\ \forall k = 1, \dots, m\},$$

which is the minimum subspace of $\mathcal{V}$ containing both $\mathcal{V}_{\tau h}$ and $\mathcal{U}$.

Now, problem (2) holds for any subspace of $\mathcal{V}$. In particular, we select $\mathcal{V}_\tau$ and adding both equations in (2), we obtain

$$B(u, v_\tau) + (u(0), v_\tau(0)) = F(v_\tau)\ \forall v_\tau \in \mathcal{V}_\tau.$$

Since the solutions of (2) and (7) are continuous in time, the jump terms are zero, so they also satisfy problems (5) and (9), respectively. Therefore, we consider the following continuous and discrete primal problems

$$\begin{cases} \text{Find } u \in \mathcal{U} \text{ and } u_{\tau h} \in \mathcal{V}_{\tau h} \text{ such that} \\ B_{DG}(u, v_\tau) = F(v_\tau)\ \ \forall v_\tau \in \mathcal{V}_\tau \subset \mathcal{V}, \\ B_{DG}(u_{\tau h}, v_{\tau h}) = F(v_{\tau h})\ \ \forall v_{\tau h} \in \mathcal{V}_{\tau h}, \end{cases} \tag{10}$$

and the continuous dual problem

$$\begin{cases} \text{Find } z \in \mathcal{U} \text{ such that} \\ B^*_{DG}(z, v_\tau) = L(v_\tau)\ \ \forall v_\tau \in \mathcal{V}_\tau \subset \mathcal{V}. \end{cases} \tag{11}$$

We define the error of the primal problem as $e = u - u_{\tau h} \in \mathcal{V}_\tau \subset \mathcal{V}$. Now, substituting u by e in (11) and integrating by parts in time and in space the advection term, we obtain the *classical error representation*

$$L(e) = B^*_{DG}(z, e) = B_{DG}(e, z), \tag{12}$$

which represents the error in the quantity of interest as an integral over the whole space-time domain $\Omega \times I$.

4. Forward-in-time pseudo-dual problem

The existing goal-oriented adaptive strategies based on time-marching schemes are computationally expensive because we need to solve the primal problem forward in time, while the dual problem is solved backwards in time.

To reduce the computational cost and implementation complexity, we introduce an alternative dual problem, which we denote pseudo-dual problem that, as the primal problem, runs forward in time.

We consider the following pseudo-dual problem

$$\left\{ \begin{array}{l} \text{Find } \tilde{z} \in \mathcal{U} \text{ such that} \\ \tilde{B}(\tilde{z}, v_\tau) = L(v_\tau) \ \forall v_\tau \in \mathcal{V}_\tau, \end{array} \right. \tag{13}$$

where $\tilde{B}(\cdot,\cdot)$ is an alternative bilinear form and since $e \in \mathcal{V}_\tau$, we obtain the following error representation,

$$L(e) = \tilde{B}(\tilde{z}, e). \tag{14}$$

For example, if we select $\tilde{B}(\tilde{z}, v) = B_{DG}(\tilde{z}, v)$, as $\tilde{z}$ is a continuous function in time, we have that

$$B(\tilde{z}, v_\tau) + (\tilde{z}(0), v_\tau(0)) = \int_I \langle g, v_\tau \rangle \, dt + (z_T, v_\tau(T)), \ \forall v_\tau \in \mathcal{V}_\tau.$$

Now, as explained in [4], we can express the right-hand-side of the previous equation as

$$B(\tilde{z}, v_\tau) + (\tilde{z}(0), v_\tau(0)) = \int_I \langle g, v_\tau \rangle \, dt + \int_I (\delta(t-T) z_T, v_\tau) \, dt, \ \forall v_\tau \in \mathcal{V}_\tau, \tag{15}$$

where $\delta(t-T)$ is a Dirac delta distribution. Therefore, following the same argument as in Section 3, we conclude that the initial condition of the pseudo-dual problem is zero and both functions g and z_T are part of the source.

We can solve problem (13) forward in time using a time-marching scheme. However, as we show in the numerical results, the pseudo-dual problem is only properly defined for particular problems.

5. Goal-oriented adaptivity in space

In this section, we first describe the classical goal-oriented adaptivity in space for a fixed time mesh [23]. Then we propose an alternative strategy based on the pseudo-dual problem (13).

In the error representation (12), we need the exact solutions u and z of the primal and dual problems, respectively. Since they are unavailable, we approximate them numerically by enriching the subspace, $\mathcal{V}_{\tau h}$ and selecting $u \sim u_{\tau \frac{h}{2}} \in \mathcal{V}_{\tau \frac{h}{2}}$, $z \sim z_{\tau \frac{h}{2}} \in \mathcal{V}_{\tau \frac{h}{2}}$.

We define the following error

$$e_{\tau \frac{h}{2}} := u_{\tau \frac{h}{2}} - u_{\tau h},$$

and we approximate the exact error as

$$e = u - u_{\tau h} \sim u_{\tau \frac{h}{2}} - u_{\tau h} = e_{\tau \frac{h}{2}}.$$

We focus on reducing the error in the quantity of interest coming from the spatial discretization, i.e., $L(e_{\tau \frac{h}{2}})$.

5.1. *Classical goal-oriented adaptive algorithm*

The classical goal-oriented adaptive strategy in space is based on the error representation (12)

$$L(e_{\tau \frac{h}{2}}) = B_{DG}\left(e_{\tau \frac{h}{2}}, z_{\tau \frac{h}{2}}\right),$$

or equivalently from (6)

$$\begin{aligned} L(e_{\tau \frac{h}{2}}) = \sum_{k=1}^{m} \sum_{i=1}^{n_k} B_{\Omega_i^k}(e_{\tau \frac{h}{2}}, z_{\tau \frac{h}{2}}) + \left([[e_{\tau \frac{h}{2}}]]^{k-1}, z_{\tau \frac{h}{2}}(t_{k-1}^{+})\right)_{\Omega_i^k} + \\ + \sum_{i=1}^{n_0} \left(e_{\tau \frac{h}{2}}(0^-), z_{\tau \frac{h}{2}}(0^-)\right)_{\Omega_i^0}, \end{aligned} \tag{16}$$

where $\{\Omega_i^k\}_{i=1,\dots,n_k}$, $\forall k = 0, \dots, m$ is a partition of the spatial domain Ω at $t = t_k$. Here, $(\cdot,\cdot)_{\Omega_i^k}$ and $B_{\Omega_i^k}$ denote the restrictions of $(\cdot,\cdot)$ and $B^k(\cdot,\cdot)$ to each element Ω_i^k, respectively.

Applying the triangle inequality in (16), we obtain an upper bound of the error in the quantity of interest

$$\begin{aligned} |L(e_{\tau \frac{h}{2}})| \leq \sum_{k=1}^{m} \sum_{i=1}^{n_k} \left| B_{\Omega_i^k}(e_{\tau \frac{h}{2}}, z_{\tau \frac{h}{2}}) + \left([[e_{\tau \frac{h}{2}}]]^{k-1}, z_{\tau \frac{h}{2}}(t_{k-1}^{+})\right)_{\Omega_i^k} \right| + \\ + \sum_{i=1}^{n_0} \left| \left(e_{\tau \frac{h}{2}}(0^-), z_{\tau \frac{h}{2}}(0^-)\right)_{\Omega_i^0} \right|, \end{aligned} \tag{17}$$

which may guide the goal-oriented adaptive process.

Now, we define the error estimator of each time step as

$$\begin{aligned} Est_0 &:= \left| \left(e_{\tau\frac{h}{2}}(0^-), z_{\tau\frac{h}{2}}(0^-) \right) \right|, \\ Est_k &:= \left| B^k(e_{\tau\frac{h}{2}}, z_{\tau\frac{h}{2}}) + \left([[e_{\tau\frac{h}{2}}]]^{k-1}, z_{\tau\frac{h}{2}}(t_{k-1}^+) \right) \right|, \end{aligned} \tag{18}$$

$\forall k = 1, \ldots, m$, and the error estimator of each spatial element Ω_i^k as

$$\begin{aligned} \eta_i^0 &:= \left| \left(e_{\tau\frac{h}{2}}(0^-), z_{\tau\frac{h}{2}}(0^-) \right)_{\Omega_i^0} \right|, \\ \eta_i^k &:= \left| B_{\Omega_i^k}(e_{\tau\frac{h}{2}}, z_{\tau\frac{h}{2}}) + \left([[e_{\tau\frac{h}{2}}]]^{k-1}, z_{\tau\frac{h}{2}}(t_{k-1}^+) \right)_{\Omega_i^k} \right|, \end{aligned} \tag{19}$$

$\forall i = 1, \ldots, n_k$, and $\forall k = 1, \ldots, m$.

Finally, Figure 1 illustrates a classical process to perform goal-oriented adaptivity in space for a fixed time grid. The input arguments of the classical algorithm are the time grid $\{\tau_k\}_{k=1,\ldots,m}$, the spatial mesh at each time step $\{\mathcal{M}_h^k\}_{k=0,\ldots,m}$, a tolerance tol_1 and two parameters $\theta, \lambda \in [0,1]$. We first calculate the primal solutions $u_{\tau h}$ and $u_{\tau\frac{h}{2}}$ forward in time. Then, we compute the dual solution $z_{\tau\frac{h}{2}}$ and the estimators Est_k backward in time. For all spatial meshes satisfying $Est_k \geq \theta \cdot \max\limits_{0\leq k\leq m} Est_k$, we refine those elements in space that satisfy $\eta_i^k \geq \lambda \cdot \max\limits_{1\leq i\leq n_k} \eta_i^k$. The process ends when the relative error $100 \cdot |L(e_{\tau\frac{h}{2}})|/|L(u_{\tau\frac{h}{2}})|$ is lower than the fixed tolerance tol_1.

5.2. Forward-in-time goal-oriented adaptive algorithm

We now describe a goal-oriented adaptive strategy based on the pseudo-dual problem (13) when $\tilde{B}(\tilde{z}, v) = B_{DG}(\tilde{z}, v)$ that runs forwards is time.

From (14), we have

$$L(e_{\tau\frac{h}{2}}) = B_{DG}(\tilde{z}_{\tau\frac{h}{2}}, e_{\tau\frac{h}{2}}),$$

and obtain an upper bound of the error in the quantity of interest

$$\begin{aligned} |L(e_{\tau\frac{h}{2}})| \leq & \sum_{k=1}^{m} \sum_{i=1}^{n_k} \left| B_{\Omega_i^k}(\tilde{z}_{\tau\frac{h}{2}}, e_{\tau\frac{h}{2}}) + \left([[\tilde{z}_{\tau\frac{h}{2}}]]^{k-1}, e_{\tau\frac{h}{2}}(t_{k-1}^+) \right)_{\Omega_i^k} \right| + \\ & + \sum_{i=1}^{n_0} \left| \left(\tilde{z}_{\tau\frac{h}{2}}(0^-), e_{\tau\frac{h}{2}}(0^-) \right)_{\Omega_i^0} \right|. \end{aligned} \tag{20}$$

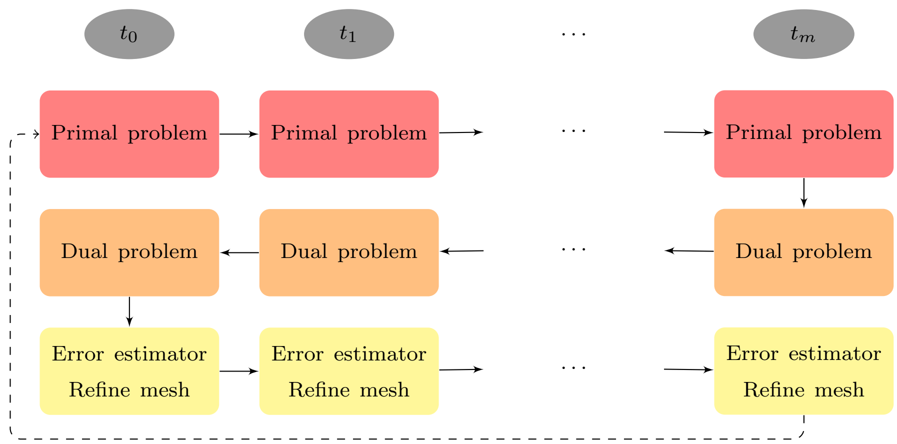


Figure 1: Classical goal-oriented adaptive algorithm.

We construct the estimators following a similar technique to the one we describe for (18) and (19). The algorithm we propose describes a forward-in-time goal-oriented adaptivity process for the spatial error for a fixed time grid. As before, the input arguments are the time grid $\{\tau_k\}_{k=1,\ldots,m}$, the spatial mesh at each time step $\{\mathcal{M}_h^k\}_{k=0,\ldots,m}$, a tolerance tol_2, a parameter $\lambda \in [0,1]$ and the maximum number of iterations per time step. We calculate the solutions $u_{\tau h}$, $u_{\tau \frac{h}{2}}$, $z_{\tau \frac{h}{2}}$ and the estimators Est_k at each time step. Then, if Est_k is greater than a tolerance tol_2, we refine the elements in space satisfying $\eta_i^k \geq \lambda \cdot \max\limits_{1 \leq i \leq n_k} \eta_i^k$. Figure 2 illustrates the proposed adaptive algorithm.

Although the computational time in the proposed forward-in-time adaptive algorithm is similar to the classical one, there is a considerable saving in memory storage. In this algorithm, we do not need to store the solutions of the primal and dual problems at *all* times to perform the adaptive process. The adaptivity is performed as the solutions are calculated. However, as we show in the numerical results, the pseudo dual problem we proposed in (13) is inadequate to solve certain problems. In those situations, we propose a hybrid algorithm in the next section.

5.3. Hybrid algorithm

In order to overcome the difficulties of the forward-in-time adaptive algorithm to properly refine the mesh in some problems, we propose a hybrid

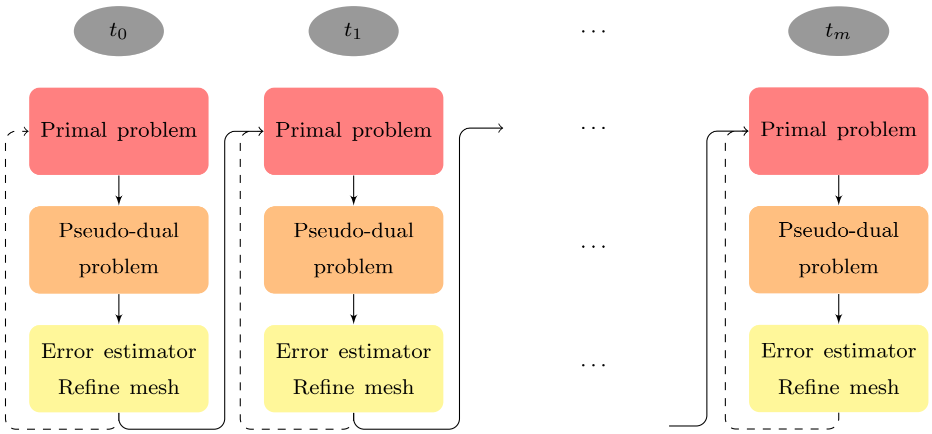


Figure 2: Proposed forward-in-time goal-oriented adaptive algorithm.

algorithm. We select the pseudo-dual problem (13) with

$$\tilde{B}(\tilde{z}, v_\tau) = B_{DG}(\tilde{z}, v_\tau) - (z_{\tau \frac{h}{2}}(0), v_\tau(0)), \tag{21}$$

where $z_{\tau \frac{h}{2}}(0)$ is the approximated solution of the classical dual problem (7) at $t = 0$. Selecting (21), formula (15) becomes

$$B(\tilde{z}, v_\tau) + (\tilde{z}(0), v_\tau(0)) = \int_I \langle g, v_\tau \rangle \, dt + \int_I (\delta(t-T) z_T, v_\tau) \, dt + (z_{\tau \frac{h}{2}}(0), v_\tau(0)),$$

and therefore, the initial condition of the pseudo-dual problem is $z_{\tau \frac{h}{2}}(0)$.

Now, from (21) we have that

$$\begin{aligned} |L(e_{\tau \frac{h}{2}})| \leq \sum_{k=1}^{m} \sum_{i=1}^{n_k} \left| B_{\Omega_i^k}(\tilde{z}_{\tau \frac{h}{2}}, e_{\tau \frac{h}{2}}) + \left([[\tilde{z}_{\tau \frac{h}{2}}]]^{k-1}, e_{\tau \frac{h}{2}}(t_{k-1}^+) \right)_{\Omega_i^k} \right| + \\ + \sum_{i=1}^{n_0} \left| \left(\tilde{z}_{\tau \frac{h}{2}}(0^-), e_{\tau \frac{h}{2}}(0^-) \right)_{\Omega_i^0} - \left(z_{\tau \frac{h}{2}}(0^-), e_{\tau \frac{h}{2}}(0^-) \right)_{\Omega_i^0} \right|. \end{aligned} \tag{22}$$

However, $\tilde{z}_{\tau \frac{h}{2}}(0^-) = z_{\tau \frac{h}{2}}(0^-)$ from the definition of (21), so the last term of (22) is zero and we do not have an error indicator for the first mesh.

Therefore, we define the following upper bound of the error in the quantity of interest that is larger than (22) but includes an estimator for the first mesh in time as in Sections 5.1 and 5.2

$$\begin{aligned}|L(e_{\tau\frac{h}{2}})| \leq \sum_{k=1}^{m}\sum_{i=1}^{n_k}\left| B_{\Omega_i^k}(\tilde{z}_{\tau\frac{h}{2}}, e_{\tau\frac{h}{2}}) + \left([[\tilde{z}_{\tau\frac{h}{2}}]]^{k-1}, e_{\tau\frac{h}{2}}(t_{k-1}^{+})\right)_{\Omega_i^k}\right| + \\ + \sum_{i=1}^{n_0}\left|\left(\tilde{z}_{\tau\frac{h}{2}}(0^-), e_{\tau\frac{h}{2}}(0^-)\right)_{\Omega_i^0}\right|.\end{aligned} \tag{23}$$

As in Section 5.1, from (23) we define the error estimators Est_k and η_i^k, $\forall i = 1, \ldots, n_k$ and $\forall k = 0, \ldots, m$. In this hybrid algorithm, we first solve the classical dual problem (7) in the fine mesh backwards in time. Then, we follow the same strategy defined in Section 5.2 employing $z_{\tau\frac{h}{2}}(0)$ as the initial condition for the pseudo-dual problem. Therefore, we need to solve the dual problem backwards in time once and then we can perform the entire goal-oriented adaptive process forward in time. In this algorithm, although we solve the classical dual problem once, we only need to store the solution in the first time step to start the adaptive process forwards in time as in Section 5.2. Figure 3 illustrates the proposed hybrid adaptive algorithm.

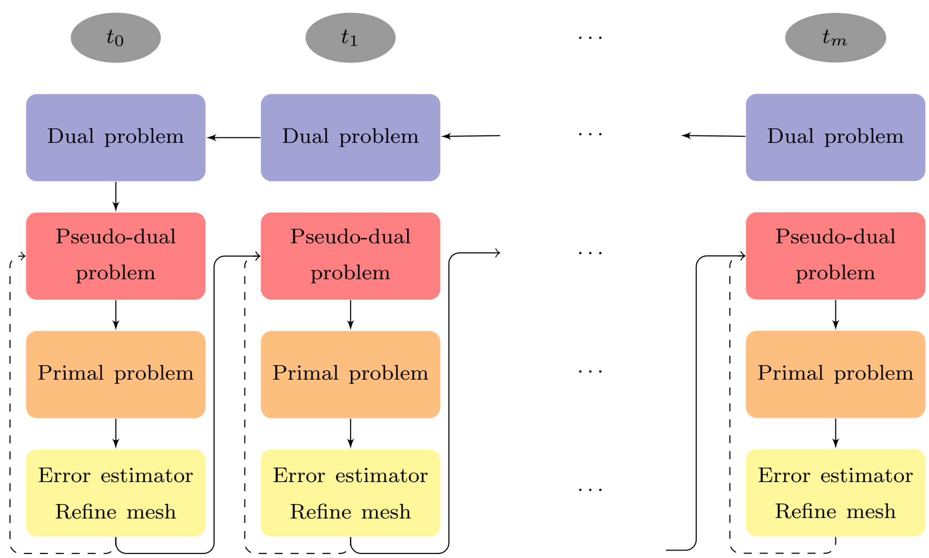


Figure 3: Proposed hybrid goal-oriented adaptive algorithm.

6. Numerical results

6.1. Example 1: Diffusion problem

We consider problem (1) with $d = 1$, $\Omega = [0, 1]$, $T = 1$, $f(x, t) = (1 + \pi^2 t)\sin(\pi x)$, $u(0) = 0$, $a = 0$ and a discontinuous diffusion coefficient

$$\kappa(x) = \begin{cases} 10, & x \in [0.25, 0.75], \\ 0.01, & \text{elsewhere.} \end{cases}$$

We also consider the following output functional

$$L(u) = \int_I \int_{\Omega_0} u(x, t)\, dxdt,$$

where $\Omega_0 = (0, 0.25) \cup (0.75, 1)$ is a subdomain of Ω. From (8), we have that the final condition of the dual problem is null and the source term is a function whose value is 1 in Ω_0 and vanishes outside Ω_0.

For the discretization, we employ constant-in-time basis functions ($r = 0$) and linear functions in space. We set 100 time steps and in space, we start the adaptive process with a coarse mesh consisting of 8 elements.

Figure 4 shows the solution of the primal problem (2) and Figure 5 shows the solution of the dual problems (7) and (13). We can see that the problem (13) is the same as (7) but running forward in time.

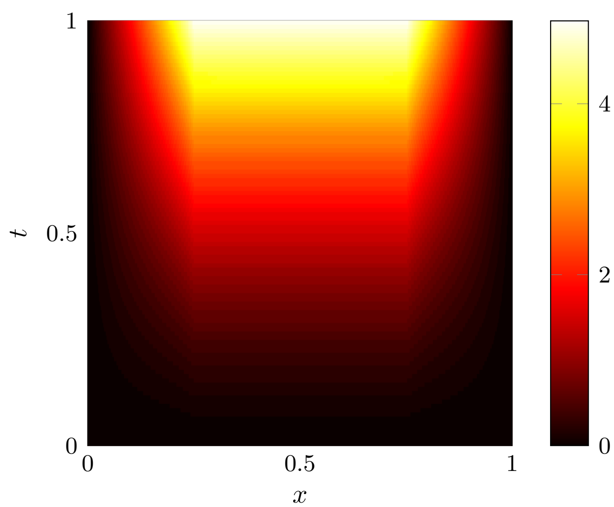
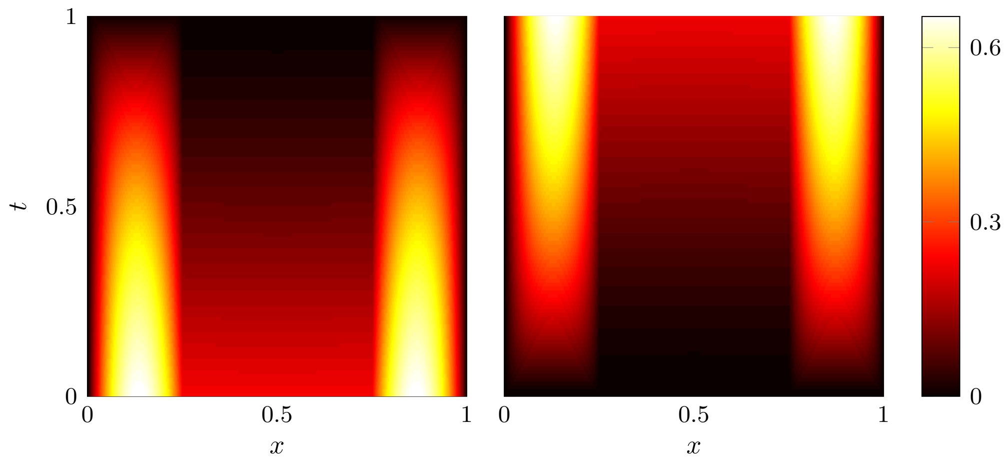


Figure 4: Solution of the primal problem.



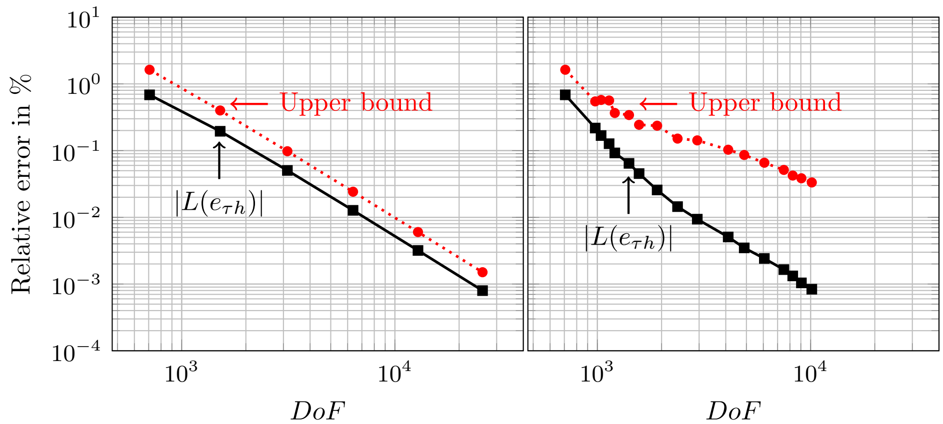
(a) Dual problem.

(b) Pseudo-dual problem.

Figure 5: Solution of the dual (left) and pseudo-dual (right) problems.

Now, we perform goal-oriented adaptivity employing both the classical strategy and the proposed forward-in-time process. Figure 6 shows the estimation and the upper bound of the error in the quantity of interest using the classical approach and setting $tol_1 = 10^{-3}$ and $\theta = \lambda = 0.25$. Figure 7 shows the error and the upper bound for the proposed algorithm when we set $tol_2 = tol_1/100$, $\lambda = 0.25$ and a maximum of seven iterations per time step. Finally, Figure 8 shows the adapted grids using both algorithms. We conclude that both grids are similar but with our algorithm we obtain more uniform refinements in the area where the diffusion term is discontinuous.

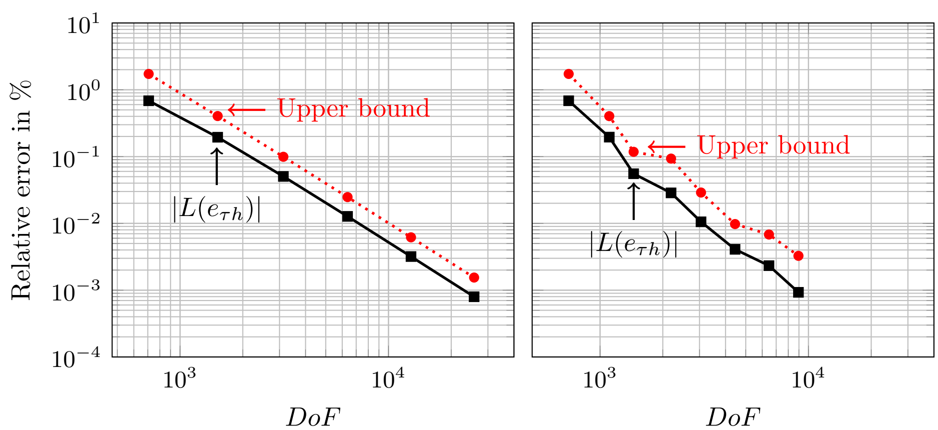
We conclude that, for this problem, the forward-in-time adaptive algorithm performs similarly to the classical algorithm. We achieve a relative error of 10^{-3} with 10^4 degrees of freedom in both cases. However, the proposed forward-in-time adaptive algorithm is computationally cheaper than the classical one as it performs the adaptivity while both problems, primal and dual, are solved forward in time. In that way, it is unnecessary to store the adjoint solution for all time steps, and one can only save the previous and current time steps solutions, maximizing memory savings. This also simplifies data structures and implementation. Also, it is expected that this algorithm will minimize the number of adaptive iterations, as confirmed via numerical results.



(a) Uniform refinements in space.

(b) Classical algorithm.

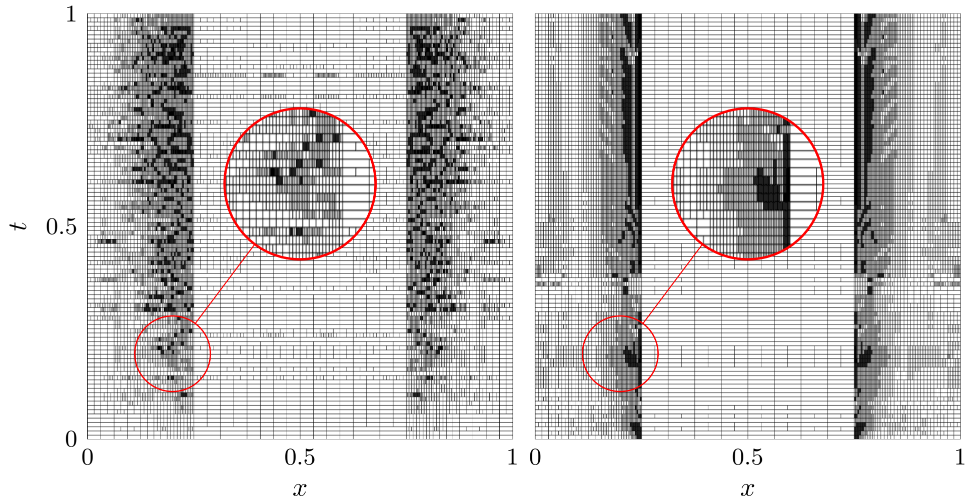
Figure 6: Error in the quantity of interest and upper bound (17) for uniform refinements in space (left) and using the classical algorithm (right).



(a) Uniform refinements in space.

(b) Forward-in-time algorithm.

Figure 7: Error in the quantity of interest and upper bound (20) for uniform refinements in space (left) and using the forward-in-time algorithm (right).



(a) Classical algorithm.

(b) Forward-in-time algorithm.

Figure 8: Adapted grids employing the classical algorithm (left) and the forward-in-time algorithm (right).

6.2. Example 2: Advection-diffusion problem

Let $d = 1$, $\Omega = [0, 1]$, $T = 0.25$, $\kappa = 0.025$, $a = 2.5$, $f(x, t) = 0$ and a discontinuous initial condition

$$u_0(x) = \begin{cases} 1, & x \in [0.125, 0.375], \\ 0, & \text{elsewhere.} \end{cases}$$

In this problem, the initial condition is propagated due to the positive advection coefficient and a boundary layer is formed at the final time steps in the right endpoint on the spatial domain. We seek to reduce the error in the boundary layer, therefore we consider an output functional of the form

$$L(u) = \int_{I_0} \int_{\Omega_0} u(x, t)\, dxdt,$$

where $I_0 \times \Omega_0 = (0.75, 1) \times (0.2, 0.25) \subset I \times \Omega$.

As before, we set 100 time steps and in space and we start the adaptive process with a coarse mesh composed of 8 elements. Figure 9 shows the solution of the primal problem (2) and Figure 10 shows the solution of the dual problem (7) and the pseudo-dual problem (13). In this case, the solution of the pseudo-dual problem is zero in the entire space-time domain except in the quantity of interest.

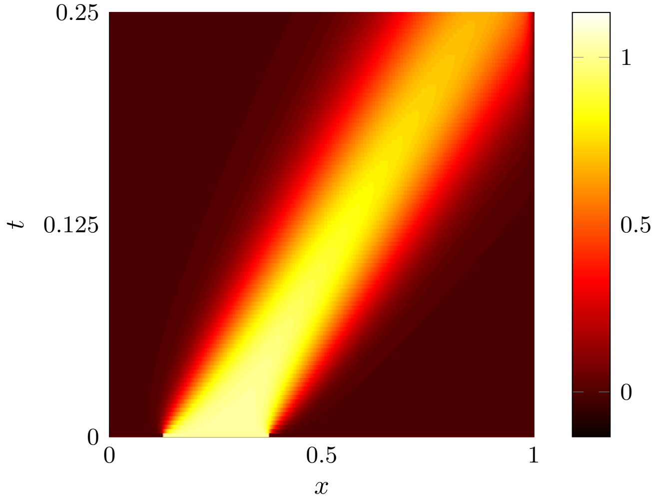
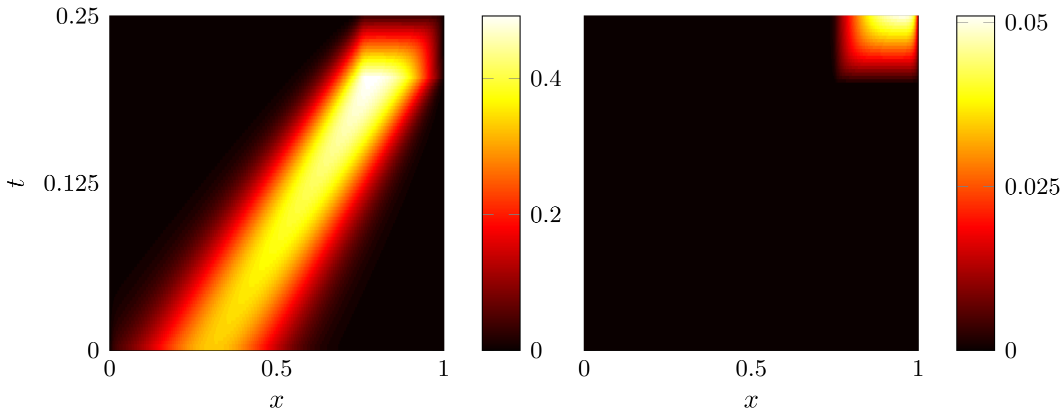


Figure 9: Solution of the primal problem.



(a) Dual problem.

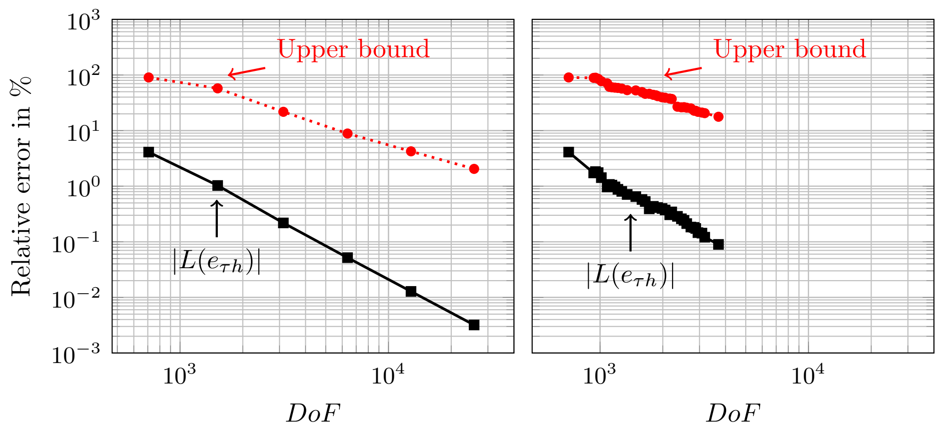
(b) Pseudo-dual problem.

Figure 10: Solution of the dual (left) and pseudo-dual (right) problems.

Figure 11 shows the error in the quantity of interest and the upper bound using the classical algorithm when we set $tol_1 = 10^{-1}$ and $\theta = \lambda = 0.2$. Figure 12 shows the error and the upper bound employing the proposed algorithm setting $tol_2 = 0$, $\lambda = 0.2$ and a maximum of seven iterations per time steps. Finally, Figure 13 shows the adapted grids using both processes.

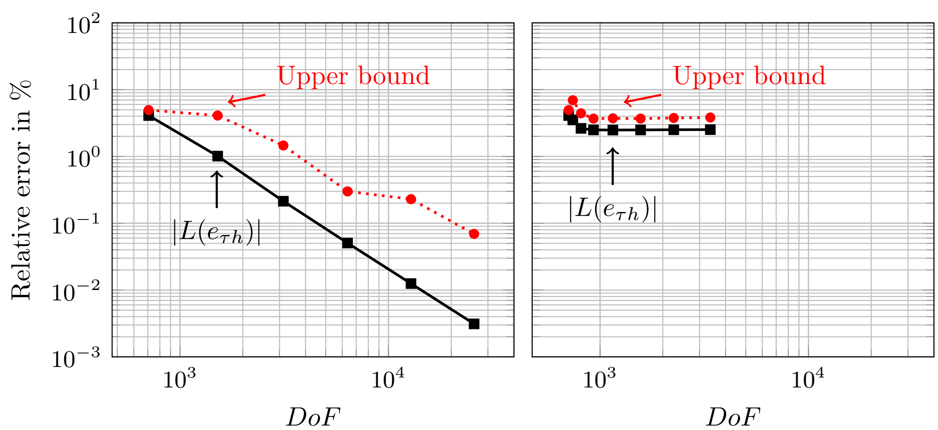
We conclude that, for this kind of problems, the proposed forward-in-time adaptive process performs poorly because the algorithm only produces refinements within the support of the output functional. By construction, this algorithm ignores the propagation of the initial condition of the primal problem due to the advection term.

As a partial remedy to this limitation of our forward-in-time algorithm, in the next section, we propose a hybrid algorithm that provides better results for advection-diffusion problems.



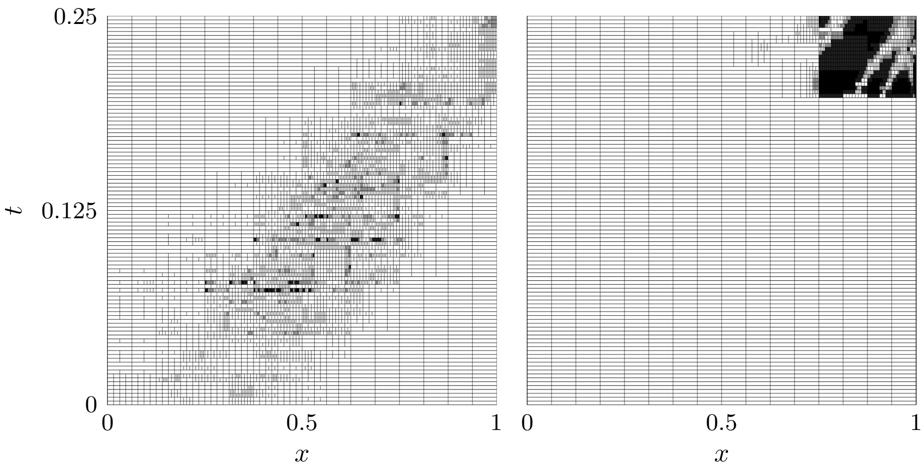
(a) Uniform refinements in space. (b) Classical algorithm.

Figure 11: Error in the quantity of interest and upper bound (17) for uniform refinements in space (left) and using the classical algorithm (right).



(a) Uniform refinements in space. (b) Forward-in-time algorithm.

Figure 12: Error in the quantity of interest and upper bound (20) for uniform refinements in space (left) and using the forward-in-time algorithm (right).



(a) Classical algorithm. (b) Forward-in-time algorithm.

Figure 13: Adapted grids employing the classical algorithm (left) and the forward-in-time algorithm (right).

6.3. Example 3: Hybrid algorithm for advection-diffusion problems

We consider the same example as in Section 6.2 and we employ the hybrid algorithm described in Section 5.3

Figure 14 shows the solution of the pseudo-dual problem. For the hybrid algorithm, we set $tol_2 = 0$, $\lambda = 0.2$ and a maximum of seven iterations per time step. Figure 15 shows the error in the quantity of interest and its upper bound, while Figure 16 exhibits the adapted grid employing the proposed hybrid algorithm.

We conclude that the hybrid algorithm not only refines the mesh where the support of the output functional is localized, but it also captures the path of the advection.

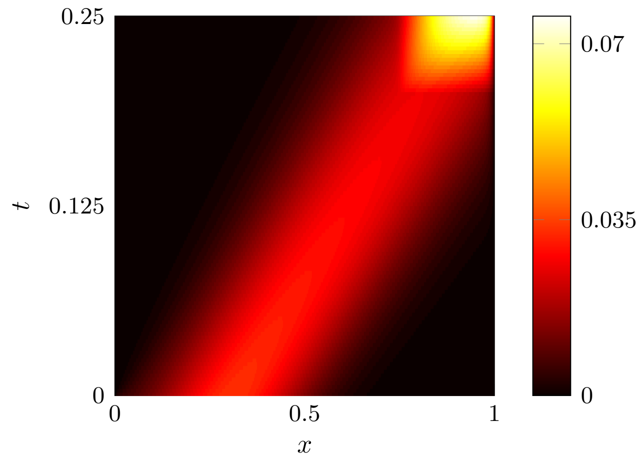
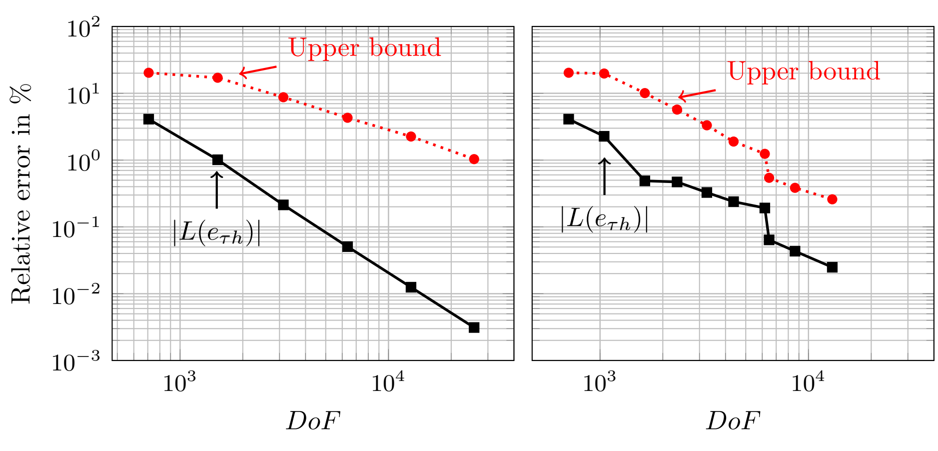


Figure 14: Solution of the pseudo-dual problem.



(a) Uniform refinements in space.

(b) Hybrid algorithm.

Figure 15: Error in the quantity of interest and upper bound (23) for uniform refinements in space (left) and employing the hybrid algorithm (right).

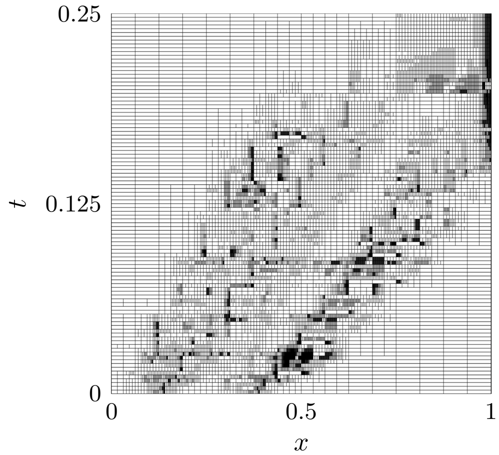


Figure 16: Adapted grid employing the proposed hybrid algorithm.

7. Conclusions

We propose a forward-in-time goal-oriented adaptive algorithm for the linear advection-diffusion equation. We define a pseudo-dual problem that, as the primal problem, runs forwards in time. Then, we derive an error representation employing the error of the primal problem and the solution of the pseudo-dual problem. The goal-oriented adaptive algorithm we present performs local refinements in space for a fixed time grid. We compare the proposed algorithm with the classical one for one-dimensional advection-diffusion problems. We conclude that our algorithm only performs properly for diffusion problems where the output functional has support in the whole time interval. For advection-diffusion problems, other strategies need to be considered. Herein, we propose to compute a first representation of the adjoint on the whole problem and use this partial information to guide the refinement process. We may pursue further refinements along these lines to improve the robustness of the method in a future work.

Acknowledgments

The first four authors have received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 777778 (MATHROCKS).

David Pardo, Elisabete Alberdi and Judit Muñoz-Matute were partially funded by the Basque Government Consolidated Research Group Grant

IT649-13 on “Mathematical Modeling, Simulation, and Industrial Applications (M2SI)” and the Projects of the Spanish Ministry of Economy and Competitiveness with reference MTM2016-76329-R (AEI/FEDER, EU), and MTM2016-81697-ERC/AEI.

David Pardo has also received funding from the BCAM “Severo Ochoa” accreditation of excellence SEV-2017-0718 and the Basque Government through the BERC 2018-2021 program.

Victor M. Calo was partially funded by the CSIRO Professorial Chair in Computational Geoscience at Curtin University, the Mega-grant of the Russian Federation Government (N 14.Y26.31.0013) and the Deep Earth Imaging Enterprise Future Science Platforms of the Commonwealth Scientific Industrial Research Organisation, CSIRO, of Australia. Additional support was provided at Curtin University by The Institute for Geoscience Research (TIGeR) and by the Curtin Institute for Computation.

Judit Muñoz-Matute has received funding from the University of the Basque Country (UPV/EHU) grant No. PIF15/346.

References

[1] M. Ainsworth and J. T. Oden. *A posteriori error estimation in finite element analysis*, volume 37. John Wiley & Sons, 2011.

[2] E. Alberdi and J. J. Anza. Solution of the wave-type PDE by numerical damping control multistep methods. *Procedia Computer Science*, 29:779–789, 2014.

[3] W. Bangerth, M. Geiger, and R. Rannacher. Adaptive Galerkin finite element methods for the wave equation. *Computational Methods in Applied Mathematics*, 10(1):3–48, 2010.

[4] W. Bangerth and R. Rannacher. Adaptive finite element techniques for the acoustic wave equation. *Journal of Computational Acoustics*, 9(2):575–591, 2001.

[5] W. Bangerth and R. Rannacher. *Adaptive finite element methods for differential equations*. Birkhäuser, 2013.

[6] J. H. Chaudhry, D. Estep, V. Ginting, J. N. Shadid, and S. Tavener. A posteriori error analysis of IMEX multi-step time integration methods for advection–diffusion–reaction equations. *Computer Methods in Applied Mechanics and Engineering*, 285:730–751, 2015.

[7] V. Darrigrand, D. Pardo, and I. Muga. Goal-oriented adaptivity using unconventional error representations for the 1D Helmholtz equation. *Computers & Mathematics with Applications*, 69(9):964–979, 2015.

[8] V. Darrigrand, Á. Rodríguez-Rozas, I. Muga, D. Pardo, A. Romkes, and S. Prudhomme. Goal-oriented adaptivity using unconventional error representations for the multidimensional Helmholtz equation. *International Journal for Numerical Methods in Engineering*, 113(1):22–42, 2018.

[9] P. Díez and G. Calderón. Goal-oriented error estimation for transient parabolic problems. *Computational Mechanics*, 39(5):631–646, 2007.

[10] S. Hussain, F. Schieweck, and S. Turek. Higher-order Galerkin time discretizations and fast multigrid solvers for the heat equation. *Journal of Numerical Mathematics*, 19(1):41–61, 2011.

[11] J. Muñoz-Matute, E. Alberdi, D. Pardo, and V. M. Calo. Time-domain goal-oriented adaptivity using pseudo-dual error representations. *Computer Methods in Applied Mechanics and Engineering*, 325:395–415, 2017.

[12] J. Muñoz-Matute, V. M. Calo, D. Pardo, E. Alberdi, and K. G. van der Zee. Explicit-in-time goal-oriented adaptivity. *Computer Methods in Applied Mechanics and Engineering*, 347:176–200, 2019.

[13] J. Muñoz-Matute, D. Pardo, V. M. Calo, and E. Alberdi. Variational formulations for explicit Runge-Kutta methods. *arXiv preprint arXiv:1806.07803*, 2018.

[14] J. T. Oden and S. Prudhomme. Goal-oriented error estimation and adaptivity for the finite element method. *Computers & Mathematics with Applications*, 41(5):735–756, 2001.

[15] D. Pardo, L. Demkowicz, C. Torres-Verdín, and M. Paszynski. Simulation of resistivity logging-while-drilling (LWD) measurements using a self-adaptive goal-oriented hp-finite element method. *SIAM Journal on Applied Mathematics*, 66(6):2085–2106, 2006.

[16] D. Pardo, L. Demkowicz, C. Torres-Verdin, and M. Paszynski. A self-adaptive goal-oriented hp-finite element method with electromagnetic applications. Part II: Electrodynamics. *Computer methods in applied mechanics and engineering*, 196(37):3585–3597, 2007.

[17] D. Pardo, L. Demkowicz, C. Torres-Verdín, and L. Tabarovsky. A goal-oriented hp-adaptive finite element method with electromagnetic applications. Part I: Electrostatics. *International Journal for Numerical Methods in Engineering*, 65(8):1269–1309, 2006.

[18] S. Prudhomme and J. T. Oden. Computable error estimators and adaptive techniques for fluid flow problems. *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*, 25:207–268, 2003.

[19] B. Rivière. *Discontinuous Galerkin methods for solving elliptic and parabolic equations: theory and implementation.* Society for Industrial and Applied Mathematics, 2008.

[20] A. Romkes, J. T. Oden, and K. Vemaganti. Multi-scale goal-oriented adaptive modeling of random heterogeneous materials. *Mechanics of materials*, 38(8):859–872, 2006.

[21] W. E. Schiesser. *The numerical method of lines: integration of partial differential equations.* Elsevier, 2012.

[22] M. Schmich and B. Vexler. Adaptivity with dynamic meshes for space-time finite element discretizations of parabolic equations. *SIAM Journal on Scientific Computing*, 30(1):369–393, 2008.

[23] G. Şimşek, X. Wu, K. van der Zee, and E. van Brummelen. Duality-based two-level error estimation for time-dependent PDEs: application to linear and nonlinear parabolic equations. *Computer Methods in Applied Mechanics and Engineering*, 288:83–109, 2015.

[24] K. Van der Zee, E. Van Brummelen, I. Akkerman, and R. De Borst. Goal-oriented error estimation and adaptivity for fluid–structure interaction using exact linearized adjoints. *Computer Methods in Applied Mechanics and Engineering*, 200(37):2738–2757, 2011.

[25] P. Vignal, N. Collier, L. Dalcin, D. Brown, and V. M. Calo. An energy-stable time-integrator for phase-field models. *Computer Methods in Applied Mechanics and Engineering*, 316:1179–1214, 2017.

[26] X. Wu, K. van der Zee, G. Simsek, and H. Van Brummelen. A posteriori error estimation and adaptivity for nonlinear parabolic equations using IMEX-Galerkin discretization of primal and dual equations. *arXiv preprint arXiv:1706.04281*, 2017.