

A–posteriori error estimators and RFB

Andrea Cangiani ^a Endre Süli

^aThe first author acknowledges the financial support of INdAM and EPSRC

We derive *a posteriori* error bounds for the *residual-free bubble* (RFB) method for the solution of convection–dominated diffusion equations. Both linear functional error control and energy norm error control are considered. The implementation of a reliable and efficient *h*–adaptive algorithm is discussed. Finally, we propose an *hb*–adaptive algorithm in which the local bubble stabilisation is automatically turned off (*b*–derefinement) in large parts of the computational domain during the *h*–refinement process, without compromising the accuracy of the method.

Subject classifications: AMS(MOS): 65N30, 76M10, 65N15

Key words and phrases: convection-diffusion problems, residual-free bubbles, a posteriori error analysis

Oxford University Computing Laboratory
Numerical Analysis Group
Wolfson Building
Parks Road
Oxford, England OX1 3QD

September, 2004

Contents

1	Introduction	3
2	<i>A posteriori</i> dual-weighted error bounds	4
2.1	Adaptive algorithm	7
2.2	Approximation of the dual solution	8
3	Implementation	13
4	The bubble as error estimator	22
5	<i>hb</i>-adaptivity	27
6	Conclusions	30

1 Introduction

The *residual-free bubble* (RFB) method is a two-level finite element method originally introduced by Brezzi and Russo [4] and Franca and Russo [14] for the stable and accurate computation of numerical solutions of convection-dominated diffusion problems.

Later on, the RFB method has been employed on a wide range of equations, such as the diffusion equation with rough coefficients [23], the Stokes equation [13], the incompressible Navier-Stokes equation [6] and the Helmholtz equation [12], showing that the method is quite general.

The RFB method consists of enriching a classical finite element (FE) space of piecewise polynomial functions with the richest possible space of bubble functions with respect to the given triangulation, i.e. the space of all H^1 functions with support contained in the elements of the triangulation. Static condensation of the bubble leads to a modified (generalised Galerkin) formulation in terms of the original FE space. In the case of convection-diffusion problems, this takes the form of a stabilised FE formulation in which the stabilising term depends on the bubble part of the numerical solution. This report is dedicated to the *a posteriori* error analysis of such RFB formulation.

In many scientific and engineering applications, the objective is to compute a few output functional with a prescribed accuracy. For this reason, the *a posteriori* analysis in terms of a given linear functional of the solution and its application in a reliable and effective adaptive mesh refinement algorithm, is the main concern of this paper. The algorithm proposed here is driven by a residual-based *a posteriori* error bound depending explicitly on the solution of a dual problem the data of which is the quantity of interest. See the Type I error bounds described by Giles and Süli in [15] and introduced by Becker and Rannacher in [1].

The *a posteriori* analysis of the energy-norm error of the RFB method has been considered previously by Russo [21] and Sangalli [22]. The advantage of our approach is that the error analysis is in terms of the piecewise polynomial part of the solution, rather than in terms of the component of the RFB solution from an *ad hoc* and problem-dependent function space as, for instance, in [22].

The *a posteriori* error bound is composed of three terms: the two classical residual-based terms of the Galerkin formulation (internal residual and boundary jump of the gradient), and a third term due to static condensation of the bubbles. We particularly emphasize the relevance of the bubble term in our error estimator, extending the result discussed in [21], concerning the equivalence of the L_2 -norm of the bubble with the elemental L_2 -norm of the residual, to non-constant coefficient problems.

In Section 5 we propose to use the relative magnitude of the terms in the error bound to explore whether the bubble (stabilisation) can be turned off locally. This idea leads to a new *hb*-adaptive algorithm in which the bubble stabilisation is automatically phased out locally (*b*-derefinement) while the mesh is refined (*h*-refinement). We investigate the robustness of the algorithm through numerical examples.

The paper is organized as follows. In Section 2 we present the *a posteriori* dual-weighted error analysis in terms of linear functional evaluation. Section 3 is dedicated to the implementation of an adaptive mesh refinement algorithm and to some numerical

examples. In Section 4 we present the *a posteriori* energy–norm error analysis and show that the L_2 –norm of the bubble is bounded by the L_2 –norm of the internal residual. Finally, in Section 5 we define the *hb*–adaptive algorithm and test it on two numerical examples characterized by a boundary layer and an internal layer, respectively.

2 *A posteriori* dual–weighted error bounds

Our aim is to combine the use of the RFB method for the numerical solution of boundary value problems for convection–dominated convection–diffusion equations with mesh adaptation techniques.

Let us consider the convection–diffusion operator

$$Lu := -\varepsilon\Delta u + \mathbf{a} \cdot \nabla u,$$

where ε is a positive constant and the velocity field $\mathbf{a} \in L^\infty(\Omega)^2$ is divergence-free in an open polygonal domain $\Omega \subset \mathbb{R}^2$. Given a function $f \in L_2(\Omega)$, we consider the associated homogeneous boundary–value problem in variational form

$$\begin{cases} \text{find } u \in V \text{ such that} \\ \mathcal{L}(u, v) = (f, v) \quad \forall v \in V, \end{cases} \quad (2.1)$$

where $V = H_0^1(\Omega)$, (\cdot, \cdot) denotes the inner product in $L_2(\Omega)$ and the bilinear form $\mathcal{L}(\cdot, \cdot)$ is defined on $V \times V$ as

$$\mathcal{L}(w, v) = \varepsilon \int_{\Omega} \nabla w \cdot \nabla v \, d\mathbf{x} + \int_{\Omega} (\mathbf{a} \cdot \nabla w) v \, d\mathbf{x}.$$

It is often the case that the quantity of interest is not the solution u itself but a linear functional $u \rightarrow J(u)$ of the solution (e.g. a point value, a flux, an average, etc.). *A posteriori* bounds on the error $J(u) - J(u_h)$, where u_h is the computed solution, can be obtained through duality arguments. Here we assume that

$$J(u) = (u, g), \quad g \in L_2(\omega),$$

leaving the treatment of other linear functionals such as fluxes through a Neumann boundary to a later example.

We define the following dual problem:

$$\begin{cases} \text{find } z \in V \text{ such that} \\ \mathcal{L}(w, z) = J(w) \quad \forall w \in V. \end{cases} \quad (2.2)$$

The (adjoint) differential operator L^* involved in (2.2) can be recovered from (2.2) through integration by parts; for the partial differential operator L under consideration, it will be found to be defined by

$$L^*z := -\varepsilon\Delta z - \mathbf{a} \cdot \nabla z \quad z \in V.$$

We shall perform such an *a posteriori* error analysis for the RFB method seen as a generalised Galerkin approximation for the polynomial part of the RFB solution. That is, we assume that static condensation of the bubble part of the solution has been performed resulting in a stabilised finite element approximation on the given piecewise polynomial space V_h and triangulation \mathcal{T}_h ; i.e., a formulation which we can write as

$$\begin{cases} \text{find } u_h \in V_h \text{ such that} \\ \mathcal{L}(u_h, v_h) + \sum_{T \in \mathcal{T}_h} (u_b, L^* v_h)_T = (f, v_h) \quad \forall v_h \in V_h, \end{cases} \quad (2.3)$$

where u_b is defined element-wise as the unique solution in $H_0^1(T)$ of

$$L_T u_b|_T = (f - L u_h)|_T.$$

Here, $L_T : H_0^1(T) \rightarrow H^{-1}(T)$ denotes the restriction of the operator L to T . See, e.g., [3] for details. Regarding the issue of the computation of the bubble u_b , see Section 3 where the implementation of the method is discussed.

The family of triangulations \mathcal{T}_h , $h > 0$, is *admissible (regular)* if any two triangles in \mathcal{T}_h either have a common edge or common vertex, or they do not intersect at all.

In the sequel, we denote by \mathbf{n}_T the unit outward normal vector to ∂T defined on the edges of any element T . Further, we denote by $\mathbf{n}_T \cdot [\nabla u_h]$ the jump of the normal derivative of u_h across the given edge.

We have

$$\begin{aligned} J(u) - J(u_h) &= J(u - u_h) \\ &= \mathcal{L}(u - u_h, z) \\ &= \mathcal{L}(u - u_h, z - z_h) + \sum_{T \in \mathcal{T}_h} (u_b, L^* z_h)_T \\ &= \sum_{T \in \mathcal{T}_h} ((f - L u_h, z - z_h)_T - (\varepsilon \mathbf{n}_T \cdot \nabla u_h, z - z_h)_{\partial T \cap \Omega}) + \sum_{T \in \mathcal{T}_h} (u_b, L^* z_h)_T \\ &= \sum_{T \in \mathcal{T}_h} \left((R_T(u_h), z - z_h)_T - \frac{1}{2} (\varepsilon \mathbf{n}_T \cdot [\nabla u_h], z - z_h)_{\partial T \cap \Omega} + (u_b, L^* z_h)_T \right) \\ &= \sum_{T \in \mathcal{T}_h} ((\rho_T^1, \omega_T^1)_T + (\rho_T^2, \omega_T^2)_{\partial T \cap \Omega} + (\rho_T^3, \omega_T^3)_T), \end{aligned} \quad (2.4)$$

having denoted the elemental residual terms by

$$\rho_T^1 = R_T(u_h), \quad \rho_T^2 = -\frac{1}{2} \varepsilon \mathbf{n}_T \cdot [\nabla u_h], \quad \rho_T^3 = u_b,$$

and defined the weights

$$\omega_T^1 = \omega_T^2 = z - z_h, \quad \omega_T^3 = L^* z_h.$$

Thus, the error representation formula (2.4) is a sum of three terms: the two classical residual-based terms and a third one due to the stabilisation term in (2.3). This

identity is analogous to the error representation formulas for stabilised finite element approximations of first-order hyperbolic problems presented by Houston *et al.* in [16]. In particular, we will see later that, when the RFB method is equivalent to the stabilised finite element method considered by the authors of [16] in the context of first-order hyperbolic PDEs, then the identity (2.4) corresponds to their first error representation formula for functionals.

We now discuss a mesh adaptation algorithm based on the *a posteriori* error representation (2.4).

Given a positive tolerance TOL, the goal is the computation of $u_h \in V_h$ such that

$$|J(u) - J(u_h)| \leq \text{TOL}. \quad (2.5)$$

From (2.4) we have that

$$|J(u) - J(u_h)| = \left| \sum_{T \in \mathcal{T}_h} ((\rho_T^1, \omega_T^1)_T + (\rho_T^2, \omega_T^2)_{\partial T \cap \Omega} + (\rho_T^3, \omega_T^3)_T) \right| =: \mathcal{E}_1(u_h). \quad (2.6)$$

Thus, the constraint (2.5) is satisfied as soon as

$$\mathcal{E}_1(u_h) \leq \text{TOL}.$$

This will be our *stopping criterion*.

We now need to choose a *refinement criterion*, i.e. a marking strategy for the refinement of the elements in the mesh, and establish a way to actually compute the error representation formula (2.6).

The design of a refinement criterion is based on the localization of $\mathcal{E}_1(u_h)$. Putting the absolute value sign under the summation sign, we get

$$|J(u) - J(u_h)| \leq \sum_{T \in \mathcal{T}_h} |(\rho_T^1, \omega_T^1)_T + (\rho_T^2, \omega_T^2)_{\partial T \cap \Omega} + (\rho_T^3, \omega_T^3)_T| =: \sum_{T \in \mathcal{T}_h} \eta_T =: \mathcal{E}_2(u_h),$$

and a decision as to which elements to refine can now be taken depending on the magnitude of the *local error indicator* η_T .

There are many possible refinement criteria; see [1] for a review. An optimal strategy, known as *error per cell strategy*, would be to equilibrate the local error indicator η_T by refining or coarsening according to the criterion

$$\eta_T \approx \frac{\text{TOL}}{N_{\text{el}}},$$

where N_{el} is the number of elements in the subdivision.

A criterion that may be more suitable if coarsening is not considered, is the *fixed fraction strategy* in which the elements are ordered according to the size of η_T and then some portion of those with largest η_T is refined. We have chosen to use the fixed fraction strategy proposed by Papastavrou and Verfürth in their article dedicated to the

comparison of *a posteriori* error estimators for convection–diffusion problems [19]. The authors of [19] suggest to refine those elements for which

$$\eta_T \geq c_{\text{ref}} \bar{\eta},$$

for some user–selected threshold parameter $c_{\text{ref}} \in (0, 1)$; the reference value $\bar{\eta}$ is taken to be the maximum of η_T after cutting the upper 10% or 5% of the values (in order to preclude runaway values). That is, a second parameter p_{ref} , usually fixed to 0.1 or 0.05, is defined and $\bar{\eta}$ is obtained as the maximum of η_T after discarding the $\lfloor p_{\text{ref}} N_{\text{el}} \rfloor$ elements with the largest η_T .

Regarding the computation of (2.6), the difficulty is in the evaluation of the dual solution. Here the dual solution z is computed using a new mesh \mathcal{T}_H , different from \mathcal{T}_h . Given the finite element space V_H corresponding to \mathcal{T}_H and the RFB solution $z_H \in V_H$ of the dual problem (2.2), the approximation z_h is taken as the projection or the interpolant of the computed z_H from the primal finite element space.

We reconsider the error representation formula (2.6). By decomposing

$$\omega_T^1 = (z - z_h)|_T = (z_H - z_h)|_T + (z - z_H)|_T = \tilde{\omega}_T + \bar{\omega}_T,$$

we get

$$\begin{aligned} |J(u) - J(u_h)| &\leq \left| \sum_{T \in \mathcal{T}_h} ((\rho_T^1, \tilde{\omega}_T)_T + (\rho_T^2, \tilde{\omega}_T)_{\partial T \cap \Omega} + (\rho_T^3, \omega_T^3)_T) \right| \\ &\quad + \left| \sum_{T \in \mathcal{T}_h} ((\rho_T^1, \bar{\omega}_T)_T + (\rho_T^2, \bar{\omega}_T)_{\partial T \cap \Omega}) \right| \\ &= \tilde{\mathcal{E}}_1(u_h) + \bar{\mathcal{E}}_1(u_h). \end{aligned} \tag{2.7}$$

In this way we have isolated in $\bar{\mathcal{E}}_1(u_h)$ the uncomputable terms of the error bound.

2.1 Adaptive algorithm

Let us assume for a moment that $\bar{\mathcal{E}}_1(u_h)$, i.e. the term depending on the difference $z - z_H$, is such that $\bar{\mathcal{E}}_1(u_h) \ll \tilde{\mathcal{E}}_1(u_h)$. The validity of this hypothesis will be discussed in Section 3 below. We then define the new local refinement indicator

$$\tilde{\eta}_T = |(\rho_T^1, \tilde{\omega}_T)_T + (\rho_T^2, \tilde{\omega}_T)_{\partial T \cap \Omega} + (\rho_T^3, \omega_T^3)_T| \quad \forall T \in \mathcal{T}_h, \tag{2.8}$$

and redefine $\bar{\eta}$ accordingly. Further, we define the new stopping criterion

$$\tilde{\mathcal{E}}_1(u_h) \leq \frac{\text{TOL}}{C}, \tag{2.9}$$

for some constant $1 < C < 2$.

We may then consider the following adaptive algorithm:

1. Define an initial mesh;
2. Calculate u_h and the dual solutions z_H and z_h on the current meshes;
3. Check the stopping criterion: IF $\tilde{\mathcal{E}}_1(u_h) \leq \text{TOL}/C$ then STOP;
4. Apply the refinement criterion: refine those elements T whose local error indicator η_T exceeds $c_{\text{ref}}\bar{\eta}$ and GOTO 2.

2.2 Approximation of the dual solution

To ensure that the stopping criterion (2.9) is reliable, i.e. that the approximation error is below the given tolerance, we need to control the size of $\bar{\mathcal{E}}_1(u_h)$. To this end, we observe that the global-residual

$$R(u_h) : v \longrightarrow (f, v) - \mathcal{L}(u_h, v),$$

is a bounded linear functional in V . Hence, returning to (2.4), we see that we can write the term $\bar{\mathcal{E}}_1(u_h)$ as

$$\begin{aligned} \mathcal{L}(u - u_h, z - z_H) &= \mathcal{L}(u, z - z_H) - \mathcal{L}(u_h, z - z_H) \\ &= (f, z - z_H)_\Omega - \mathcal{L}(u_h, z - z_H) \\ &= \langle R(u_h), z - z_H \rangle, \end{aligned} \tag{2.10}$$

where $\langle \cdot, \cdot \rangle$ is the duality pairing between V and its dual space V' . We now notice that the right-hand side in (2.10) defines a new linear functional

$$N(v) = \langle R(u_h), v \rangle.$$

Thus, we can estimate the error terms in $\bar{\mathcal{E}}_1(u_h)$ by performing an *a posteriori* analysis of the error $N(z) - N(z_H)$; for this purpose we consider the dual of the dual problem

$$\begin{cases} \text{find } t \in V \text{ such that} \\ \mathcal{L}(t, v) = N(v) \quad \forall v \in V. \end{cases} \tag{2.11}$$

Let $t_H \in V_H$ be some approximation of t . We have

$$\begin{aligned} N(z) - N(z_H) &= N(z - z_H) \\ &= \mathcal{L}(t, z - z_H) \\ &= \mathcal{L}(t - t_H, z - z_H) + \sum_{K \in \mathcal{T}_H} (Lt_H, z_b)_K \\ &= \sum_{K \in \mathcal{T}_H} \left((t - t_H, R_K(z_H))_K - \frac{1}{2} \varepsilon (t - t_H, \mathbf{n}_K \cdot [\nabla z_H])_{\partial K \cap \Omega} + (Lt_H, z_b)_K \right) \end{aligned}$$

where, as before, the elemental-residual is defined as $R_K(z_H) = (g - L^*z_H)|_K$. Finally, z_b represents the bubble part of the RFB solution to the dual problem.

The new error representation just obtained is in terms of $t - t_H$ which is just as uncomputable as $z - z_H$. To avoid a possibly infinite sequence of duality arguments, we bound $t - t_H$ in terms of a stability constant. The bound obtained in this way need not be sharper than the one we would obtain if instead we were to bound $z - z_H$ directly as was done by Eriksson *et al.* in [10] and [11]. From the practical point of view, though, the crudeness of the bound of $\tilde{\mathcal{E}}_1(u_h)$ is not of particular concern since all we need $\tilde{\mathcal{E}}_1(u_h)$ for is to generate an adequate sequence of finite element approximations z_H which we can use to compute $\tilde{\mathcal{E}}_1(u_h)$.

From now on, we assume that linear finite elements are used, so

$$V_H = \{ \varphi \in C(\bar{\Omega}) : \varphi|_K \in \mathcal{P}_1 \quad \forall K \in \mathcal{T}_H \},$$

and assume that for any triangulation \mathcal{T}_H and any element $T \in \mathcal{T}_H$ the number of neighbors of T is bounded.

We choose $t_H = P_H t$, where P_H is the modification of the quasi-interpolation operator of Clément [9] analysed by Verfürth [25]. With this choice we can take advantage of explicit interpolation error bounds in terms of the H^1 -seminorm.

The definition of the modified Clément's quasi-interpolant requires the introduction of the following notational conventions. Let \mathcal{E}_H and \mathcal{N}_H be the sets of all the edges and all the vertices in \mathcal{T}_H , respectively. Further, let

$$\mathcal{E}_H = \mathcal{E}_{H,\Omega} \cup \mathcal{E}_{H,D}, \quad \mathcal{N}_H = \mathcal{N}_{H,\Omega} \cup \mathcal{N}_{H,D},$$

be the decompositions of \mathcal{E}_H and \mathcal{N}_H into the subsets of internal and boundary edges and vertices, respectively. For any $S \in \mathcal{T}_H \cup \mathcal{E}_H$, let $\mathcal{N}(S)$ be the set of its vertices; and for any $K \in \mathcal{T}_H$, let $\mathcal{E}(K)$ be the set of all faces of K which are internal to Ω . Finally, for any vertex $x \in \mathcal{N}_H$, denote by ω_x the union of all triangles which have x as a vertex.

The following definition and bounds can be extended to $H_{\Gamma_D}^1(\Omega)$ for any $\Gamma_D \subset \partial\Omega$, essentially by interpreting the nodes sitting on the Neumann boundary $\partial\Omega \setminus \Gamma_D$ as internal.

Given a function $u \in L_2(\omega_x)$ we associate to any $x \in \mathcal{N}_H$ the value

$$\pi_x u = \frac{1}{|\omega_x|} \int_{\omega_x} u.$$

The quasi-interpolation operator $P_H : V \rightarrow V_H$ is defined as follows:

$$P_H u = \sum_{x \in \mathcal{N}_{H,\Omega}} (\pi_x u) \varphi_x,$$

where $\varphi_x \in V_H$ is the finite element basis function associated with x . Notice that P_H is *not* a projection operator.

Lemma 2.1 *For all $v \in V$, all $K \in \mathcal{T}_H$ and all $E \in \mathcal{E}_H$ we have*

$$\|v - P_H v\|_{0,K} \leq \sum_{x \in \mathcal{N}(K)} C_{K;x} H_x |v|_{1,\omega_x}, \quad (2.12)$$

$$\|v - P_H v\|_{0,E} \leq \sum_{x \in \mathcal{N}(E)} C_{E;x} H_x^{1/2} |v|_{1,\omega_x}, \quad (2.13)$$

where H_x is the maximum length of an edge having x as an end point. The values of the constants $C_{K;x}$ and $C_{E;x}$ are given explicitly by Verfürth [25] in terms of the following mesh-quality related quantities

$$\kappa_{3,x} := \max_{\substack{K_1, K_2 \in \mathcal{T}_H \\ x \in \mathcal{N}(K_1) \cap \mathcal{N}(K_2)}} \frac{|K_1|}{|K_2|}, \quad \kappa_{4,x} := \max_{\substack{K \in \mathcal{T}_H, E \in \mathcal{E}_H \\ x \in \mathcal{N}(K) \cap \mathcal{N}(E)}} \frac{|E|H_K}{|K|}.$$

Moreover, the following bound holds on the interpolation error in the H^1 -seminorm:

$$|v - P_H v|_{1,K} \leq \sum_{x \in \mathcal{N}(K)} C'_{K;x} |v|_{1,\omega_x}, \quad (2.14)$$

with the constant $C'_{K;x}$ depending explicitly on $\kappa_{3,x}$ and on the shape-regularity of K , i.e. the ratio $\kappa_K = H_K/\rho_K$ between the diameter of K and the diameter of the largest circle inscribed into K .

Proof. The bounds (2.12) and (2.13) are due to Verfürth, see [25]. We prove (2.14) by adapting Verfürth's proof of (2.12). To start with, (2.14) is to be shown for an operator which does not need to satisfy the Dirichlet boundary conditions. That is, a new operator \bar{P}_H , a modification of P_H , is defined as

$$\bar{P}_H u = \sum_{x \in \mathcal{N}_H} (\pi_x u) \varphi_x.$$

Fix an arbitrary $K \in \mathcal{K}_H$. Since $\sum_{x \in \mathcal{N}(K)} \varphi_x = 1$ on K , we have

$$\begin{aligned} \|\nabla(u - \bar{P}_H u)\|_{0,K} &= \left\| \sum_{x \in \mathcal{N}(K)} ((u - \pi_x u) \nabla \varphi_x + (\nabla u) \varphi_x) \right\|_{0,K} \\ &\leq \sum_{x \in \mathcal{N}(K)} \|\nabla \varphi_x\|_{\infty,K} \|u - \pi_x u\|_{0,K} + \sum_{x \in \mathcal{N}(K)} \|\varphi_x\|_{\infty,K} \|\nabla u\|_{0,K} \\ &\leq \sum_{x \in \mathcal{N}(K)} \frac{\kappa_K}{H_K} \|u - \pi_x u\|_{0,K} + \sum_{x \in \mathcal{N}(K)} \|\nabla u\|_{0,K}. \end{aligned} \quad (2.15)$$

We have now reached the point at which this part of the proof follows that of (2.12) given in [25].

The following result is Lemma 4.3 in [25]: for any $x \in \mathcal{N}_H$ and $u \in H^1(\omega_x)$ we have

$$\|u - \pi_x u\|_{0,\omega_x} \leq C_x H_x \|\nabla u\|_{0,\omega_x}, \quad (2.16)$$

where C_x is an explicit constant depending on the ratio

$$\kappa_{1,x} = \frac{H_x}{\rho_x},$$

and ρ_x is the minimum length of an edge having x as an end point (in particular, if ω_x is convex, then $C_x = 2/\pi$).

Applying (2.16) in (2.15) we obtain the desired result for \bar{P}_H .

The proof is completed by bounding the norm of the difference between $\nabla P_H u$ and $\nabla \bar{P}_H u$:

$$\begin{aligned} \|\nabla(P_H u - \bar{P}_H u)\|_{0,K} &= \left\| \sum_{x \in \mathcal{N}(K) \cap \mathcal{N}_{H,D}} \pi_x u \nabla \varphi_x \right\|_{0,K} \\ &\leq \sum_{x \in \mathcal{N}(K) \cap \mathcal{N}_{H,D}} |\pi_x u| \|\nabla \varphi_x\|_{0,K} \\ &\leq \sum_{x \in \mathcal{N}(K) \cap \mathcal{N}_{H,D}} \frac{\sqrt{12} \kappa_K}{H_K} |\pi_x u| \|\varphi_x\|_{0,K}. \end{aligned}$$

We are again within the framework of the proof of (2.12) given in [25]. This time, the proof is completed by using the following result (equation (5.6) in [25]):

$$|\pi_x u| \|\varphi_x\|_{0,K} \leq \frac{1}{2} \left(\frac{|K|}{|K_x|} \right)^{1/2} (\|u - \pi_x u\|_{0,K_x} + H_x \|\nabla u\|_{0,K_x}),$$

(here K_x represents any triangle K with the following properties: x is a vertex of K and K shares an entire edge with $\partial\Omega$), and applying again (2.16). \square

We are now ready to obtain a computable bound on $|N(z) - N(z_H)|$. By applying the Cauchy–Schwarz inequality and the interpolation error bounds (2.12), (2.13) and (2.14) after noting that $LP_H t = \mathbf{a} \cdot \nabla P_H t$, we have

$$\begin{aligned} |N(z) - N(z_H)| &\leq \sum_{K \in \mathcal{T}_H} \left(\|R_K(z_H)\|_{0,K} \|t - P_H t\|_{0,K} \right. \\ &\quad \left. + \sum_{E \in \mathcal{E}(K)} \left\| \frac{1}{2} \varepsilon \mathbf{n}_K \cdot [\nabla z_H] \right\|_{0,E} \|t - P_H t\|_{0,E} + \|z_b\|_{0,K} \|\mathbf{a} \cdot \nabla P_H t\|_{0,K} \right) \\ &\leq \sum_{K \in \mathcal{T}_H} \left(\|R_K(z_H)\|_{0,K} \sum_{x \in \mathcal{N}(K)} C_{K;x} H_x |t|_{1,\omega_x} \right. \\ &\quad \left. + \sum_{E \in \mathcal{E}(K)} \left(\left\| \frac{1}{2} \varepsilon \mathbf{n}_K \cdot [\nabla z_H] \right\|_{0,E} \sum_{x \in \mathcal{N}(E)} C_{E;x} H_x^{1/2} |t|_{1,\omega_x} \right) \right. \\ &\quad \left. + \|\mathbf{a}\|_{\infty,K} \|z_b\|_{0,K} \left(|t|_{1,K} + \sum_{x \in \mathcal{N}(K)} C'_{K;x} |t|_{1,\omega_x} \right) \right). \end{aligned}$$

The solution t_H has now been removed from the bound, but t is still present. We can eliminate t as follows, at the expense of breaking up the sum over the elements of the

triangulation. Let

$$\begin{aligned} C_K &= \max_{x \in \mathcal{N}(K)} (C_{K;x} \kappa_{1,x}), \\ C_{\partial K} &= \max_{E \in \mathcal{E}(K)} \max_{x \in \mathcal{N}(E)} (C_{E;x} \kappa_{1,x}^{1/2}), \\ C'_K &= 1 + \max_{x \in \mathcal{N}(K)} C'_{K;x}. \end{aligned}$$

Then, since the number of neighbors of any $T \in \mathcal{T}_H$ is bounded, we get

$$\begin{aligned} & |N(z) - N(z_H)| \\ & \leq \sum_{K \in \mathcal{T}_H} \left((C_K H_K \|R_K(z_H)\|_{0,K} + 2C_{\partial K} H_K^{1/2} \|\frac{1}{2} \varepsilon \mathbf{n}_K \cdot [\nabla z_H]\|_{0,\partial K \cap \Omega}) \sum_{x \in \mathcal{N}(K)} |t|_{1,\omega_x} \right. \\ & \quad \left. + C'_K \|\mathbf{a}\|_{\infty,K} \|z_b\|_{0,K} \sum_{x \in \mathcal{N}(K)} |t|_{1,\omega_x} \right) \\ & \leq 3 \left(\sum_{K \in \mathcal{T}_H} (C_K H_K \|R_K(z_H)\|_{0,K} + C_{\partial K} H_K^{1/2} \|\varepsilon \mathbf{n}_K \cdot [\nabla z_H]\|_{0,\partial K \cap \Omega} \right. \\ & \quad \left. + C'_K \|\mathbf{a}\|_{\infty,K} \|z_b\|_{0,K}^2 \right)^{1/2} |t|_{1,\Omega}. \end{aligned}$$

To quantify $|t|_{1,\Omega}$, we reconsider the problem (2.11) in strong form with forcing term $R(u_h) \in H^{-1}(\Omega)$:

$$\begin{cases} Lt = R(u_h) & \text{in } \Omega, \\ t = 0 & \text{on } \partial\Omega. \end{cases} \quad (2.17)$$

Multiplying the first equation in (2.17) by t and integrating by parts over Ω , we get

$$\varepsilon |t|_{1,\Omega}^2 = \langle R(u_h), t \rangle \leq |t|_{1,\Omega} \sup_{v \in V \setminus \{0\}} \frac{\langle R(u_h), v \rangle}{|v|_{1,\Omega}}.$$

Hence, we have the stability estimate

$$|t|_{1,\Omega} \leq \varepsilon^{-1} \|R(u_h)\|_{-1,\Omega},$$

and we conclude that

$$\begin{aligned} & |N(z) - N(z_H)| \\ & \leq 3 \left(\sum_{K \in \mathcal{T}_H} (C_K H_K \|\varepsilon^{-1} R_K(z_H)\|_{0,K} + C_{\partial K} H_K^{1/2} \|\mathbf{n}_K \cdot [\nabla z_H]\|_{0,\partial K \cap \Omega}) \right. \\ & \quad \left. + \varepsilon^{-1} C'_K \|\mathbf{a}\|_{\infty,K} \|z_b\|_{0,K}^2 \right)^{1/2} \|R(u_h)\|_{-1,\Omega} \\ & := \bar{\mathcal{E}}_2(u_h). \end{aligned} \quad (2.18)$$

In this way, we have obtained the following computable error bound:

$$|J(u) - J(u_h)| \leq \tilde{\mathcal{E}}_1(u_h) + \bar{\mathcal{E}}_2(u_h) \quad (2.19)$$

Two remarks are in order. First, the number $\|R(u_h)\|_{-1,\Omega}$ is not directly computable from u_h . We can still compute it, but at the expense of the solution of an auxiliary problem; see the next section for details. Finally, the derivation of sharp and computable *a posteriori* error bounds is a nontrivial task. Our bound is no exception in this respect: by bounding $z - z_H$ we achieved computability and reliability of the bound, but we anticipate a loss in terms of sharpness.

In the next section, we investigate the sharpness of the error bound (2.19), and study the reliability and effectivity of the alternative error estimator $\bar{\mathcal{E}}_1(u_h)$.

3 Implementation

We experiment with the *a posteriori* error bound described in the previous section by considering linear finite elements on triangles for both the primal and dual computations. We have chosen to define \mathcal{T}_H as the triangulation obtained by subdividing every triangle in \mathcal{T}_h into the four triangles obtained by bisection of its edges. A survey of other possible choices can be found, for example, in [1]; see also the comments in the review article by Giles and Süli [15].

Let $V_h \subset V = H_0^1(\Omega)$ denote the usual space of linear finite elements

$$V_h = \{ \varphi \in C(\bar{\Omega}) : \varphi|_T \in \mathcal{P}_1 \quad \forall T \in \mathcal{T}_h \},$$

and similarly for V_H .

To implement the RFB formulation (2.3) over V_h we need to calculate the bubble part of the solution u_b . As explained, for example, in [4] and [3], this is given on every element $T \in \mathcal{T}_h$ as the solution in $H_0^1(T)$ of the *bubble equation*

$$L_T u_b|_T = (f - L u_h)|_T,$$

where $L_T : H_0^1(T) \rightarrow H^{-1}(T)$ denotes the restriction of the operator L to T .

In what follows we assume that, in the evaluation of the bubble term of the RFB formulation (2.3), the velocity field \mathbf{a} and the forcing term f can be treated as piecewise constant functions on \mathcal{T}_h .

As proved in [4], these assumptions ensure that the RFB method is equivalent to SDFEM. Indeed, in this case, the bubble part of the solution of the RFB formulation is given by

$$u_b|_T = (f - \mathbf{a} \cdot \nabla u_h)|_T b_T,$$

where b_T is the solution of the local problem

$$\begin{cases} -\varepsilon \Delta b_T + \mathbf{a} \cdot \nabla b_T = 1 & \text{in } T, \\ b_T = 0 & \text{on } \partial T. \end{cases}$$

(In other words, the bubble space is one-dimensional on every element). Thus, for any $T \in \mathcal{T}_h$, the static condensation of the bubble results in the classical streamline-diffusion stabilisation term, since

$$\begin{aligned} (u_b, L^* v_h)_T &= (L_T^{-1}(f - Lu_h)|_T, v_h)_T \\ &= -(f - \mathbf{a} \cdot \nabla u_h)|_T (\mathbf{a} \cdot \nabla v_h)|_T \int_T b_T d\mathbf{x} \\ &= \frac{\int_T b_T d\mathbf{x}}{|T|} \int_T (\mathbf{a} \cdot \nabla u_h - f)(\mathbf{a} \cdot \nabla v_h) d\mathbf{x} \\ &= \tau_T (\mathbf{a} \cdot \nabla u_h - f, \mathbf{a} \cdot \nabla v_h)_T, \end{aligned}$$

where the SD-parameter τ_T is defined via the bubble b_T as

$$\tau_T = \frac{1}{|T|} \int_T b_T d\mathbf{x}.$$

Moreover, assuming that $\varepsilon \ll |\mathbf{a}|$, i.e. that the equation is convection-dominated, we can approximate the integral average of the bubble b_T by integrating, instead, the solution \tilde{b}_T of the reduced problem:

$$\begin{cases} \mathbf{a} \cdot \nabla \tilde{b}_T = 1 & \text{in } T, \\ \tilde{b}_T = 0 & \text{on } \partial T_-, \end{cases}$$

where ∂T_- denotes the inflow boundary of T . Further, let us consider the approximation

$$\tau_T \approx \tilde{\tau}_T := \frac{1}{|T|} \int_T \tilde{b}_T d\mathbf{x} = \frac{h\mathbf{a}}{3|\mathbf{a}|}, \quad (3.1)$$

where $h\mathbf{a}$ is the length of the longest segment contained in T in the direction of \mathbf{a} .

The third term in the *a posteriori* error estimator $\tilde{\mathcal{E}}_1(u_h)$ defined in (2.7) can be approximated in exactly the same way, since

$$\begin{aligned} (\rho_T^3, \omega_T^3)_T &= (u_b, L^* z_h)_T = \tau_T (f - \mathbf{a} \cdot \nabla u_h, -\mathbf{a} \cdot \nabla z_h)_T \\ &\approx \tilde{\tau}_T |T| (\mathbf{a} \cdot \nabla u_h - f)|_T (\mathbf{a} \cdot \nabla z_h)|_T. \end{aligned}$$

We notice that the first line above coincides with the third term in the first error estimator for functionals defined in [16]. That is, in the special case in which the RFB method coincides with the streamline-diffusion method, our estimator $\tilde{\mathcal{E}}_1(u_h)$ is identical to that for the streamline-diffusion method.

Regarding the computation of $\|R(u_h)\|_{-1, \Omega}$ in $\bar{\mathcal{E}}_2(u_h)$ (see (2.18)), we may proceed as follows. We define the auxiliary problem

$$\begin{cases} -\Delta \phi = R(u_h) & \text{in } \Omega, \\ \phi = 0 & \text{on } \partial\Omega. \end{cases} \quad (3.2)$$

By definition of the norm of the dual space V' , we have

$$\begin{aligned} \|R(u_h)\|_{-1,\Omega} &= \sup_{\psi \in V} \frac{(R(u_h), \psi)}{\|\nabla\psi\|_{0,\Omega}} \\ &= \sup_{\psi \in V} \frac{(-\Delta\phi, \psi)}{\|\nabla\psi\|_{0,\Omega}} \\ &= \sup_{\psi \in V} \frac{(\nabla\phi, \nabla\psi)}{\|\nabla\psi\|_{0,\Omega}} \\ &= \|\nabla\phi\|_{0,\Omega}. \end{aligned}$$

Thus, we have reduced the problem of the computation of $\|R(u_h)\|_{-1,\Omega}$ to that of the computation of $\|\nabla\phi\|_{0,\Omega}$, where ϕ is the solution of (3.2). This will be done approximately, by considering (3.2) in weak form and solving it by a finite element method on the triangulation \mathcal{T}_h . That is, we will solve the problem

$$\begin{cases} \text{find } \phi_h \in V_h \text{ such that} \\ (\nabla\phi_h, \nabla v_h) = \langle R(u_h), v_h \rangle = (f, v_h) - \mathcal{L}(u_h, v_h) \quad \forall v_h \in V_h, \end{cases} \quad (3.3)$$

and compute $\|\nabla\phi_h\|_{0,\Omega}$.

At every iteration of the adaptive algorithm we need to refine the triangulation, basing the refinement on the marking strategy described in the previous section. In our numerical computations we utilize the MATLAB pde-toolbox refinement routine `refinemesh`, which performs *red refinement* on the marked elements. That is, every element marked for refinement is subdivided into four triangles (sons) by connecting the mid-points of the edges. To avoid the creation of hanging-nodes, and hence to comply with the constraint of admissibility of the triangulations given before, the neighbors of the marked elements are subjected to blue refinement following the longest edge bisection scheme [20], as shown in Figure 1. This technique is not ideal, since it involves refinement of distant neighbours, but it has the advantage of ensuring the shape-regularity of the triangulation.

Example 1. We consider the boundary value problem

$$\begin{cases} -\varepsilon\Delta u + u_x + u_y = f & \text{in } \Omega = (0, 1)^2, \\ u = 0 & \text{in } \partial\Omega, \end{cases} \quad (3.4)$$

with f defined in such a way that the exact solution is given by

$$u(x, y) = 2 \sin(x) y^2 (1 - e^{-(1-x)/\varepsilon}) (1 - e^{-(1-y)/\varepsilon}).$$

Our aim is the computation of the mean-flow over Ω , i.e.

$$J(u) = \int_{\Omega} u \, d\mathbf{x}, \quad (3.5)$$

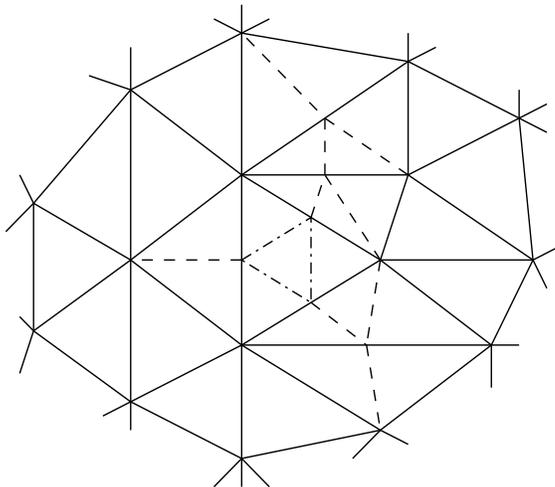


Figure 1: Example of red refinement (dot-dashed edges) and subsequent refinement of the neighbors (dashed edges). The rule for blue refinement is that the longest edge is always bisected.

with the aim to ensure that the error does not exceed a given tolerance TOL.

To start with, we compare the true error $|J(u) - J(u_h)|$ and the *a posteriori* error bound (2.19) on successively refined uniform meshes. We recall that the r.h.s. of (2.19) consists of two terms: $\tilde{\mathcal{E}}_1(u_h)$ which is related to the difference $z_H - z_h$, and $\bar{\mathcal{E}}_2(u_h)$ which is an upper bound for $\bar{\mathcal{E}}_1(u_h)$ which refers to $z - z_H$.

The results for $\varepsilon = 10^{-2}$ are shown in Figure 2(a): the log-log plot makes it apparent that $\bar{\mathcal{E}}_2(u_h)$ is over-estimating the true error. On the other hand, the term $\tilde{\mathcal{E}}_1(u_h)$ alone agrees remarkably well with the error, see also the effectivity indices reported in Table 1.

In order to conclude that $\tilde{\mathcal{E}}_1(u_h)$ can be used in an adaptive algorithm as an *a posteriori* error bound, we still need to ascertain its reliability, which depends on whether or not $\bar{\mathcal{E}}_1(u_h)$, i.e. the omitted term in (2.7), is of higher order.

We can compare $\tilde{\mathcal{E}}_1(u_h)$ and $\bar{\mathcal{E}}_1(u_h)$ for a slightly different problem, in which the target linear functional is chosen so that the dual solution z is known. This is achieved by performing the change of variables $(x, y) \rightarrow (1 - x, 1 - y)$ in the primal problem. Thus we define

$$z(x, y) = 2 \sin(1 - x)(1 - y)^2(1 - e^{-x/\varepsilon})(1 - e^{-y/\varepsilon}),$$

and fix the new target functional J^* consequently. The results obtained for this new problem are shown in Figure 2(b). The bound $\bar{\mathcal{E}}_2(u_h)$ is still over-estimating the error in approximately the same way as before, while we observe that (except on the coarsest grid) the sharper bound $\bar{\mathcal{E}}_1(u_h)$ is indeed of higher order than the true error.

We conclude that the adaptive algorithm described in Section 2.1, which uses $\tilde{\mathcal{E}}_1(u_h)$ as error estimator, is, for this problem at least, reliable and efficient.

We can further investigate the relative magnitudes of the terms in the *a posteriori*

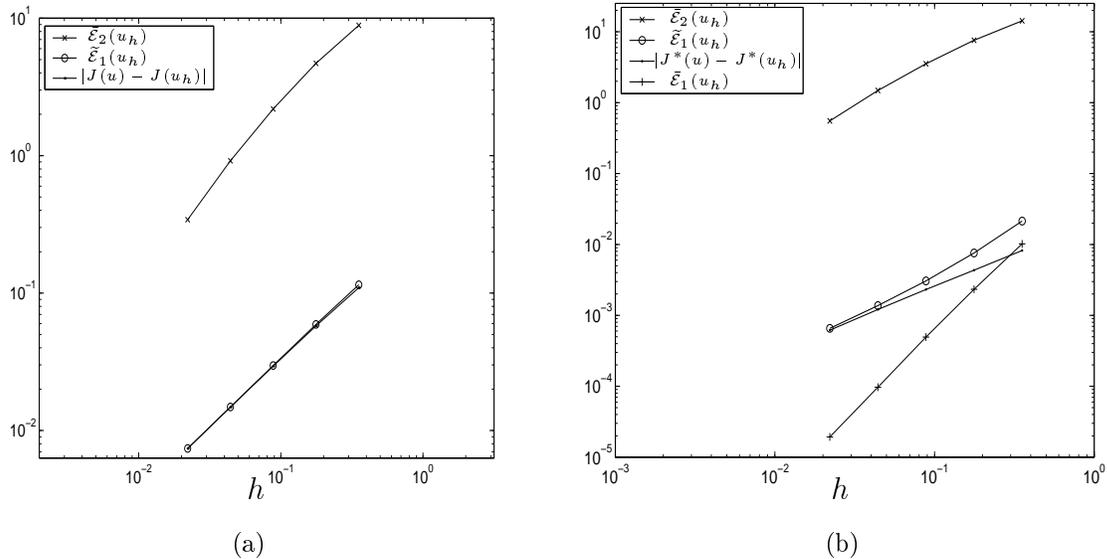


Figure 2: Example 1. The error and the *a posteriori* error bounds under successive uniform refinements with respect to the functional $J(\cdot)$ (a) and $J^*(\cdot)$ (b), with $\varepsilon = 10^{-2}$.

error bound by considering separately the three terms comprising $\tilde{\mathcal{E}}_1(u_h)$. Define

$$\begin{aligned}
 D_{\text{res}} &= \left| \sum_{T \in \mathcal{T}_h} (R_T(u_h), z_H - z_h)_T \right|, \\
 D_{\text{jump}} &= \left| \frac{1}{2} \sum_{T \in \mathcal{T}_h} (\varepsilon \mathbf{n} \cdot [\nabla u_h], z_H - z_h)_T \right|, \\
 D_{\text{bubl}} &= \left| \sum_{T \in \mathcal{T}_h} (u_b, L^* z_h)_T \right|.
 \end{aligned}$$

The behavior of these three terms can be appreciated from the numbers in Table 1 and

h	$ J(u) - J(u_h) $	$\tilde{\mathcal{E}}_1(u_h)$	eff	D_{bubl}	D_{res}	D_{jump}	$\tilde{\mathcal{E}}_2(u_h)$
1/4	1.09×10^{-1}	1.15×10^{-1}	1.05	1.08×10^{-1}	7.5×10^{-3}	5.8×10^{-4}	8.88
1/8	5.76×10^{-2}	5.91×10^{-2}	1.025	5.8×10^{-2}	1.4×10^{-3}	2.5×10^{-4}	4.69
1/16	2.93×10^{-2}	2.97×10^{-2}	1.01	2.95×10^{-2}	3.1×10^{-4}	9.0×10^{-5}	2.187
1/32	1.47×10^{-2}	1.48×10^{-2}	1.003	1.48×10^{-2}	7.7×10^{-5}	2.6×10^{-5}	0.917
1/64	7.39×10^{-3}	7.4×10^{-3}	1.001	7.39×10^{-3}	1.9×10^{-5}	6.3×10^{-6}	0.342

Table 1: Example 1. Convergence of $|J(u) - J(u_h)|$ and $\tilde{\mathcal{E}}_1(u_h)$ and its components, with $\varepsilon = 10^{-2}$.

the graphs in Figure 3. We observe that D_{res} and D_{jump} are of higher order and the true error is well approximated by the term D_{bubl} alone. Given that D_{bubl} is computable

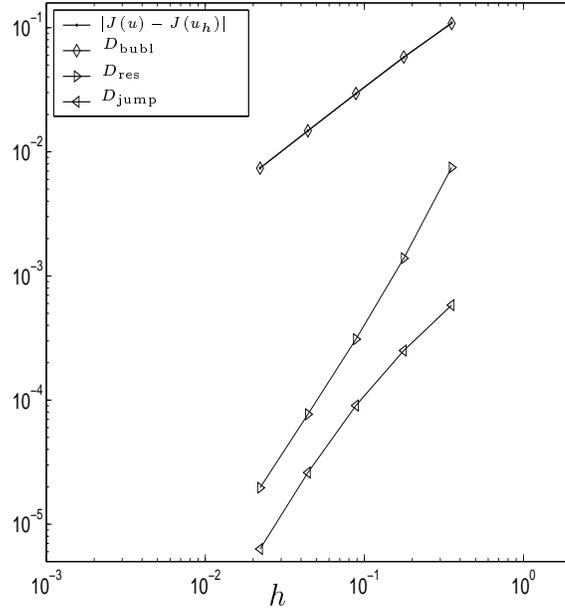


Figure 3: Example 1. The error and the three terms of the *a posteriori* error estimator $\tilde{\mathcal{E}}_1$ under successive uniform refinements with respect to the functional (3.5); $\varepsilon = 10^{-2}$.

(from u_b and z_h), this fact suggests to use D_{bubl} as a correction term by moving it across to the left-hand side of the error representation formula (this viewpoint is discussed in Giles & Süli [15]). In other words, the quantity

$$J_{\text{cor}}(u_h) = J(u_h) - \sum_{T \in \mathcal{T}_h} (u_b, L^* z_h)_T, \quad (3.6)$$

should give a better approximation to $J(u)$ than $J(u_h)$. The error representation formula now becomes:

$$J(u) - J_{\text{cor}} = \mathcal{L}(u - u_h, z - z_h).$$

For this we can rewrite the *a posteriori* error bound in the following form:

$$|J(u) - J_{\text{cor}}| \leq D_{\text{res}} + D_{\text{jump}} + \bar{\mathcal{E}}_2(u_h).$$

We know already that this error bound is not sharp, due to the third term on the right-hand side. But, this time the term $\bar{\mathcal{E}}_1(u_h)$ is no longer negligible in comparison with $D_{\text{res}} + D_{\text{jump}}$ alone. Indeed the quantity $D_{\text{res}} + D_{\text{jump}}$ under-estimates the true error, and so it cannot be used as an *a posteriori* error bound in an adaptive algorithm.

To show this, we have run the adaptive algorithm described in the previous section, using the following values of the parameters in the adaptive algorithm:

$$c_{\text{ref}} = .5, \quad p_{\text{ref}} = .1, \quad \text{TOL} = 10^{-3}. \quad (3.7)$$

The results are shown in Table 2, for $\varepsilon = 10^{-2}$ and $\varepsilon = 10^{-6}$. We observe that:

1. The error estimator $\tilde{\mathcal{E}}_1(u_h)$ is very effective in predicting the error $J(u) - J(u_h)$, robustly with respect to ε ;
2. The corrected quantity (3.6) gives, as expected, a considerably better approximation of the target quantity $J(u)$;
3. $D_{\text{res}} + D_{\text{jump}}$ under-estimates the true error, as indicated by the related effectivity index

$$eff_{\text{cor}} = \frac{D_{\text{res}} + D_{\text{jump}}}{|J(u) - J_{\text{cor}}(u_h)|}.$$

We conclude that the quantity (3.6) is best used as a more accurate approximation of $J(u)$ after the last step in an adaptive algorithm based on the error estimator $\tilde{\mathcal{E}}_1(u_h)$.

N_{el}	$ J(u) - J(u_h) $	$\tilde{\mathcal{E}}_1(u_h)$	eff	$ J(u) - J_{\text{cor}}(u_h) $	$D_{\text{res}} + D_{\text{jump}}$	eff_{cor}
32	1.095×10^{-1}	1.15×10^{-1}	1.05	1.1388×10^{-3}	8.05×10^{-3}	0.43
69	5.667×10^{-2}	5.93×10^{-2}	1.046	2.192×10^{-3}	1.62×10^{-3}	0.27
155	2.736×10^{-2}	2.935×10^{-2}	1.073	3.038×10^{-3}	1.05×10^{-3}	0.19
337	1.332×10^{-2}	1.499×10^{-2}	1.125	3.07×10^{-3}	1.39×10^{-3}	0.18
779	6.8×10^{-3}	7.45×10^{-3}	1.097	1.071×10^{-3}	4.09×10^{-4}	0.15
1680	3.543×10^{-3}	3.87×10^{-3}	1.092	6.017×10^{-4}	2.74×10^{-4}	0.14
4644	1.995×10^{-3}	2.144×10^{-3}	1.075	2.246×10^{-4}	7.55×10^{-5}	0.06

N_{el}	$ J(u) - J(u_h) $	$\tilde{\mathcal{E}}_1(u_h)$	eff	$ J(u) - J_{\text{cor}}(u_h) $	$D_{\text{res}} + D_{\text{jump}}$	eff_{cor}
32	1.119×10^{-1}	1.194×10^{-1}	1.067	2.44×10^{-3}	5.12×10^{-3}	2.1
69	5.825×10^{-2}	6.2×10^{-2}	1.064	2.84×10^{-3}	9.07×10^{-4}	0.32
166	2.835×10^{-2}	3.11×10^{-2}	1.099	2.94×10^{-3}	1.32×10^{-4}	0.045
345	1.376×10^{-2}	1.64×10^{-2}	1.192	2.64×10^{-3}	8.1×10^{-6}	0.003
763	6.99×10^{-3}	8.36×10^{-3}	1.193	1.3×10^{-3}	4.69×10^{-5}	0.036
1595	3.28×10^{-3}	4.1×10^{-3}	1.25	8.41×10^{-4}	1.93×10^{-5}	0.023
3296	1.648×10^{-3}	2.03×10^{-3}	1.234	3.98×10^{-4}	1.28×10^{-5}	0.032
6552	7.514×10^{-4}	1.02×10^{-3}	1.357	2.92×10^{-4}	2.35×10^{-5}	0.08

Table 2: Example 1. The error under successive refinements with respect to the functional $J(u) = \int_{\Omega} u \, dx$; $\varepsilon = 10^{-2}$ (above) and $\varepsilon = 10^{-6}$ (below).

Repeating the same experiment for the modified target functional J^* we have observed that the terms in $\tilde{\mathcal{E}}_1(u_h)$ are quantitatively comparable to D_{res} and D_{jump} , and hence they cannot be neglected.

Example 2. We consider the boundary value problem

$$\begin{cases} -\varepsilon \Delta u + u_x + u_y = f & \text{in } \Omega = (0, 1)^2, \\ u = 0 & \text{on } \partial\Omega, \end{cases} \quad (3.8)$$

N_{el}	$ J(u) - J(u_h) $	$\tilde{\mathcal{E}}_1(u_h)$	eff	N_{el}	$ J(u) - J(u_h) $	$\tilde{\mathcal{E}}_1(u_h)$	eff
32	2.12×10^{-2}	9.17×10^{-2}	4.3	32	2.58×10^{-2}	1.28×10^{-1}	4.98
64	2.5×10^{-3}	8.7×10^{-3}	3.5	82	1.02×10^{-3}	1.13×10^{-3}	1.1
146	1.7×10^{-3}	2.61×10^{-3}	1.5	188	3.77×10^{-3}	4.61×10^{-3}	1.2
290	8.78×10^{-4}	1.24×10^{-3}	1.4	378	2.14×10^{-4}	6.52×10^{-3}	3.04
616	2.1×10^{-4}	4.06×10^{-4}	1.9	708	5.08×10^{-4}	2.09×10^{-3}	4.1
1396	7.43×10^{-5}	1.24×10^{-4}	1.6	1262	2.0×10^{-4}	7.04×10^{-4}	3.5
				2176	1.19×10^{-4}	3.2×10^{-4}	2.7
				3858	9.39×10^{-5}	2.82×10^{-4}	3

Table 3: Example 2. The error under successive refinements with respect to the functional $J(u) = u((.49, .49))$; $\varepsilon = 10^{-2}$ (left) and $\varepsilon = 10^{-6}$ (right).

with f defined in such a way that the exact solution is given by

$$u(x, y) = 2xy(1 - e^{-(1-x)/\varepsilon})(1 - e^{-(1-y)/\varepsilon}).$$

Notice that the function u is symmetric with respect to the line $x = y$.

The objective is the computation of the solution at a given point $P = P(x_0, y_0)$ with the aim to ensure that the error between $u(x_0, y_0)$ and $u_h(x_0, y_0)$ does not exceed a given tolerance TOL. We apply the algorithm described in Section 2.1 using the same parameter values as in (3.7).

The successive meshes produced by the algorithm to calculate the solution at $P = (.49, .49)$ with $\varepsilon = 10^{-2}$ and $\varepsilon = 10^{-6}$ are depicted in Figure 6 and Figure 7, respectively.

The meshes respect the symmetry of the problem. Moreover, we notice that initially the mesh is refined down-wind of P : the algorithm recognizes that some resolution of the boundary-layers is necessary in order to ensure any accuracy at the point of interest. In subsequent refinement the boundary layer zone is left unchanged, the refinements being concentrated upwind of the point, along the subcharacteristic curve passing through P .

The effectivity of the *a posteriori* error estimator $\tilde{\mathcal{E}}_1(u_h)$ is reported in Table 3: the estimator is robust with respect to the diffusion parameter.

Example 3. We solve the boundary value problem with discontinuous boundary conditions

$$\left\{ \begin{array}{l} -\varepsilon \Delta u + (\cos(\pi/3), \sin(\pi/3))^T \cdot \nabla u = 1 \quad \text{in } \Omega = (0, 1)^2, \\ u = 1 \quad \text{for } \left\{ \begin{array}{l} x \leq 1/2, \\ x = 0 \end{array} \right. \quad y = 0, \\ u = 0, \quad \text{otherwise.} \end{array} \right. \quad (3.9)$$

The solution of this problem has an internal layer propagating across Ω from the discontinuity in the boundary condition at $(0.5, 0) \in \partial\Omega$. As for Example 1, we defined the target functional to be the mean-flow in the entire domain Ω ; that is $J(u) = \int_{\Omega} u \, d\mathbf{x}$ and fixed $\varepsilon = 10^{-6}$. The algorithm is again the one described in Section 2.1, which employs

the computable error estimator $\tilde{\mathcal{E}}_1(u_h)$. The values for the parameters in the adaptive algorithm are:

$$c_{\text{ref}} = .5, \quad p_{\text{ref}} = .05, \quad \text{TOL} = 10^{-3}.$$

As we can see in Figure 8, first the mesh gets refined in the boundary layer, which is the major source of error. Only when the boundary layer has been partially resolved, is the mesh refined along the internal layer, showing that the indicator correctly identifies locations in the mesh that most affect the accuracy of the approximation of the functional. The final mesh consists of 7405 triangles, approximately 5 out of 7 of which are located in the proximity of the boundary layer given by $y > 7/8$.

Example 4. We consider the mixed boundary–value problem for the convection–diffusion equation specified in Figure 4 (top left). Homogeneous Neumann boundary condition is imposed on $\Gamma_N = \{(x, y) \in \Gamma : y = 0 \text{ or } x = 1\}$, while on $\Gamma_D = \partial\Omega \setminus \Gamma_N$ a Dirichlet boundary–condition is given.

The objective is to evaluate the mean–flow over the Neumann boundary. That is,

$$J(u) = \int_{\Gamma_N} u \, dx.$$

The corresponding dual–problem is given by

$$\begin{cases} -\varepsilon\Delta z - \mathbf{a} \cdot \nabla z = 0 & \text{in } \Omega = (0, 1)^2 \\ z = 0 & \text{on } \Gamma_D \\ \varepsilon \mathbf{n} \cdot \nabla z + \mathbf{n} \cdot \mathbf{a} z = 1 & \text{on } \Gamma_N. \end{cases} \quad (3.10)$$

This time, the *a posteriori* analysis proceeds as follows:

$$\begin{aligned} J(u) - J(u_h) &= J(u - u_h) \\ &= \mathcal{L}(u - u_h, z) \\ &= \mathcal{L}(u - u_h, z - z_h) + \sum_{T \in \mathcal{T}_h} (u_b, L^* z_h)_T \\ &= \sum_{T \in \mathcal{T}_h} ((f - Lu_h, z - z_h)_T - (\varepsilon \mathbf{n}_T \cdot \nabla u_h, z - z_h)_{\partial T \cap (\Omega \cup \Gamma_N)}) \\ &\quad + \sum_{T \in \mathcal{T}} (u_b, L^* z_h)_T \\ &= \sum_{T \in \mathcal{T}_h} \left((R_T(u_h), z - z_h)_T - \frac{1}{2} (\varepsilon \mathbf{n}_T \cdot [\nabla u_h], z - z_h)_{\partial T \cap \Omega} \right. \\ &\quad \left. - (\varepsilon \mathbf{n}_T \cdot \nabla u_h, z - z_h)_{\partial T \cap \Gamma_N} + (u_b, L^* z_h)_T \right), \end{aligned}$$

with the new term related to the presence of the Neumann boundary.

The results obtained by using the error estimator $\tilde{\mathcal{E}}_1$, modified to include the Neumann term, are shown in Figure 4. We notice that the refinement is mainly concentrated

along the internal layer in the primal solution, but some refinement is performed where features of the dual solution are present. This should be compared with the output of mesh adaptation driven by norm error examined in the next section.

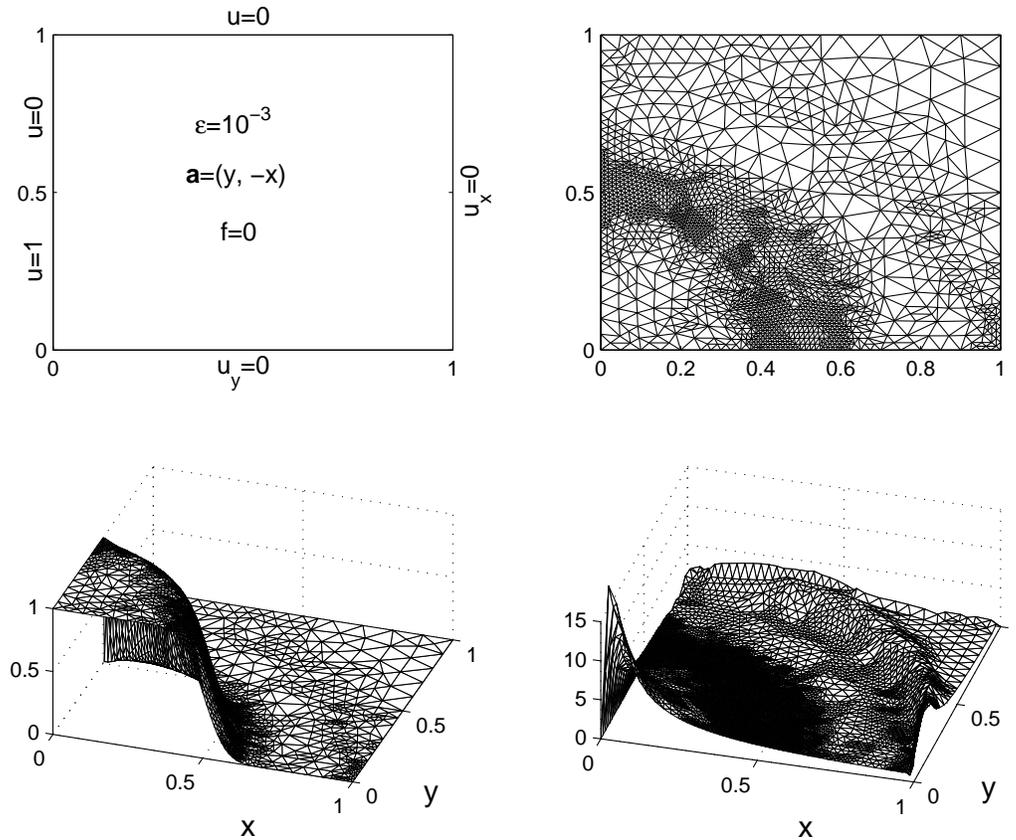


Figure 4: Example 4. Above: problem specifications (left) and mesh after five refinements (right) for $J(u) = \int_{\Gamma_N} u dx$ and $\varepsilon = 10^{-3}$. Below: The corresponding solution (left) and dual solution (right).

4 The bubble as error estimator

In this section we discuss *a posteriori* error estimation with respect to the energy norm on shape regular triangulations.

We consider the following model problem with mixed Dirichlet and Neumann bound-

ary conditions:

$$\begin{cases} -\varepsilon\Delta u + \mathbf{a} \cdot \nabla u = f & \text{in } \Omega, \\ u = 0 & \text{on } \Gamma_D, \\ \varepsilon \mathbf{n} \cdot \nabla u = g & \text{on } \Gamma_N, \end{cases} \quad (4.1)$$

with $\partial\Omega = \Gamma_D \cap \Gamma_N$ and $\Gamma_D \cup \Gamma_N = \emptyset$, assuming that Γ_D is closed and has nonzero measure.

Let \mathcal{T}_h , $h > 0$ be a family of triangulations of Ω such that the following conditions hold.

1. *Conformity*: Any two triangles in \mathcal{T}_h either have a common edge or common vertex, or they do not intersect at all;
2. *Consistency*: Any triangle–edge lying on the boundary is contained either in Γ_D or in Γ_N ;
3. *Shape regularity*: For any triangle, the ratio of the largest circumscribed circle to that of the smallest inscribed circle is bounded by a constant which does not depend on the triangle and on h .

As we have mentioned in Section 2, the bubble part of the RFB finite element solution is given element–wise as the function $u_b|_T \in H_0^1(T)$ satisfying

$$-\varepsilon\Delta u_b|_T + \mathbf{a} \cdot \nabla u_b|_T = (f - Lu_h)|_T \quad \forall T \in \mathcal{T}_h. \quad (4.2)$$

Thus, we can bound the L_2 –norm of the bubble u_b using the following stability result for the convection–diffusion equation.

Let ω be a bounded Lipschitz domain in \mathbb{R}^2 and let $w \in H_0^1(\omega)$ be such that

$$\begin{cases} -\varepsilon\Delta w + \mathbf{a} \cdot \nabla w = f & \text{in } \omega, \\ w = 0 & \text{on } \partial\omega, \end{cases}$$

where $\varepsilon > 0$, $f \in L_2(\omega)$ and $\mathbf{a} \in (C^1(\omega))^2$ satisfies

$$\begin{cases} \nabla \cdot \mathbf{a} \leq 0 & \text{in } \omega, \\ \mathbf{a} \text{ has no closed integral curves in } \bar{\omega}. \end{cases}$$

Then, according to [18], there exist a constant C dependent only on ω and on \mathbf{a} such that

$$\varepsilon^{1/2} \|w\|_{1,\omega} + \|w\|_{0,\omega} \leq C \|f\|_{0,\omega}. \quad (4.3)$$

The following result is mentioned by Russo in [21] where the case of constant coefficients is treated.

Proposition 4.1 *Let $u_b|_T \in H_0^1(T)$ be the solution of (4.2). If $\nabla \cdot \mathbf{a} \leq 0$ and \mathbf{a} has no closed integral curves in T , then*

$$\|u_b\|_{0,T} \leq Ch_T \|f - \mathbf{a} \cdot \nabla u_h\|_{0,T}, \quad (4.4)$$

where C is independent of ε and h_T .

Proof. Upon scaling problem (4.2) by using the transformation $A : (x, y) \rightarrow h_T(\xi, \eta)$ we deduce that the image of $u_b|_T$, say $\hat{u}_b \in H_0^1(\hat{T})$, where $\hat{T} = A(T)$, satisfies

$$-\frac{\varepsilon}{h_T} \Delta \hat{u}_b(\xi, \eta) + \mathbf{a} \cdot \nabla \hat{u}_b(\xi, \eta) = h_T (f - \mathbf{a} \cdot \nabla u_h)(x(\xi, \eta), y(\xi, \eta)).$$

To obtain (4.4), we scale the stability estimate (4.3) applied to \hat{u}_b :

$$\|u_b\|_{0,T} \leq Ch_T \|\hat{u}_b\|_{0,\hat{T}} \leq Ch_T^2 \|(f - \mathbf{a} \cdot \nabla u_h) \circ A^{-1}\|_{0,\hat{T}} \leq Ch_T \|f - \mathbf{a} \cdot \nabla u_h\|_{0,T},$$

where C depends on \mathbf{a} and the shape regularity of T . \square

This result permits us to repeat, for the RFB formulation with linear finite elements, the *a posteriori* analysis carried out by Verfürth [24] for the SUPG method.

From the coercivity of \mathcal{L} we have

$$\varepsilon^{1/2} |u - u_h|_{1,\Omega} \leq \sup_{v \in V \setminus \{0\}} \frac{\mathcal{L}(u - u_h, v)}{\varepsilon^{1/2} |v|_{1,\Omega}}.$$

Now let $v \in V$ be such that $\varepsilon^{1/2} |v|_{1,\Omega} = 1$. For any such v we have

$$\varepsilon^{1/2} |u - u_h|_{1,\Omega} \leq \mathcal{L}(u - u_h, v - P_h v) + \mathcal{L}(u - u_h, P_h v) = I + II.$$

Let $\mathcal{E}_{h,N}$ be the set of all edges contained in the Neumann boundary Γ_N . To bound the first term on the right-hand side we employ Green's formula element-wise:

$$\begin{aligned} I &= \sum_{T \in \mathcal{T}_h} \left((f - Lu_h, v - P_h v)_T - \frac{1}{2} (\varepsilon \mathbf{n}_T \cdot [\nabla u_h], v - P_h v)_{\partial T \cap \Omega} \right) \\ &\quad + \sum_{E \in \mathcal{E}_{h,N}} (g - \varepsilon \mathbf{n} \cdot \nabla u_h, v - P_h v)_E \\ &\leq \sum_{T \in \mathcal{T}_h} \left(\|f - Lu_h\|_{0,T} \|v - P_h v\|_{0,T} + \frac{1}{2} \|\varepsilon \mathbf{n}_T \cdot [\nabla u_h]\|_{0,\partial T \cap \Omega} \|v - P_h v\|_{0,\partial T \cap \Omega} \right) \\ &\quad + \sum_{E \in \mathcal{E}_{h,N}} \|g - \varepsilon \mathbf{n} \cdot \nabla u_h\|_{0,E} \|v - P_h v\|_{0,E} \\ &\leq C \left(\sum_{T \in \mathcal{T}_h} \left(h_T^2 \varepsilon^{-1} \|f - Lu_h\|_{0,T}^2 + \frac{1}{2} h_T \varepsilon \|\mathbf{n}_T \cdot [\nabla u_h]\|_{0,\partial T \cap \Omega}^2 \right) \right. \\ &\quad \left. + \sum_{E \in \mathcal{E}_{h,N}} h_T \varepsilon^{-1} \|g - \varepsilon \mathbf{n} \cdot \nabla u_h\|_{0,E}^2 \right)^{1/2}, \end{aligned} \quad (4.5)$$

having made use of the Cauchy–Schwarz inequality, the interpolant approximation properties (2.12) and (2.13) applied on T and the shape regularity of \mathcal{T}_h .

To bound II , we notice that since u is the solution of (2.1), u_h solves (2.3) and $\Delta P_h v = 0$, we have

$$\begin{aligned}
II &= \sum_{T \in \mathcal{T}_h} (u_b, L^* P_h v)_T \\
&= - \sum_{T \in \mathcal{T}_h} (u_b, \mathbf{a} \cdot \nabla P_h v)_T \\
&\leq \sum_{T \in \mathcal{T}_h} \|u_b\|_{0,T} \|\mathbf{a} \cdot \nabla P_h v\|_{0,T} \\
&\leq C \sum_{T \in \mathcal{T}_h} h_T \varepsilon^{-1/2} \|f - \mathbf{a} \cdot \nabla u_h\|_{0,T} \|\mathbf{a}\|_{\infty,T} \varepsilon^{1/2} |P_h v|_{1,T} \\
&\leq C \left\{ \sum_{T \in \mathcal{T}_h} \|\mathbf{a}\|_{\infty,T}^2 h_T^2 \varepsilon^{-1} \|f - \mathbf{a} \cdot \nabla u_h\|_{0,T}^2 \right\}^{1/2} \tag{4.6}
\end{aligned}$$

thanks to (4.4), (2.14) and the shape regularity of \mathcal{T}_h .

Thus, from (4.5) and (4.6) we conclude that the following error bound holds:

$$\varepsilon^{1/2} |u - u_h|_{1,\Omega} \leq C \mathcal{E}_S, \tag{4.7}$$

with the error estimator \mathcal{E}_S given by

$$\begin{aligned}
\mathcal{E}_S^2 &:= \sum_{T \in \mathcal{T}_h} \left(h_T^2 \varepsilon^{-1} \|f - Lu_h\|_{0,T}^2 + \frac{1}{2} h_T \varepsilon \|\mathbf{n} \cdot [\nabla u_h]\|_{0,\partial T \cap \Omega}^2 \right) \\
&+ \sum_{E \in \mathcal{E}_{h,N}} \|\mathbf{a}\|_{\infty,T}^2 h_T \varepsilon^{-1} \|g - \varepsilon \mathbf{n} \cdot \nabla u_h\|_{0,E}^2.
\end{aligned}$$

Remark. The *a posteriori* error analysis in both the $\varepsilon^{1/2}$ -weighted H^1 -seminorm and the L_2 -norm error of the RFB method based on general finite elements (i.e. not necessarily linear) was carried out by Sangalli [22] under the hypothesis that the vector field \mathbf{a} has no closed integral curves on the whole domain $\overline{\Omega}$. Sangalli's result, though, applies to the component of the RFB solution in the space

$$\widetilde{W}_h = \{v \in V_{\text{RFB}} : L^* v = 0 \text{ in each element } T \in \mathcal{T}_h\},$$

instead of the piecewise polynomial component as is the case of our bound.

Under such a hypothesis on \mathbf{a} and again restricting ourselves to the case of linear finite elements, we can repeat Sangalli's argument, this time applied to the polynomial component of the solution by employing (4.4). In this way the error bound (4.7) follows, with the extra control on the L_2 -norm error.

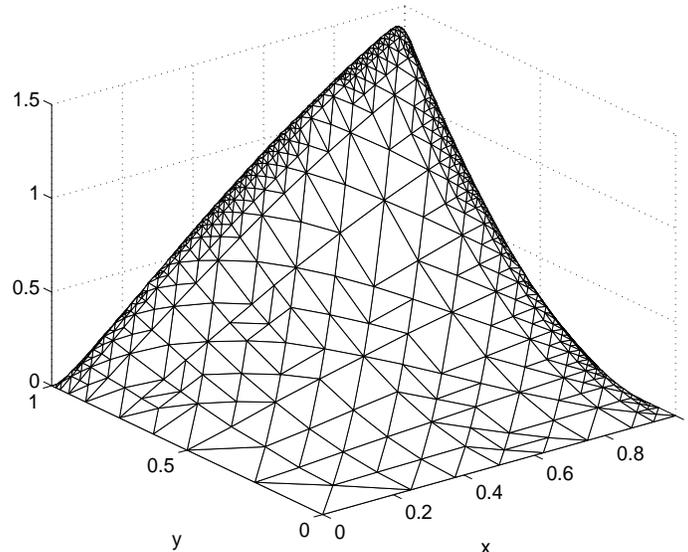


Figure 5: Example 5. The final solution produced by the algorithm based on the \mathcal{E}_S error indicator.

Example 5. We consider again the boundary-value problem of Example 1 of the previous section. The outcome of the implementation of the error estimator (4.7) in an adaptive algorithm is displayed in Table 4 (top). The quantities S_i and S_b reported in the table are the two terms in the error estimator which are relevant to this problem, i.e.

$$S_i^2 = \sum_{T \in \mathcal{T}_h} h_T^2 \varepsilon^{-1} \|f - Lu_h\|_{0,T}^2,$$

$$S_b^2 = \frac{1}{2} \sum_{T \in \mathcal{T}_h} h_T \varepsilon \|\mathbf{n} \cdot [\nabla u_h]\|_{0,\partial T \cap \Omega}^2.$$

In the computation the marking parameters and the tolerance were set to

$$c_{\text{ref}} = .3, \quad p_{\text{ref}} = .1, \quad \text{TOL} = 0.1.$$

The final solution and mesh can be seen in Figure 5. The intermediate meshes are similar to those shown in Figure 10 below.

We have repeated the same experiment but using as error indicator the L_2 -norm of the bubble part of the RFB solution, which we know to be a representation of the internal residual. Thus, we define the new indicator

$$\mathcal{E}_B^2 = \sum_{T \in \mathcal{T}_h} \|\mathbf{a}\|_{\infty,T}^2 \varepsilon^{-1} \|u_b\|_{0,T}^2.$$

The results obtained using this error indicator are shown in Table 4 (bottom). Of course, \mathcal{E}_B is reliable only as long as the internal residual term S_i dominates the edge

residual term S_b . From the table we observe that this may be true initially, but not asymptotically. Indeed, it has been proved by Kunert and Verfürth [17] and Carstensen and Verfürth [7] that edge residuals *dominate* the error of the finite element method in the case of low order finite elements, by showing that the edge residuals yield global upper and local lower bounds on the error, both in the H^1 - and the L_2 -norm.

So \mathcal{E}_B cannot alone be considered as a reliable *a posteriori* error bound. However, it gives a more effective estimate of the internal residual than S_i ; see the effectivities reported in Table 4.

Alternatively, \mathcal{E}_B may be used to decide wherever to locally turn off the stabilisation term. This idea is developed in the next section.

Example 6. Our final example of this section regards the mixed boundary value problem Example 4 in the previous section. The problem data are specified in Figure 4 (above-left): a discontinuity in the Dirichlet boundary condition at $(x, y) = (0, 1/2)$ is propagated (and smoothed) inside the domain and exits through the Neumann boundary.

Figure 9 shows the successive refinements obtained using the bubble error indicator \mathcal{E}_B , with the tolerance set to $\text{TOL} = 0.05$. The solution on the final mesh is also shown.

Refinement is limited to the areas with relatively strong variations of ∇u . Figure 9 should be compared with Figure 4 which reports the triangulation obtained by solving the same boundary-value problem but with the refinement driven by a linear functional of the solution as target.

5 *hb*-adaptivity

We have examined adaptive mesh refinement algorithms for the stabilised Galerkin finite element formulation (2.3) derived from the RFB method. The stabilisation term in (2.3) depends on the bubble part of the solution u_b . Using the assumption of local constant coefficients, we have been able to reduce the complex task of the evaluation of such term to that of the evaluation of an average of the bubble. Further, the approximation (3.1) led to the even simpler task of the evaluation of the elemental diameter in the direction of the convective field. As a consequence, in terms of computational complexity, the resulting method in the case of piecewise linear elements is equivalent to the SDFEM, with the advantage that the stabilisation parameter is given by the method.

This way of proceeding, of course, has some limitations. The approximation (3.1) may not be sufficiently accurate if the coefficients in the p.d.e. cannot be treated as piecewise constant functions or if the mesh is sufficiently refined. One may also wish to compute the bubbles more accurately, for example to include an edge-stabilisation of the sort discussed in [5]. Finally, it is clear that in parts of the computational domain where the mesh has been sufficiently refined or where the solution is relatively flat, there is no reason for stabilising in the first place.

This justifies the idea of including in the mesh adaptation algorithms discussed so far (*h-refinement*) an automatic way of ‘turning off’ the stabilising term wherever this is no more required (*b-derefinement*).

N_{el}	$\ u - u_h\ _{0,\Omega}$	$\varepsilon^{1/2} u - u_h _{1,\Omega}$	$\mathcal{E}_S(u_h)$	S_i	S_b	eff_{l2}	eff_{en}
32	0.29	0.9	10.4	10.4	0.44	35.9	11.5
69	0.19	0.83	7.13	7.1	0.58	36.3	8.6
155	0.12	0.74	4.71	4.6	0.66	39.1	6.3
329	0.06	0.59	2.99	2.9	0.69	45	5
711	0.034	0.4	1.83	1.7	0.62	52.9	4.6
1501	0.018	0.24	1.1	0.1	0.5	58.8	4.6
3706	0.01	0.13	0.66	0.5	0.35	66.5	4.8
11272	0.005	0.07	0.38	0.3	0.22	72.7	5.06

N_{el}	$\ u - u_h\ _{0,\Omega}$	$\varepsilon^{1/2} u - u_h _{1,\Omega}$	$\mathcal{E}_B(u_h)$	eff_{l2}	eff_{en}	S_i	S_b
32	0.29	0.9	3.01	10.3	3.3	10.4	0.44
69	0.19	0.83	2.03	10.3	2.4	7.1	0.58
143	0.12	0.7	1.33	10.9	1.79	4.7	0.67
310	0.068	0.59	0.83	12.2	1.4	3.03	0.69
650	0.036	0.4	0.49	13.6	1.21	1.8	0.62
1358	0.019	0.24	0.27	14.1	1.12	1.04	0.51
3137	0.011	0.14	0.15	14.2	1.07	0.6	0.37
9367	0.0058	0.08	0.08	14.4	1.02	0.3	0.24
28705	0.0032	0.04	0.04	14.6	0.98	0.19	0.15

Table 4: Example 5. Error over successive refinements in the L_2 and $\varepsilon^{1/2}$ -weighted H^1 -seminorm (energy norm); $\varepsilon = 10^{-2}$. \mathcal{E}_S indicator (above) and bubble indicator \mathcal{E}_B (below).

To address the crucial issue of when and where to phase out the bubble stabilisation alongside the h -refinement process, we may use the particular residual term in the *a posteriori* error bound which is related to the stabilising term itself.

Let us consider, in particular, the $\varepsilon^{1/2}$ -weighted H^1 -seminorm error bound \mathcal{E}_S in (4.7), discarding, for simplicity, any term related to Neumann boundary conditions. Looking back to (4.6) we see that \mathcal{E}_S in fact consists of three terms, S_i , S_b and \mathcal{E}_b , related to the internal residual, the boundary jump and the stabilising term, respectively. Let S_i^T , S_b^T and \mathcal{E}_b^T be the local components of these terms, that is,

$$\begin{aligned}
S_i^T &= h_T \varepsilon^{-1/2} \|f - (-\varepsilon \Delta u_h + \mathbf{a} \cdot \nabla u_h)\|_{0,T} \\
S_b^T &= \left(\frac{h_T \varepsilon}{2}\right)^{1/2} \|\mathbf{n} \cdot [\nabla u_h]\|_{0,\partial T \cap \Omega} \\
\mathcal{E}_b^T &= \|\mathbf{a}\|_{\infty,T} \varepsilon^{-1/2} \|u_b\|_{0,T}.
\end{aligned}$$

We have seen that the term \mathcal{E}_b^T is bounded by S_i^T . Moreover, we expect the term \mathcal{E}_b^T to be relatively small wherever stabilisation is not crucial. Hence we use the relative magnitude of \mathcal{E}_b^T with respect to S_i^T and S_b^T as an indicator for b -derefinement.

Let

$$\eta_T = S_i^T + S_b^T,$$

be the local error indicator and let $\bar{\eta}$ be the maximum of η_T after cutting the upper 10% or 5% of the values. Finally, let $c_{\text{ref}} \in (0, 1)$ and $c_b > 0$ be some user-selected threshold parameters and TOL a given tolerance.

We propose the following algorithm, which is a modification of the one defined in Section 2.1:

1. Define an initial mesh;
2. Calculate u_h on the current mesh;
3. Check the stopping criterion: IF $\mathcal{E}_S \leq \text{TOL}/C$ then STOP;
4. Apply the h -refinement criterion: refine those elements whose error estimator $\mathcal{E}_S^T = S_i^T + S_b^T$ exceeds $c_{\text{ref}}\bar{\eta}$;
5. Apply the b -derefinement criterion: IF $\mathcal{E}_b^T < c_b (S_i^T + S_b^T)$, turn off the stabilisation term on T or its newly defined sons and GOTO 2.

Since we are not assuming any *a priori* knowledge about the behavior of the solution, the algorithm is started up with the stabilising term turned on everywhere. Later on, the fraction of elements selected for b -derefinement depends on the threshold parameter c_b . Its ‘correct’ value should be expected to be subject to the relative sharpness of the different components of the error bound. In both examples below the value $c_b = 0.1$ produced satisfactory results.

Example 7. It is instructive to apply the hb -adaptive algorithm to Example 5 above. We recall that we are solving:

$$\begin{cases} -\varepsilon\Delta u + u_x + u_y = f & \text{in } \Omega = (0, 1)^2, \\ u = 0 & \text{in } \partial\Omega. \end{cases}$$

The exact solution exhibits a boundary layer at the outflow boundary $x = 1$ and $y = 1$ (see Figure 5). Ideally, stabilisation should be employed only in the layer. The sequence of meshes produced is shown in Figure 10: the shadowed elements are those where stabilisation is turned on. We notice that stabilisation is soon removed away from the layer and, when this is resolved, also inside it.

The effectiveness of the new algorithm can be appreciated from the numbers displayed in Table 5: the outputs of the old algorithm are reproduced on the left (cf. Table 4), while, on the right, we present those of the hb -refinement algorithm. Similar accuracy is achieved with comparable meshes. The new algorithm uses more elements while removing the stabilisation from the layer. On the other hand, the final mesh, which is layer-resolving, is slightly more effective since the same accuracy is obtained with 32

N_{el}	$\varepsilon^{1/2} u - u_h _{1,\Omega}$	$\mathcal{E}_S(u_h)$	eff_{en}	N_{el}	$\varepsilon^{1/2} u - u_h _{1,\Omega}$	$\mathcal{E}_S(u_h)$	eff_{en}
32	0.9	10.4	11.5	32	0.9	10.4	11.5
69	0.83	7.13	8.6	69	0.83	7.13	8.6
155	0.74	4.71	6.3	155	0.74	4.86	6.5
329	0.59	2.99	5	332	0.59	3.15	5.3
711	0.4	1.83	4.6	727	0.39	1.92	4.8
1501	0.24	1.1	4.6	1523	0.22	1.16	5.2
3706	0.13	0.66	4.8	3808	0.123	0.67	5.4
11272	0.07	0.38	5.06	11240	0.067	0.38	5.7

Table 5: Example 5. Error and \mathcal{E}_S indicator over successive refinements in the $\varepsilon^{1/2}$ -weighted H^1 -seminorm (energy norm); $\varepsilon = 10^{-2}$. We show the output of the h -refinement algorithm (left) and hb -algorithm (right).

elements less. Over all, if the parameter c_b is appropriately tuned, the numerical solution is not corrupted and the algorithm is robust.

Example 8. To conclude, we test the hb -refinement algorithm on a problem whose solution exhibits an internal layer, namely Example 6 above. The sequence of mesh refinements is depicted in Figure 11. Again the shadowed elements are those where stabilisation is present. To highlight more clearly such elements, the final mesh is plotted a second time at the bottom-right of the figure with the stabilised elements lifted out of the xy -plane. The meshes are very similar to those obtained without b -derefinement, cf Figure 9: this may have to do with the fact that the layer in the solution of this problem is less severe than the one of Example 7 above.

6 Conclusions

We have developed an h -adaptive algorithm that is driven by a residual-based *a posteriori* error bound. The error is represented in terms of the residual of the finite element approximation weighted by the solution of the dual problem. The numerical approximation of linear functionals of the solution as well as energy norm error estimation have been considered.

We have shown that the elimination of the dual solution from the *a posteriori* bound, via strong stability estimates, leads to crude bounds. For this reason, the algorithm proposed is based on the so called Type I error bounds, i.e. as few as possible steps are performed in the bounding of the error. The downside of this approach is that the dual solution is not eliminated from the bound and needs to be approximated in order to obtain a computable error bound. For the error bound to be reliable, it is necessary that the dual problem is solved accurately. The approach adopted here has been to use a dual mesh with mesh size half of that of the primal mesh. On the examples considered

the algorithm was reliable and effective.

The *a posteriori* analysis for the energy norm error of the RFB method presented here is similar to that of the SUPG method performed by Verfürth [24]. The error bound obtained consists of three terms: the two classical residual-based terms of the Galerkin formulation, i.e. internal residual and boundary jump of the gradient, plus a third term due to static condensation of the bubbles. We have shown how the latter can be bounded in terms of the internal residual using an appropriate stability result applied to the bubble part of the solution.

We also introduced a new *hb*-adaptive algorithm in which, to avoid the evaluation of the bubble part of the solution where this is not crucial, the bubble stabilisation is phased out locally depending on the relative magnitude of the terms in the *a posteriori* error bound mentioned above.

References

- [1] BECKER, R., AND RANNACHER, R. A feed-back approach to error control in finite element methods: basic analysis and examples. *East-West J. Numer. Math.* 4, 4 (1996), 237–264.
- [2] BRENNER, S. C., AND SCOTT, L. R. *The mathematical theory of finite element methods*. Springer-Verlag, New York, 1994.
- [3] BREZZI, F., MARINI, D., AND SÜLI, E. Residual-free bubbles for advection-diffusion problems: the general error analysis. *Numer. Math.* 85, 1 (2000), 31–47.
- [4] BREZZI, F., AND RUSSO, A. Choosing bubbles for advection-diffusion problems. *Math. Models Methods Appl. Sci.* 4, 4 (1994), 571–587.
- [5] CANGIANI, A., AND SÜLI, E. Enhanced RFB method. *Oxford University Computing Laboratory research report no. 03/17* (2003), submitted for publication in *Numer. Math.*
- [6] CANUTO, C., RUSSO, A., AND VAN KEMENADE, V. Stabilized spectral methods for the Navier-Stokes equations: residual-free bubbles and preconditioning. *Comput. Methods Appl. Mech. Engrg.* 166, 1-2 (1998), 65–83.
- [7] CARSTENSEN, C., AND VERFÜRTH, R. Edge residuals dominate a posteriori error estimates for low order finite element methods. *SIAM J. Numer. Anal.* 36, 5 (1999), 1571–1587 (electronic).
- [8] CIARLET, P. G. *The finite element method for elliptic problems*. North-Holland Publishing Co., Amsterdam, 1978. Studies in Mathematics and its Applications, Vol. 4.

- [9] CLÉMENT, P. Approximation by finite element functions using local regularization. *Rev. Française Automat. Informat. Recherche Opérationnelle Sér. Rouge Anal. Numér.* 9, R-2 (1975), 77–84.
- [10] ERIKSSON, K., ESTEP, D., HANSBO, P., AND JOHNSON, C. Introduction to adaptive methods for differential equations. In *Acta numerica, 1995*, Acta Numer. Cambridge Univ. Press, Cambridge, 1995, pp. 105–158.
- [11] ERIKSSON, K., ESTEP, D., HANSBO, P., AND JOHNSON, C. *Computational differential equations*. Cambridge University Press, Cambridge, 1996.
- [12] FRANCA, L. P., FARHAT, C., MACEDO, A. P., AND LESOINNE, M. Residual-free bubbles for the Helmholtz equation. *Internat. J. Numer. Methods Engrg.* 40, 21 (1997), 4003–4009.
- [13] FRANCA, L. P., AND RUSSO, A. Approximation of the Stokes problem by residual-free macro bubbles. *East-West J. Numer. Math.* 4, 4 (1996), 265–278.
- [14] FRANCA, L. P., AND RUSSO, A. Deriving upwinding, mass lumping and selective reduced integration by residual-free bubbles. *Appl. Math. Lett.* 9, 5 (1996), 83–88.
- [15] GILES, M. B., AND SÜLI, E. Adjoint methods for PDEs: a posteriori error analysis and postprocessing by duality. In *Acta numerica, 2002*, vol. 11 of *Acta Numer.* 2002, pp. 145–236.
- [16] HOUSTON, P., RANNACHER, R., AND SÜLI, E. A posteriori error analysis for stabilised finite element approximations of transport problems. *Comput. Methods Appl. Mech. Engrg.* 190, 11-12 (2000), 1483–1508.
- [17] KUNERT, G., AND VERFÜRTH, R. Edge residuals dominate a posteriori error estimates for linear finite element methods on anisotropic triangular and tetrahedral meshes. *Numer. Math.* 86, 2 (2000), 283–303.
- [18] NÄVERT, U. *A Finite Element Method for Convection–Diffusion Problems*. Ph.D. thesis, Chalmers University of Technology, Göteborg, Sweden, 1982.
- [19] PAPASTAVROU, A., AND VERFÜRTH, R. A posteriori error estimators for stationary convection-diffusion problems: a computational comparison. *Comput. Methods Appl. Mech. Engrg.* 189, 2 (2000), 449–462.
- [20] ROSENBERG, I. G., AND STENGER, F. A lower bound on the angles of triangles constructed by bisecting the longest side. *Math. Comp.* 29 (1975), 390–395.
- [21] RUSSO, A. A posteriori error estimators via bubble functions. *Math. Models Methods Appl. Sci.* 6, 1 (1996), 33–41.
- [22] SANGALLI, G. A robust a posteriori estimator for the residual-free bubbles method applied to advection-diffusion problems. *Numer. Math.* 89, 2 (2001), 379–399.

- [23] SANGALLI, G. Capturing small scales in elliptic problems using a residual-free bubbles finite element method. *Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal* 1, 3 (2003), 485–503.
- [24] VERFÜRTH, R. A posteriori error estimators for convection-diffusion equations. *Numer. Math.* 80, 4 (1998), 641–663.
- [25] VERFÜRTH, R. Error estimates for some quasi-interpolation operators. *M2AN Math. Model. Numer. Anal.* 33, 4 (1999), 695–713.

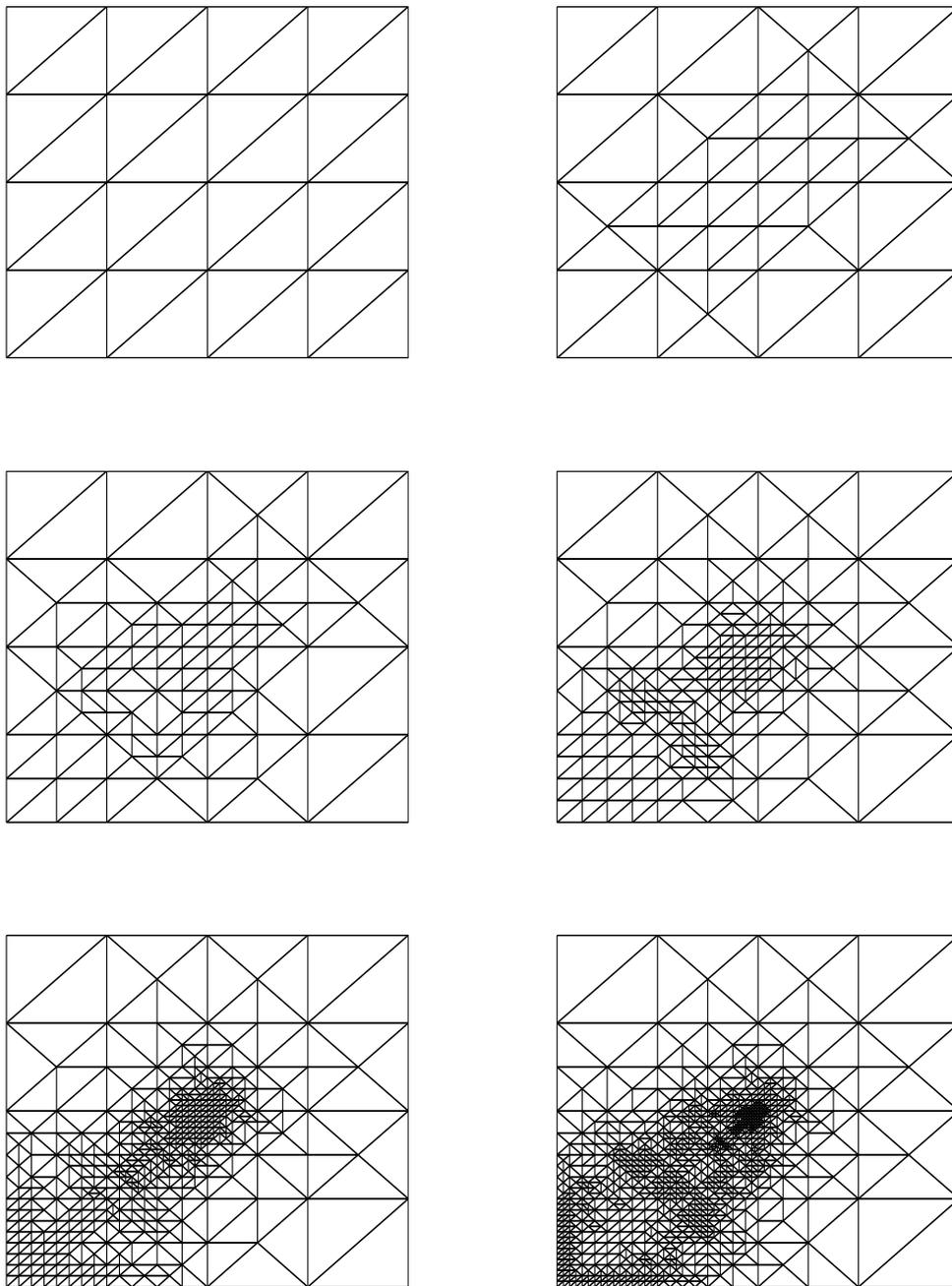


Figure 6: Example 2. Successive mesh refinements, $\varepsilon = 10^{-2}$, $P = (.49, .49)$

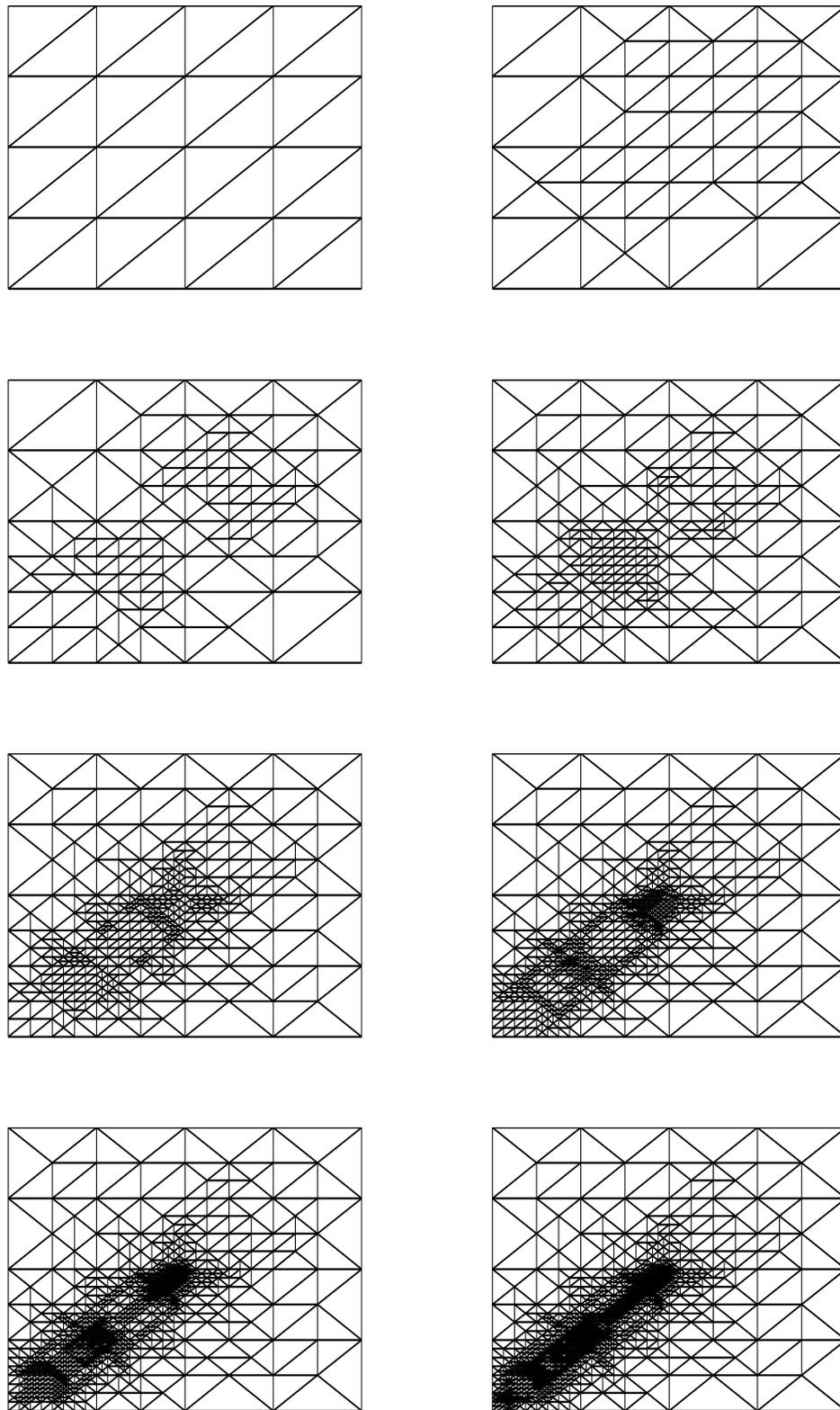


Figure 7: Example 2. Successive mesh refinements, $\varepsilon = 10^{-6}$, $P = (.49, .49)$

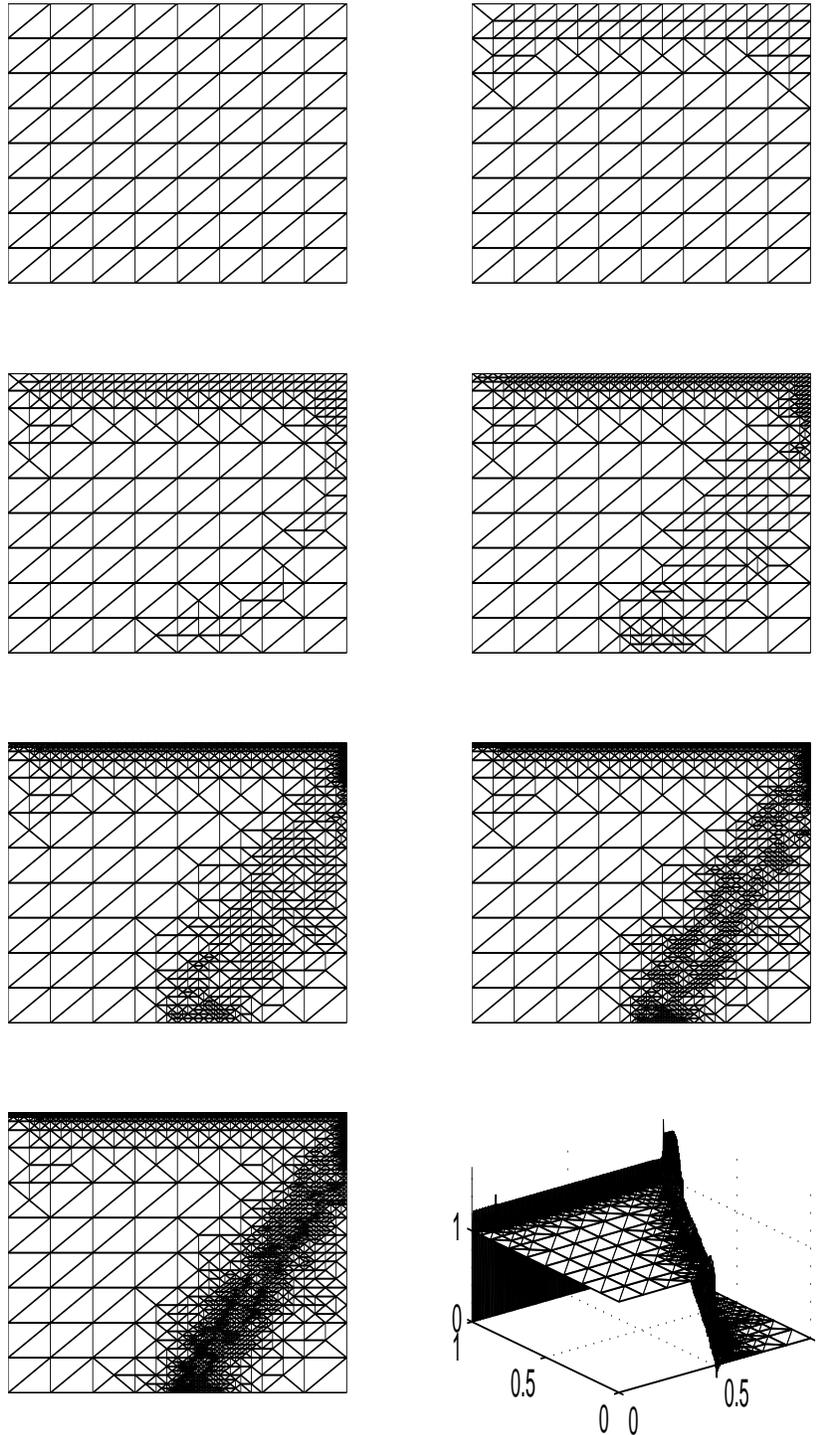


Figure 8: Example 3. Successive mesh refinement and final solution for $J(u) = \int_{\Omega} u$ and $\varepsilon = 10^{-6}$.

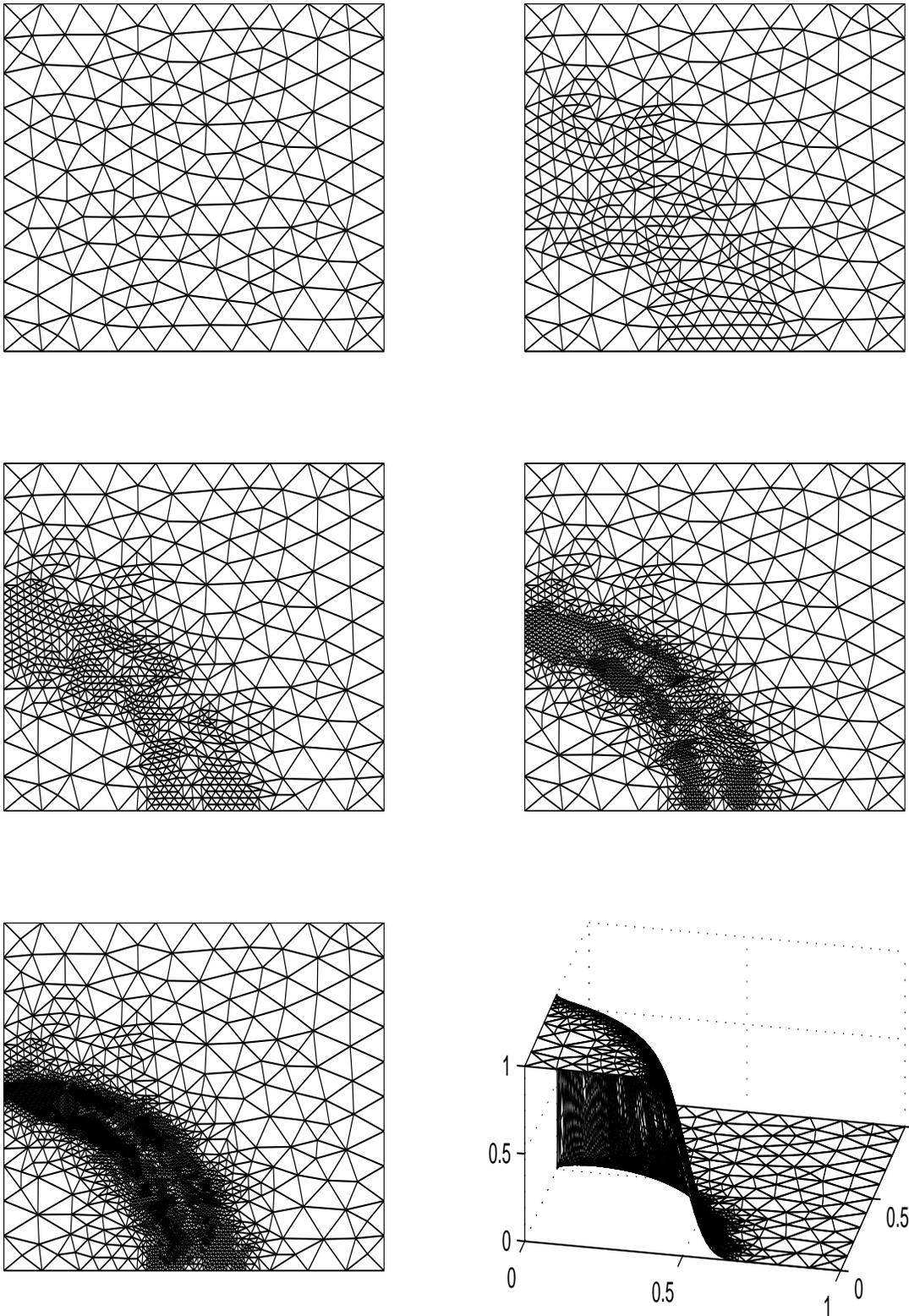


Figure 9: Example 6. Successive mesh refinement using the error estimator \mathcal{E}_B , $\varepsilon = 10^{-3}$.

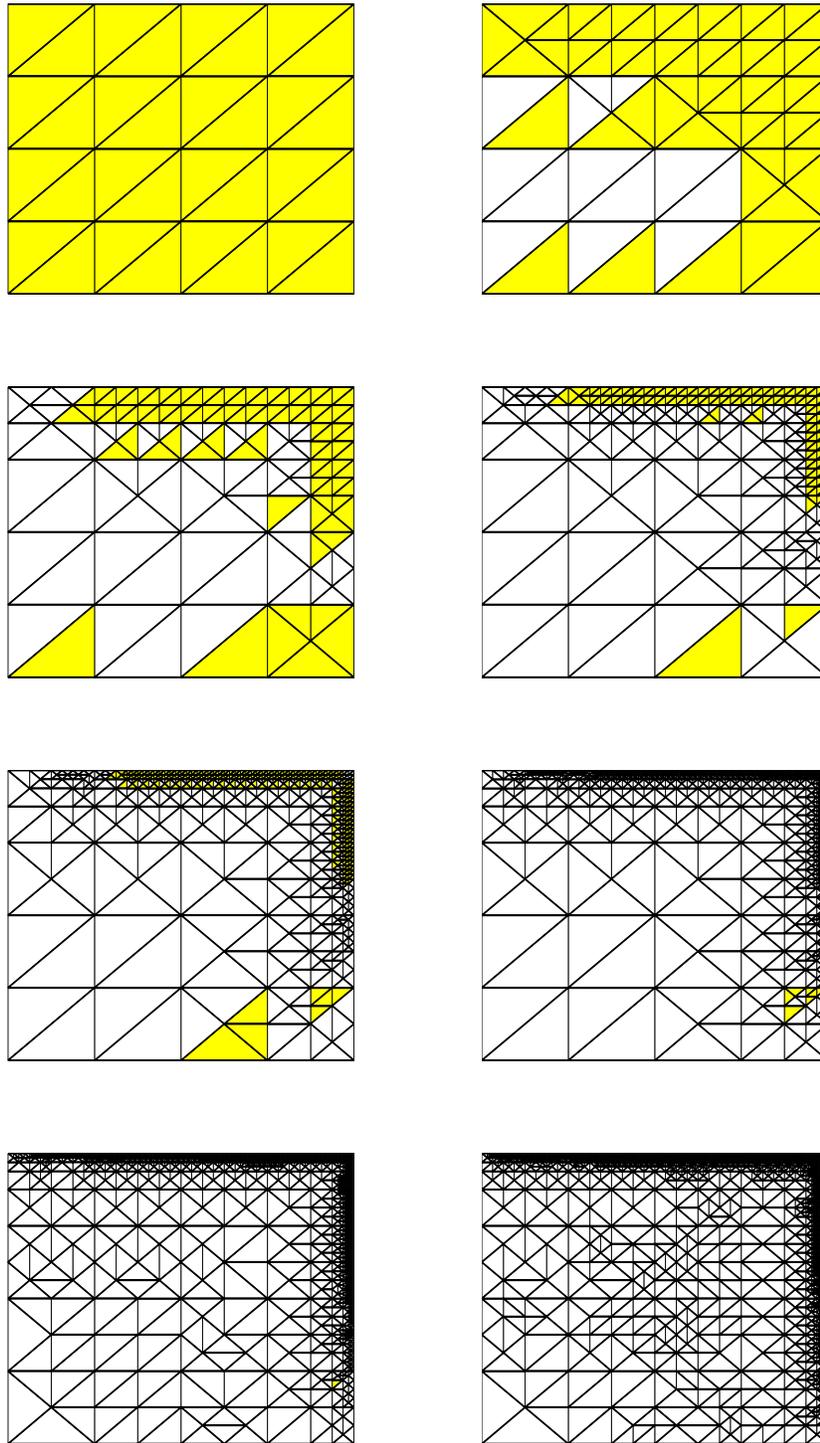


Figure 10: Example 7. Successive mesh refinement and bubble derefinement, $\varepsilon = 10^{-2}$.

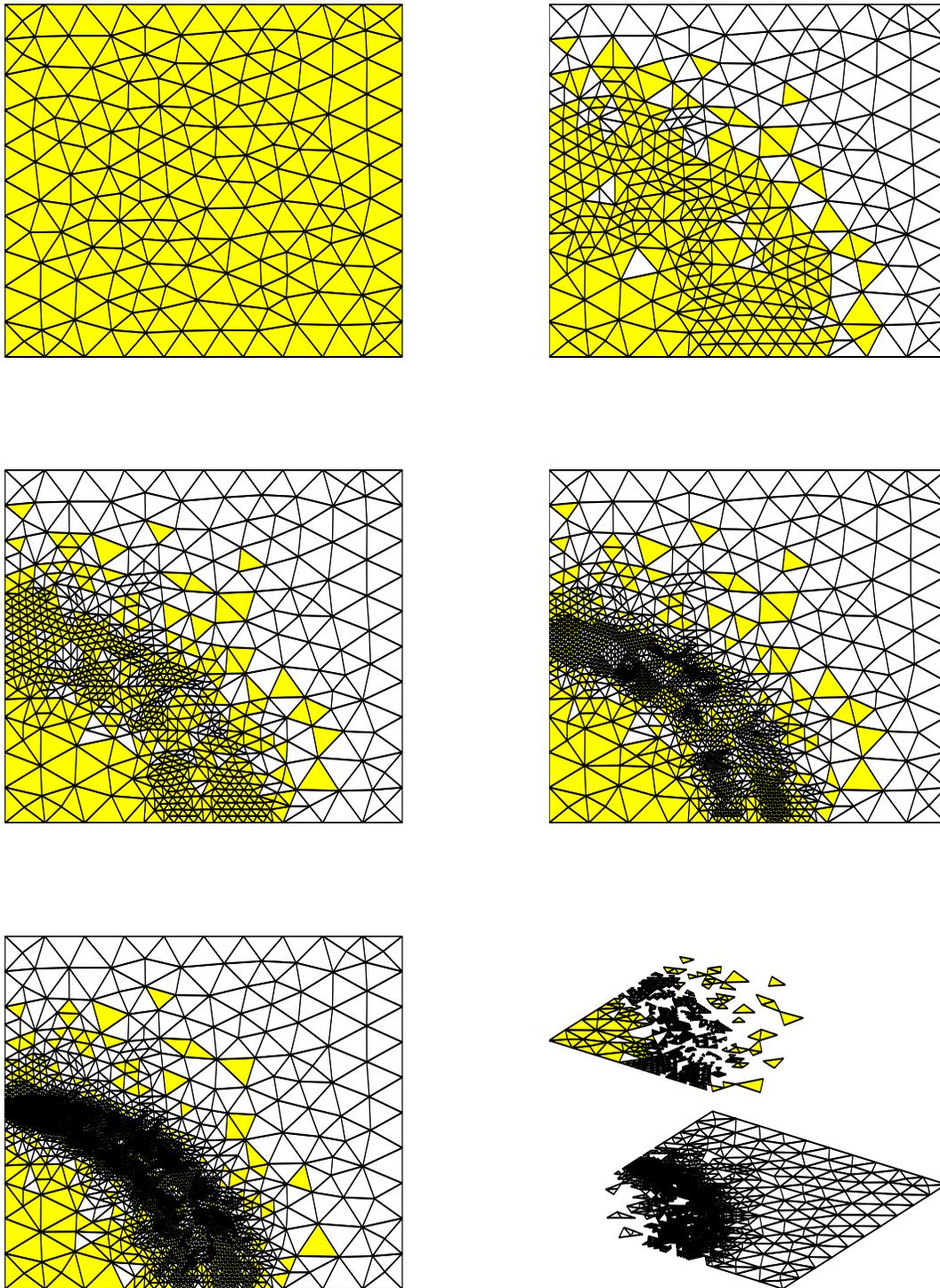


Figure 11: Example 8. Successive mesh refinement and bubble derefinement, $\varepsilon = 10^{-3}$.