View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by HAL-Ecole des Ponts ParisTech

Weighted Error Estimators for the Incompressible Navier-Stokes Equations

Roland Becker

To cite this version:

Roland Becker. Weighted Error Estimators for the Incompressible Navier-Stokes Equations. RR-3458, INRIA. 1998. <inria-00073232>

HAL Id: inria-00073232

https://hal.inria.fr/inria-00073232

Submitted on 24 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Weighted Error Estimators for the Incompressible Navier-Stokes Equations

Roland Becker

N° 3458

Juillet 1998

THÈME 4

Rapport de recherche

ISSN 0249-6399

Weighted Error Estimators for the Incompressible Navier-Stokes Equations

Roland Becker

Thème 4 — Simulation et optimisation
de systèmes complexes

Projet Caiman

Rapport de recherche n° 3458 — Juillet 1998 — 23 pages

Abstract: We present the application of *weighted* a posteriori error estimation to the incompressible Navier-Stokes equations. Using information from an appropriate dual problem, we control the discretization error of a functional of the solution as the drag or lift coefficient of an object immerged in a fluid. As test cases we consider a benchmark problem of a chanel flow and the flow around a cylinder in an unbounded domain.

Key-words: Adaptivity, Finite Elements, CFD

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, B.P. 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : 04 92 38 77 77 - International : +33 4 92 38 77 77 — Fax : 04 92 38 77 65 - International : +33 4 92 38 77 65

Estimateurs d'erreurs ponderés pour les équations de Navier-Stokes incompressibles

Résumé : Nous présentons l'application des estimateurs d'erreurs a posteriori *pondérés* aux équations de Navier-Stokes incompressibles. En utilsant les informations d'un problème dual approprié, nous développons un estimateur pour le contrôle de la valeur d'une fonctionnelle donnée de la solution, comme par exemple la traînée ou la portance d'un corps submergé dans un fluide. Nous considerons ici un benchmark "écoulement autour d'un cylindre" ainsi qu'un écoulement dans un domaine non-borné.

Mots-clés : Adaptivité, Eléments Finis, CFD

1. Introduction

This paper describes the application of weighted a posteriori error estimators to finite element discretizations of the stationary incompressible Navier-Stokes equations. The general idea of a posteriori error estimates is the control of discretization errors produced by numerical methods. However, in many engineering applications the question arises in which norm the error should be measured. Numerical simulations are often done to approximate certain quantities as drag or lift forces. It has therefore been proposed in [4] to derive a posteriori error estimates directly for these physical quantities, which are considered as functionals on the space of solutions. The general approach presented therein and further analyzed in [3] relies on the concept of a posteriori estimates via duality arguments developed by Eriksson, Johnson and their coworkers, [9], [12] and the literature cited there.

We consider two applications to two-dimensional flow around a cylinder. The first is a benchmark for incompressible flow solvers which was designed to compare the performance of different numerical approaches. We concentrate on the simplest test case proposed, a stationary flow at low Reynolds number. We demonstrate that our approach leads to automatic construction of efficient meshes. Concerning the quantitative behavior of the estimator, it turns out that at least the order of magnitude can be predicted.

The second example is again the computation of a flow around a cross-section of a circular cylinder, but in an unbounded domain. Uniform flow is supposed at infinity. In this case, there is an analytical solution for the Stokes equations. We perform computations for different Reynolds number in order to investigate the behavior of the estimator with respect to the nonlinearity.

The discretization uses continuous bilinear finite element spaces for both velocity and pressure. Stability is achieved by introduction of certain local stabilization terms as proposed for example in [11]. The generalization to other finite elements is however straight-forward.

The following section is devoted to the derivation of the estimator. Its purpose is to outline the mechanism of error control for a stabilized finite element method. Additional complications occuring in the test cases considered, as the approximation of curved boundaries, are treated seperately.

2. General Approach

In this section, we sketch the derivation of the weighted a posteriori error estimator. The exposition is on a general level in order to make the mechanism of error control transparent. First, we describe a general stabilized finite element discretization.

Let us assume, that we want to discretize the partial differential equation $Lu = f$ on a bounded domain $\Omega \subset \mathbb{R}^d$, $d = 2, 3$, with appropriate boundary conditions. For the incompressible Navier-Stokes equations, the vector of unknowns, $u = [v, p]$, consists of the velocity field v and the pressure p. The operator is $L(u) = [v \cdot \nabla v + \nabla p - \nu \Delta v, \operatorname{div} v]$, where ν is the normalized viscosity. We denote by a the semilinear form of the corresponding variational equation, defined on a space V, and suppose that it admits an unique solution. For the incompressible Navier-Stokes equations with homogeneous boundary conditions, we have $V = H_0^1(\Omega)^d \times L^2(\Omega)/\mathbb{R}$ and, denoting the test function by $\Phi = [\varphi, \xi]$,

$$a(u, \Phi) = \nu(\nabla v, \nabla \varphi) + (v \cdot \nabla v, \varphi) - (p, \operatorname{div} \varphi) + (\operatorname{div} v, \xi).$$

The discretization uses a conforming finite element space $V_h \subset V$ built from a quasi-uniform triangulation $\mathbb{T}_h = K$ with elements K of size h_K, see for example [7]. This implies that the domain is piecewise polygonal and that boundary conditions are in the finite element space. For the case that these conditions are not met, see the modifications in the next section.

To simplify reading, we introduce the following notations for element-wise scalar products:

$$< u_1, u_2 > := \sum_{K \in \mathbb{T}_h} (u_1, u_2)_K \quad \text{and} \quad < u_1, u_2 >_\delta := \sum_{K \in \mathbb{T}_h} \delta_K (u_1, u_2)_K,$$

where $\delta = \{\delta_K\}$ is a set of local stabilization parameter depending on h_K, for the choice of this parameter see for example [11]. With these notations, the discretization reads:

$$a(u_h, \Phi) + < Lu_h, S\Phi >_\delta = (f, \Phi) + < f, S\Phi >_\delta \qquad \forall \Phi \in V_h. \tag{2.1}$$

Here S denotes a partial differential operator, which is chosen in order to enhance the stability of the discrete bilinear form. Note that this discretization is automatically consistent. There are different possible choices for S. For a Least-Squares method one takes $S = L$, or a variant of this, [11]. The subgrid method ([10] and [17]) leads to $S = -L^*$. In the numerical computations presented below, we take $S\Phi = [\nu \Delta \varphi + v \cdot \nabla \varphi + \nabla \xi, 0]$.

In the following, we will abbreviate the nonlinear bilinear form on the left of (2.1) by $B_h(\cdot,\cdot)$, since it depends on the mesh size. Its derivative with respect to the first argument at u will be denoted by $B_h'[u](\cdot,\cdot)$.

If we assume the continuous solution to be locally more regular (that is in our case $v_K \in H^2(K)$ and $p_K \in H^1(K)$ $\forall K \in \mathbb{T}_h$), we get the following Galerkin property of the error $u - u_h$:

$$D_h(u - u_h, \Phi) = 0 \qquad \forall \Phi \in V_h, \tag{2.2}$$

where $D_h(\cdot,\cdot) = \int_0^1 B_h'[\lambda u + (1-\lambda)u_h]\, d\lambda(\cdot,\cdot)$ denotes the tangential bilinear form. Note that (2.2) gives a precise characterization of the discretization.

Next, we derive the a posteriori error estimator with respect to a given functional J defined on the space of solutions. For simplicity, we will suppose J linear. This functional may be thought of as the drag coefficient or a point value of the solution. But we can also derive an estimator for a global norm as the L^2-norm of the velocities for example by setting $J(\Phi) = \|v - v_h\|^{-1}(v - v_h, \varphi)$. Following [3] and [15], we define the dual solution $z \in V$ by the equation:

$$D_h(\Phi, z) = J(\Phi) \qquad \forall \Phi \in V. \tag{2.3}$$

Since the involved bilinear form involves higher order derivatives, existency and uniqueness of (2.3) have to be verified carefully, see [15].

Choosing $\Phi = u - u_h$ and using (2.2), we get the following error representation:

$$\begin{aligned} J(u - u_h) &= D_h(u - u_h, z) = D_h(u - u_h, z - z_h) \\ &= B_h(u, z - z_h) - B_h(u_h, z - z_h) \\ &= (f, z - z_h) + < f, S(z - z_h) >_\delta \\ &- a(u_h, z - z_h) - < Lu_h, S(z - z_h) >_\delta \\ &=< f - Lu_h, (z - z_h) + \delta S(z - z_h) > + \mathrm{jump}(u_h, z - z_h), \end{aligned} \tag{2.4}$$

where $z_h \in V_h$ denotes an arbitrary approximation of z and the last term expresses possible jumps over element edges (due to discontinuities of the normal derivatives of the velocities or discontinuity of the pressure, see the exact form below). Note that up to here, we have an exact expression for the error. The introduction of z_h only serves for the following localization:

Proposition 2.1. *The following generic estimates holds:*

$$J(u - u_h) \le \sum_{K \in \mathbb{T}_h} \left\{ \rho_K^1 \omega_K^1 + \rho_K^2 \omega_K^2 \right\} \qquad \textit{with} \tag{2.5}$$

$$\rho_K^1 = \|f - Lu_h\|_K, \qquad \omega_K^1 = \|z - z_h\|_K + \delta_K \|S(z - z_h)\|_K,$$
$$\rho_K^2 = \|\text{jump}(u_h)\|_{\partial K} \quad \textit{and} \quad \omega_K^2 = \|z - z_h\|_{\partial K}.$$

Proof. This follows directly from (2.4) by applying the Cauchy-Schwartz inequality on each element. □

Now, we give the detailed form of the estimator for the incompressible Navier-Stokes equations with the stabilization operator S chosen as above. The estimator for different stabilization methods is similar.

Proposition 2.2. *For the stabilized finite element method with continuous pressure approximation, we have, denoting the dual solution by $z = [w, r]$:*

$$J(u - u_h) \leq \sum_{K \in \mathbb{T}_h} \left\{ \rho_K^1 \omega_K^1 + \rho_K^2 \omega_K^2 + \rho_K^3 \omega_K^3 \right\} =: \eta^0 \tag{2.6}$$

with

$$\begin{aligned}
\rho_K^1 &= \|f + \nu \Delta v_h - v_h \cdot \nabla v_h - \nabla p_h\|_K, \\
\omega_K^1 &= \|w - w_h\|_K + \delta_K \|v \cdot \nabla(w - w_h) + \nabla(r - r_h)\|_K, \\
\rho_K^2 &= \nu \|[\partial_n v_h]\|_{\partial K}, \qquad \omega_K^2 = \|w - w_h\|_{\partial K}, \\
\rho_K^3 &= \|\operatorname{div} v_h\|_K, \qquad \omega_K^3 = \|r - r_h\|_K.
\end{aligned}$$

For a discussion on how an estimator like (2.5) can be sharp, we refer to [3]. We note however, that is has the optimal order in terms of powers of global mesh size parameter. This can be seen by local interpolation. If we know the dual solution to be in $H^2 \times H^1$, which holds true for L^2-error control of the velocites for instance, we could estimate $\sum_K \|w - w_h\|_K \leq Ch^2 \|\nabla^2 w\|_\Omega$, where $h = max_K h_K$. Together with the fact that the residuals behave $O(1)$ for linear finite elements, we immediately get that the estimator behaves like $O(h^2)$. The other terms can be treated similarly.

The estimate (2.6) is not computable, since it needs the dual solution z. In principle, there are three possibilities:

- Use Cauchy-Schwartz to liberate the estimator from the presence of z. This needs a global *a priori* estimate for the dual equation.
- Use precise *a priori* knowledge of the structure of the dual solution in order to replace the weights ω_K by analytical quantities.
- Approximate the weights numerically by solving a discrete dual problem.

The first option is the one usually chosen in a posteriori error analysis for finite elements, see for example [20]. It relies on the possibility of a sharp estimate of the dual solution. In the present case of Navier-Stokes equations this is a difficult task. Even if such *a priori* estimates are available, the estimator may be inefficient, if the dual solution (respectively its derivatives) has a strong inhomogeneous behavior, see the numerical computations below.

The second possibility requires an even sharper knowledge of the mathematical structure of the dual problem. This obviously limits its range of applications. In the case of a global norm for a singularly perturbed convection-diffusion or convection-reaction equation, the approach of Shishkin [19] can be interpreted in this way.

The third option has been proposed in [4]. Beside the more technical questions of what interpolation operator should be used (this determines the precise form of the residuals ρ_K in (2.6)) and how the weights ω_K should be approximated, there are two basic numerical approximations:

- The dual problem has to be discretized.
- For nonlinear problems, the dependence of the dual equation on the exact solution u has to be eliminated. This is done by linearizing around u_h instead of linearizing around the mean between u and u_h.

In order to limit the numerical cost of the estimator, it has been proposed in [4] to use the same mesh for the discretization of (2.3) as for the primal approximation. As shown by numerical examples for a linear model problem, the *quantitative* behavior of the estimator depends heavily on this approximation. Solving the dual problem by quadratic finite elements leads in simple cases to an asymptotically exact estimator for a linear primal approximation on the same mesh, see [3]. In general, for a linear problem, we expect that the accuracy of the estimator can be enhanced by increasing the accuracy of the dual problem. In this paper we limit ourselves to a cheap estimator and solve the dual equations with linear finite elements on the same mesh.

The second approximation is a linearization error. The bilinear form on the left of (2.3) uses both information from the discrete solution u_h and the continuous solution u. Replacing u by u_h leads to a perturbed dual solution $\tilde{z}$. The difference satisfies:

$$D_h(\Phi, z - \tilde{z}) = \int_0^1 \int_0^1 B_h''[u_h + \mu\lambda(u - u_h)][\lambda(u - u_h)](\Phi, \tilde{z})\, d\lambda\, d\mu. \tag{2.7}$$

In order to derive an (a priori) estimate from (2.7) we have to linearize around u and neglect the stabilization terms in order to get a bilinear form independent of h. But

more important, we have to make restrictive assumptions on the nonlinearity of the problem in order to estimate the terms on the right hand side.

We see from (2.7) that the linearization error is quadratic with respect to the error. We could therefore hope that it is neglectible for fine meshes. See the last section for numerical results on the behavior of the simple estimator with respect to the nonlinearity.

We summarize the adaptive algorithm to be used in the numerical computations as follows:

- Choose a coarse mesh $\mathbb{T}_{h_0}$
- Do the following iteration
  1. Compute u_{h_i} on $\mathbb{T}_{h_i}$
  2. Compute z_{h_i} on $\mathbb{T}_{h_i}$
  3. Evaluate an approximation $\tilde{\eta}$ of the estimator
  4. Construct a new mesh $\mathbb{T}_{h_{i+1}}$.

Here $\tilde{\eta}$ is the following approximation for the estimator. We replace the weights ω_K in (2.5) by $\|i_h^{(2)} z_h - z_h\|_K$, where $i_h^{(2)}$ is the quadratic interpolation operator on a patch of elements, see [3].

z_h is obtained by solving the discrete dual equations, which read for the incompressible Navier-Stokes equations:

$$\nu(\nabla\varphi, \nabla w_h) + (u_h \cdot \nabla\varphi + \varphi \cdot \nabla u_h, w_h) + (\operatorname{div}\varphi, r_h) - (\xi, \operatorname{div} w_h) + \\ < DL(u_h)^* z_h, S\Phi >_\delta = J(\Phi) \qquad \forall \Phi \in V_h,$$

where as before $z_h = [w_h, r_h]$. The left hand side corresponds to the operator to be inverted in each Newton step for the primal equations.

For construction of the new mesh we use hierarchical refinement (or de-refinement). The decision of when a cell has to be refined is either done by an ordering algorithm similar to the one described in [8] or by an optimization strategy proposed in [5].

3. Benchmark

As first example, we consider the computation of the pressure drop, the drag and lift coefficients in the viscous incompressible (stationary) flow around a cylinder in 2D. This is part of a set of benchmark problems discussed in [18].

Due to the curved boundary of the domain, additional error sources arise. We generalize the estimator derived before to this situation.

3.1. **Description of the problem.** The benchmarks requires the computation of a viscid incompressible flow in a channel around a cylinder. The geometry is shown in Figure 1. The governing equations are the two-dimensional stationary Navier-Stokes equations

$$-\nu\Delta v + v\cdot\nabla v + \nabla p = f, \quad \operatorname{div} v = 0, \quad \text{in } \Omega, \tag{3.8}$$

for the velocity v and the pressure p (density $\rho = 1$) on the bounded region Ω described in Figure 1. At the boundary $\partial\Omega$, the usual no-slip condition is posed along rigid parts together with Poiseuille inflow and free-stream outflow conditions at inlets and outlets, respectively,

$$v|_{\Gamma_{\text{rigid}}} = 0, \quad v|_{\Gamma_{in}} = v_{\text{in}}, \quad (\nu\partial_n v - p\cdot n)|_{\Gamma_{out}} = 0. \tag{3.9}$$

The quantities to be computed are the pressure difference between the front point a_{front} and back point a_{back} of the cylinder,

$$J_{\Delta p}(v,p) = p(a_{\text{front}}) - p(a_{\text{back}}), \tag{3.10}$$

as well as the drag and lift coefficients

$$\begin{aligned} J_{\text{drag}}(v,p) &= \frac{2}{\bar{v}^2 D}\int_S \{\nu\partial_n v_t n_y - p n_x\}\, ds, \\ J_{\text{lift}}(v,p) &= -\frac{2}{\bar{v}^2 D}\int_S \{\nu\partial_n v_t n_x + p n_y\}\, ds. \end{aligned} \tag{3.11}$$

(v_t denoting the tangential component of v, D the diameter of the cylinder and $\bar{v}$ the mean velocity). The Reynolds number is $Re = \bar{v}D/\nu = 20$, so that the flow is stationary. With respect to the estimator derived in the previous section, we have

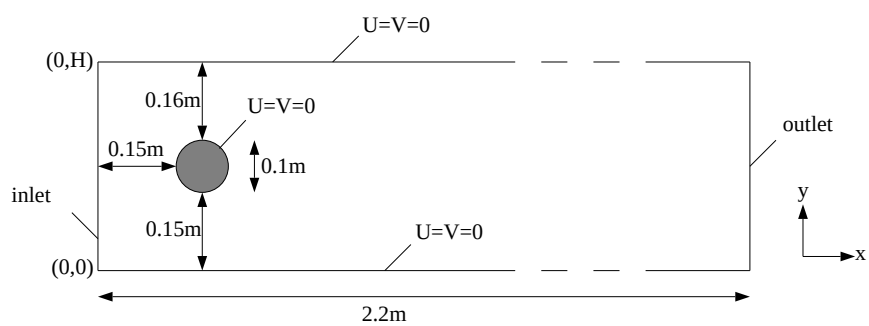

FIGURE 1. Geometry of the benchmark problem *"flow around a cylinder"*

to consider three additional sources of errors. First, the inflow profile is a quadratic

function and therefore not included in the space of linear finite elements. Second, the computational domain contains a curved boundary, leading to an approximation error of the boundary. More precisely, the discretization uses a triangulation covering a domain Ω_h strictly larger than Ω. For the sake of readability, we do not carry out all the necessary modifications. We only remark, that the integrals defining the dual equation (2.3) will be defined on the larger domain Ω_h. Functions defined on Ω will be supposed to be prolongated to the larger domain. This will be discussed below. The third additional contribution comes from the stabilization terms near the curved boundary.

Before deriving the additional terms of the estimator, we cite a lemma from [1] which is also useful for the computation of the drag and lift forces.

Lemma 3.1. *Let the tensor σ satisfy*

$$(\sigma, \nabla\varphi) = (f, \varphi) \qquad \forall \varphi \in H_0^1(\Omega)^d,$$

with $f \in L^2(\Omega)^d$. Then, for an arbitrary vector $\psi \in H^1(\Omega)^d$, there holds:

$$(\sigma, \nabla\psi) = (f, \psi) + (n \cdot \sigma, \psi)_{\partial\Omega}. \tag{3.12}$$

Taking into account the boundary approximation, the complete estimator has the form:

Proposition 3.1. *For the benchmark configuration we have the following estimate:*

$$|J(u) - J(u_h)| \leq \eta^0 + \eta_{in} + \eta_{cyl} + \eta_{cyl}^\delta =: \eta, \tag{3.13}$$

where η^0 denotes the estimator in (2.6) *and the additional terms are given by:*

$$\eta_{in} = \sum_{K \cap \Gamma_{in} \neq \emptyset} \|v_{in} - v_{in,h}\|_{\partial K \cap \Gamma_{in}} \cdot \|\nu \partial_n w - r \cdot n\|_{\partial K \cap \Gamma_{in}},$$

where $v_{in,h}$ is the discrete boundary function and

$$\eta_{cyl} = \sum_{K \cap (\Omega_h \setminus \Omega) \neq \emptyset} \|\nu \partial_n v - p \cdot n\|_{K \cap \Gamma_{cyl}} \cdot \|z_h\|_{K \cap \Gamma_{cyl}},$$

$$\eta_{cyl}^\delta = \sum_{K \cap \Gamma_{cyl} \neq \emptyset} \rho_K^1 \delta_K \|S z_h\|_K,$$

with ρ_K^1 defined in (2.6).

Proof. We only sketch the derivation of the estimator. The different boundary errors are considered separately.

For the inflow error, we remark that the Galerkin relation (2.2) still holds, but we may not immediately choose $\Phi = u - u_h$ in (2.3). We can however apply the preceding lemma with $\sigma = \nu\nabla w - r$ and $\psi = u - u_h$, noting that the lemma also holds for a right hand side not necessarily in L^2, but which involves only values with a fixed distance from the boundary. The boundary integral in (3.12) then gives an additional error contribution which reads

$$\int_{\Gamma_{\text{in}}} (\nu\partial_n w - r \cdot n)(v - v_h) = \int_{\Gamma_{\text{in}}} (\nu\partial_n w - r \cdot n)(v_{\text{in}} - v_{\text{in},h})$$

and can be estimated by η_{in}.

The curved boundary leads to a perturbed Galerkin equation (2.2), which now reads for $\Phi \in V_h$:

$$D_h(u - u_h, \Phi) = \int_{\Gamma_{\text{cyl}}} (\nu\partial_n v - p \cdot n)\varphi,$$

again due to the lemma. On the other hand we may choose $u - u_h$ as test function in (2.3), since u is prolongated by zero. The additional term is thus bounded by η_{cyl}.

Finally, due to the curved boundary, the stabilization produces another inconsistency leading to an additional perturbation of (2.2). It has the form:

$$D_h(u - u_h, \Phi) = \sum_{K \cap \Gamma_{\text{cyl}} \neq \emptyset} \delta_K \int_K (Lu, S\Phi)$$

We estimate the last term by $\eta^{\delta}_{\text{cyl}}$. □

Next we define the functionals for the three test computations. For the pressure difference, point evaluation is not well defined on the pressure space (although we expect the pressure difference to be perfectly well defined). As shown in [3], such a singular functional (we expect a behavior as r^{-3} for the weights, where r denotes the distance from the point of evaluation), should be stronger regularized than just a projection on the actual mesh. We therefore use the following functional:

$$\tilde{J}_{\Delta p}(v, p) = \frac{1}{|B^a_\epsilon|} \int_{B^a_\epsilon} p - \frac{1}{|B^f_\epsilon|} \int_{B^f_\epsilon} p$$

with $B^{x_0}_\epsilon = \{x \in \Omega_h : |x - x_0| \leq \epsilon\}$ and $\epsilon \sim \text{TOL}$, the prescribed tolerance.

For the evaluation of the drag- and lift forces, we employ the Babuska-Miller trick [1]. Instead of using the boundary integrals in (3.11) we use the following evaluation:

$$\tilde{J}_{\text{drag/lift}}(v, p) = B_h(u_h, \Psi_{\text{drag/lift}}),$$

where B_h as before and $\Psi_{\text{drag/lift}}$ is defined as follows. We prolongate the directional vectors $[1,0]$ and $[0,1]$ on the boundary of the discrete domain to a discrete vector field $\psi_{\text{drag/lift}}$. This can be done since the directional vectors are traces of functions from the finite element space. We then set $\Psi_{\text{drag/lift}} = [2/(\bar{v}^2 D)\ \psi_{\text{drag/lift}}, 0]$. Note that the definition of $\tilde{J}$ does not depend on the choice of this prolongation, since the difference of two different prolongations lies in the space of the discrete test functions. As shown in [13] for the case of polygonal boundaries, this definition leads to second order accuracy with respect to a global mesh size. For this, the authors propose a modified dual problem, which uses the directional vectors as inhomogeneous Dirichlet boundary conditions instead of the right hand side in (2.3). Following this, we define the dual solution by $z = \psi$ on Γ_{cyl} and:

$$D_h(\Phi, z) = 0 \qquad \forall \Phi \in V(\Omega_h). \tag{3.14}$$

We then get an an estimator similar to the one before:

Proposition 3.2. *For the drag and lift coefficient we have*

$$|J(u) - \tilde{J}(u_h)| \leq \eta^0 + \eta_{in} + \eta^\delta_{cyl} + \tilde{\eta}_{cyl},$$

where in η^0, η_{in} and eta^δ_{cyl} as before and $\tilde{\eta}_{cyl}$ given by:

$$\tilde{\eta}_{cyl} = \sum_{K \cap (\Omega_h \setminus \Omega \neq \emptyset)} \|\nu \partial_n v - p \cdot n\|_{K \cap \Gamma_{cyl}} \|\psi - z_h\|_{K \cap \Gamma_{cyl}}.$$

Proof. We start by splitting the error as follows:

$$J(u) - \tilde{J}(u_h) = \{J(u) - B_h(u,z)\} + \{B_h(u,z) - B_h(u_h,z)\} = \mathrm{I} + \mathrm{II}.$$

Neglecting the inflow error, the second term equals

$$\mathrm{II} = D_h(u - u_h, z) = D_h(u - u_h, z - z_h)$$

and can be estimated as before by η^0. The inflow error leads to the same modification as before and can be estimated by η_{cyl}. For the first term we denote by $\tilde{z}$ the solution of the dual problem on the domain Ω:

$$\mathrm{I} = a(u, \tilde{z}) - B_h(u,z) = (a(u,z) - B_h(u,z)) + a(u, \tilde{z} - z).$$

The consistency of the stabilization allows us to estimate the first term on the right by η^δ_Γ and for the second term we use again the lemma yielding:

$$a(u, \tilde{z} - z) = \int_{\Gamma_{\text{cyl}}} (\nu \partial_n v - p \cdot n)(\psi - z),$$

which is estimated by $\tilde{\eta}_{\text{cyl}}$. □

The additional terms of the estimator for the boundary approximations still contains the continuous dual and primal solutions. In order to obtain a computable estimator, we approxamte them as before by replacing the continuous solutions by the computed discrete ones. We further mention that the estimation of the boundary terms is a rather crude one. But this will be sufficient for the present purpose.

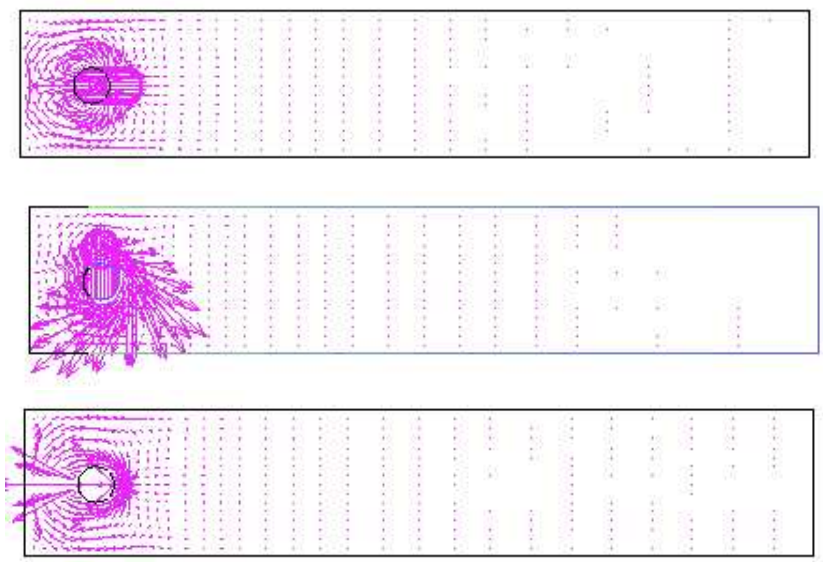

FIGURE 2. Dual solutions for C_D, C_L and Δp

3.2. **Computational results.** We shortly describe the numerical approach used for the following computations. The discretization uses isoparametric bilinear finite elements for both velocities and pressure. Numerical integration is always done using Gauss quadrature with four points. The original Gelerkin formulation is stabilized as explained before. For ease of local mesh refinement we allow elements with hanging nodes. A proper definition of the finite element functions leads to discrete spaces which are still conforming. For more details and a description of a multigrid solver on the hierarchy of spaces created by the mesh adaptation algorithm we refer to [2].

First we give a comparison of the quality of the meshes generated by the weighted estimator with some of the results given in [18]. We only compare with the apparently most accurate results produced by different second order methods. The reference

values given in [18] are $0.1172 < \Delta p < 0.1176$, $5.57 < C_D < 5.59$ and $0.0104 < C_L < 0.0110$. In order to compare the quality of the solutions, we need more accurate results. These are obtained by various adaptive computations on different grids with different adaptation strategies and also different types of evaluations of the functionals. Some of the results of these computations are given below. The most reliable values, which we take as reference, are found to be $\Delta p \approx 0.11752$, $C_D \approx 5.5796$ and $C_L \approx 0.01063$. For the comparison with the other results, we accept a solution if it has an error less than 0.1% in the drag, 1% in the lift and 0.2% in the pressure difference leading to a range $0.1172 < \Delta p < 0.1178$, $5.574 < C_D < 5.585$ and $0.01052 < C_L < 0.01074$. This choice of the tolerance is motivated by the results of the benchmark. This accuracy may appear rather excessive, but the test case is quite simple in comparison to real life applications.

In order to construct meshes which are well adapted to all three quantities required by the benchmark, we solve the three dual problems on the actual mesh and sum up the corresponding estimators. There is of course a certain ambiguity since it is not the sum of the absolute values of the errors which we want to estimate. We simply choose to weight the three estimators by an estimation of the corresponding quantities. This leads to an estimation of the sum of the relative errors. In Table 1

ref	N	C_D	C_L	Δ_p
4	297472	5.5678	0.0105	0.1179
7a	294912	5.5846	0.0106	0.1176
9a	240000	5.5803	0.0106	0.1175
10	667264	5.5718	0.0105	0.1169
this	10080	5.5818	0.0107	0.1176

TABLE 1. Comparison of results for the benchmark

we give the most economical acceptable solutions from [18] compared to the present algorithm.

We see from Table 1 that there is an important gain of a factor about 25 in the number of unknowns for the required precision.

This can be explained by the local mesh refinement produced by the weighted estimator. In fact, as shown in Figure 2, the dual solution for each of the computed quantities has a local behavior. Note that it would be rather difficult to guess the correct distribution of the weights *a priori*. Each of the dual solution has a different behaviour which leads to a different grid refinement for the corresponding function, as shown in Figure 3.

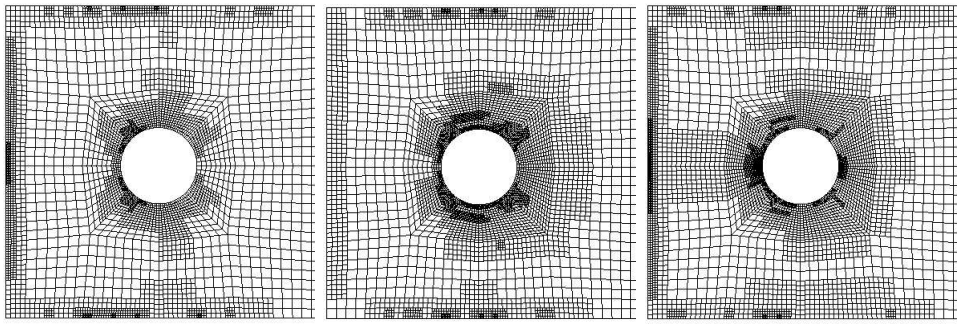

FIGURE 3. Typical grids for C_D, C_L and Δp

Now we present in more detail the computations done to find the reference values. At the same time we illustrate the quantitative behavior of the estimator.

3.2.1. *Computation of C_D.* In order to increase the reliability of the computed values, we perform computations using two different coarse meshes, shown in Figure 4. While the first one uses a very simple mesh, the second one uses a better approximation of the curved boundary. In Table 2, we give the results for the drag coefficient. The

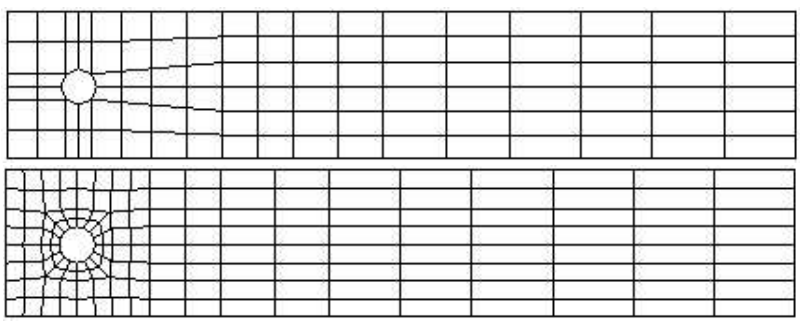

FIGURE 4. Coarse grids 1 and 2

local refinement is based on a generous algorithm which alway refines larger patches of elements. As for the subsequent Tables, N is number of unknowns, L denotes the highest level of refinement in the mesh and η is the value of the estimator.

Next, we present results obtained on grid2 with a more flexible refinement strategy. Table 3 contains two informations. First, the behavior of the estimator is given. Both the value of the estimator η and the estimated error e are given. Furthermore,

N	L	η	C_D
4128	6	7.5e-02	5.61222
9456	7	2.5e-02	5.58413
22560	8	8.9e-03	5.58170
41136	9	4.5e-03	5.58065
100464	10	1.7e-03	5.58013
248352	11	6.8e-04	5.57984
425616	12	3.7e-04	5.57968

N	L	η	C_D
3216	4	7.2e-02	5.61013
7536	6	2.5e-02	5.58302
20208	7	8.6e-03	5.58097
49008	8	3.2e-03	5.58048
90480	9	1.6e-03	5.58010
214752	11	6.5e-04	5.57973
381792	11	3.6e-04	5.57963

TABLE 2. Computations for C_D on grids 1 and 2

the quantity eff denotes the efficiency index defined by err/η. Second, we also list the value of the drag coefficient computed by the boundary integral, $\tilde{C}_D$.

N	L	C_D	$\tilde{C}_D$	η	eff	α	$\tilde{\alpha}$
984	4	5.66058	5.05619	1.1e-01	0.76	0.70	0.43
2244	5	5.59431	5.38472	3.1e-02	0.47	0.85	0.51
4368	6	5.58980	5.42795	1.8e-02	0.58	0.82	0.50
7680	6	5.58507	5.44975	8.0e-03	0.69	0.84	0.49
9444	7	5.58309	5.50251	6.3e-03	0.55	0.87	0.53
22548	8	5.58151	5.52570	2.5e-03	0.77	0.85	0.52
41952	9	5.58051	5.54776	1.2e-03	0.76	0.87	0.54
81084	9	5.58005	5.55590	6.2e-04	0.72	0.89	0.53
149232	10	5.57982	5.56380	6.1e-04	0.36	0.90	0.54
177168	11	5.57979	5.57198	4.7e-04	0.41	0.90	0.59

TABLE 3. Estimator for C_D

In order to compare the two different ways of evaluation, we suppose that the error can be written as $err = CN^{-\alpha}$. In both cases, the constant C is estimated and the values of α ($\tilde{\alpha}$ in the second case) are given. For a second order method in two dimensions we expect a value of $\alpha = 1$ and for a first order method $\alpha = 0.5$. As can be seen from Table 3, the evaluation of the drag via volume integrals is second order in this sense, even on highly locally refined meshes. The superiority with respect to $\tilde{C}_D$ is also evident.

3.2.2. *Computation of* C_L. Table 4 contains the results for the lift coefficient. The contributions of the pressure and viscous forces are listed separately ($C_L(p)$ and

$C_L(u)$). We also indicate the work spent on the solution of the primal and dual equations, w_p and w_d, measured in computation time per unknown. This number is normalized by the primal computation on the first mesh.

N	L	w_p	w_d	C_L	$C_L(p)$	$C_L(u)$	η	eff
2208	4	1.0	0.2	0.01318	0.00917	0.00400	1.3e-02	0.19
5088	5	1.1	0.2	0.01100	0.00682	0.00418	2.7e-03	0.14
14016	7	1.2	0.2	0.01071	0.00528	0.00543	7.8e-04	0.12
53040	8	1.2	0.3	0.01065	0.00730	0.00336	1.8e-04	0.18
142896	9	1.5	0.4	0.01064	0.00661	0.00403	6.4e-05	0.25
489648	11	1.6	0.6	0.01063	0.00653	0.00410	1.8e-05	0.27

TABLE 4. Estimator for C_L

3.2.3. *Computation of* Δp. The results for the pressure difference are presented in Table 5. For this quantity the refinement is much sharper as can be seen from the higher number of levels with respect to the preceding cases. This is explained by singularities in the dual solution. It can also be seen, that the estimator is less reliable in this case.

N	L	Δp	$p(a_{\text{front}})$	$p(a_{\text{back}})$	η	eff
2784	4	0.117972	0.133317	0.015346	3.6e-03	0.12
6096	6	0.117805	0.133020	0.015215	1.4e-03	0.21
16320	8	0.117541	0.132488	0.014948	4.1e-04	0.05
49296	10	0.117530	0.132344	0.014814	1.2e-04	0.09
86016	11	0.117529	0.132313	0.014784	6.2e-05	0.14
158160	12	0.117524	0.132260	0.014736	3.2e-05	0.12
281280	13	0.117522	0.132247	0.014725	1.7e-05	0.12

TABLE 5. Computation for Δp

We close this section with a comparison of the errors of the three quantities on three different grids generated by the different estimators. For this, we choose in each case a mesh with about 25000 elements. This shows that the meshes are well adapted exactly to the quantity they are designed for.

η	N	L	e_{C_D}	e_{C_L}	$e_{\Delta p}$
C_D	20208	7	2.20e-03	1.13e-01	4.62e-03
C_A	27696	7	4.80e-03	2.35e-02	2.00e-03
Δp	26880	13	3.24e-02	5.70e-01	2.47e-04

TABLE 6. Relative errors on grids generated by the estimator η

4. An Example on an unbounded Domain

In this section we consider the computation of the drag coefficient for the cut of a circular cylinder in an unbounded domain. The cylinder is immerged in a uniform flow field. The case of the Stokes equations admits an analytical solution, which shows that the drag depends linearly on the viscosity. For the Navier-Sokes equations no analytical solution is known and the design of approximative formulae for the drag coefficient in dependance of the Reynolds number has been the subject of many studies, see [21].

We use this example to investigate the behavior of the estimator for different Reynolds numbers.

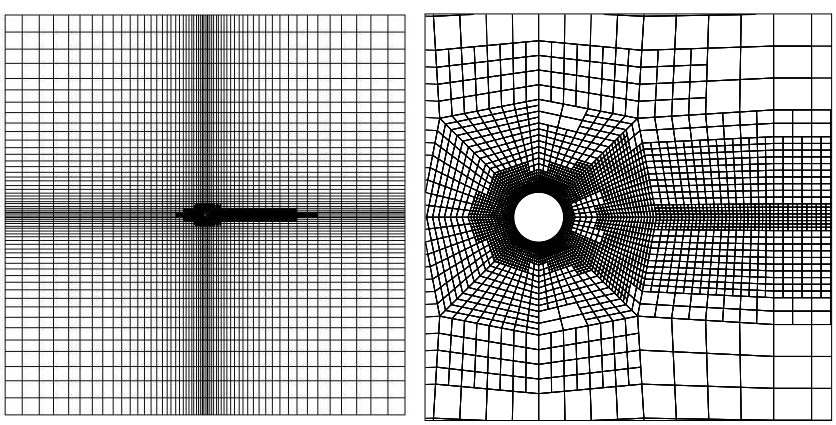

FIGURE 5. Mesh with zoom for $Re = 20$

We model the unbounded domain by choosing a large box as computational domain. The size of the box is taken 200 times larger than the cross section of the cylinder in order to avoid influence from the artificial boundaries. The estimator should indicate

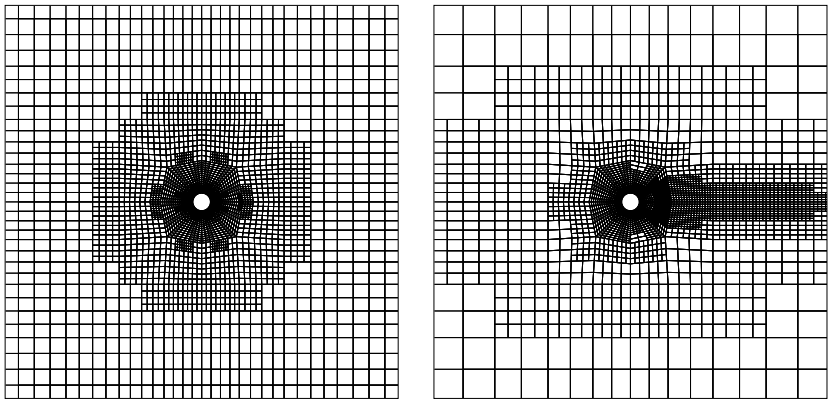

FIGURE 6. Zoom of grids for $Re = 0.1$ and $Re = 100$

whether this is justified. Small weights near the artificial boundary mean that there is no influence on the drag computation.

As we can see from Figure 5, the estimator leads to a grid refinement near the cylinder. Relatively few elements are used to cover the field far from the obstacle. This is true for all the computations performed in the range from $Re = 0.1$ up to $Re = 100$, Figure 6 shows meshes for the extreme values. However, the generated meshes are quite different depending on the Reynolds number. For high viscosity, the refinement is concentrated symmetrically around the cylinder, whereas for higher Reynolds number the refinement is much flatter and the adaptive procedure leads to a resolution of the recirculation zone. For a Reynolds number above the critical value ($Re \sim 40$), the flow tends to be unstable and the velocity profile is still not uniform at the outlet. But this does not harm the computation of the drag coefficient, as can be seen from the dual solution, see Figure 7.

For higher Reynolds numbers, the dual solution becomes concentrated on a thin layer in front of the obstacle. This is because convection is in opposite direction in the dual equations. Nevertheless, the higher weights at this place do not lead to refinement, since the flow is uniform at a certain distance in front of the cylinder and consequently the residual are very small at this place.

In Table 7, we present some computational results for different Reynolds numbers. Whereas convergence is from below for high viscosity, this situation is reversed in the high Reynolds case. Comparing the efficiency of the estimator for a given number of elements, we see that the estimator tends to underestimate the error for increasing Reynolds numbers. The highest Reynolds number case in Table 7 corresponds to an

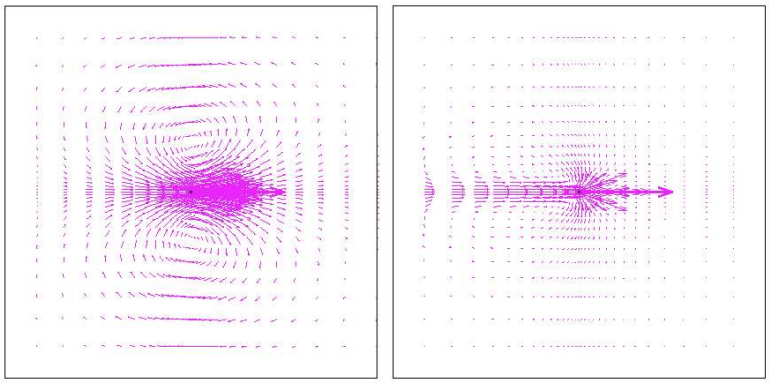

FIGURE 7. Dual solutions for $Re = 0.1$ and $Re = 40$, scaled by a factor of 100

N	L	η	C_D	eff	N	L	η	C_D	eff
		$Re = 0.2$					$Re = 2$		
11328	2	5.6e+00	31.7142	0.1	11328	2	1.1e+00	5.8597	0.1
19272	4	1.2e-01	32.0341	0.2	19236	4	1.6e-02	5.9122	0.2
33054	6	5.8e-02	32.0470	0.2	32742	6	5.0e-03	5.9143	0.3
55506	6	3.1e-02	32.0530	0.1	55422	7	2.7e-03	5.9153	0.2
94566	7	2.0e-02	32.0563	*	94596	8	1.6e-03	5.9158	*
123156	8	1.5e-02	32.0571	*	123102	8	1.2e-03	5.9159	*
		$Re = 20$					$Re = 66,7$		
11328	2	2.3e-01	1.8154	0.7	11328	2	2.0e-01	1.3355	1.7
19290	4	6.9e-03	1.6656	1.0	19326	4	9.1e-03	1.0189	2.6
32892	6	6.1e-04	1.6584	1.0	33012	6	6.9e-04	0.9952	0.2
55818	8	2.7e-04	1.6586	1.6	56268	8	2.0e-04	0.9947	1.5
93912	9	1.4e-04	1.6589	*	94938	9	1.0e-04	0.9950	*
121920	9	1.0e-04	1.6590	*	123108	10	7.4e-05	0.9950	*

TABLE 7. Results for different Reynolds numbers

unstable flow, since we know that the formation of the Van-Karman vortex street starts at approximately $Re = 40$, see [21]. The results are similar even in the case of $Re = 100$, the highest value for which we could achieve convergence. This clearly shows that instabilities cannot be predicted by our relatively cheap estimator. In order to improve the behavior of the estimator, one would need better information

about the continuous solution. For example, one could compute a more accurate solution $\tilde{u}_h$ and substitute it in place of u in the definition of the dual problem (2.3). Unfortunately, this makes the estimator more expansive than the original problem.

The mesh refinement used in the computations for Table 7 is based on the optimization strategy proposed in [5]. For a given number N of unknowns we try to minimize the value of the estimator. In each adaptation step i, we increase the maximally allowed number of unknowns N_i by a given factor. Obviously, this leads to qualitatively different meshes with respect to the Reynolds number, see Figure 6. For the highest Reynolds number we get for $N \sim 125000$ 10 levels of refinement, whereas for low Reynolds we only have 8 levels.

We close this section by a comparison of the computed drag coefficient with the formula of Sucker and Bauer (see [21] page 183), which seems to be in good agreement with experimental data for a wide range of Reynolds numbers.

A we can see from Figure 8, the numerical values are very close to the curve generated by the formula up to $Re = 10$. Beyond this value, the computed drag coefficient is smaller.

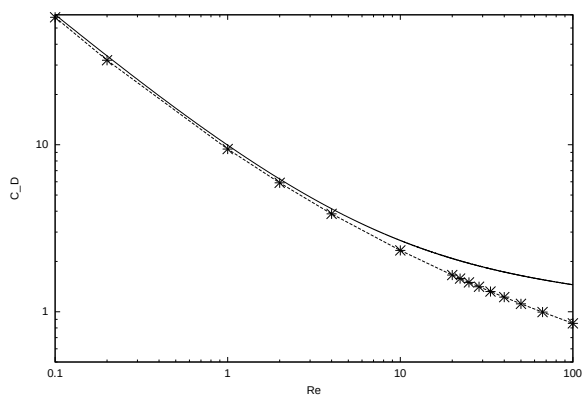

FIGURE 8. Comparison between Computation and Formula

5. CONCLUSION

We have outlined a general approach to a posteriori error control in finite elements. The application to the incompressible Navier-Stokes equations showed that the use of the weighted error estimator presented leads to automatic generation of efficient meshes.

The strength of our method being the possibility of handling different error measures, we expect a wide range of applications in engineering computations. Applications of this approach to elasto-plasticity can be found in [16] and to reactive flow problems in [6].

Although the order of magnitude of the error can be predicted, accurate quantitative error estimation (or even asymptotic exactness) seems not be achievable by means of the cheap estimator used here. A more precise estimator can however be derived on the basis of the dual weighted approach.

In order to improve the efficiency of the numerical computations, it would also be of interest to derive stopping criteria for the different iterations implied to solve the discrete equations. This could also be done on the basis of the a posteriori estimator as proposed in [14] for control of the multi-grid iteration error for a model problem.

Finally, the presented approach can be immediately applied to non-stationary problems. Since the dual equations are backward in time, this affords a loop over the whole time interval. Therefore, the situation is similar to time-dependent optimal control problems and the design of an efficient algorithm is not evident.

References

[1] I. Babuska and A. Miller, The post processing approach in the finite element method, part 1 and 2, Internat. J. Numer. Methods Engrg., 34 (1996), pp. 1085–1109 and 1111–1129.

[2] R. Becker, An Adaptive Finite Element Method for the Incompressible Navier-Stokes Equation on Time-dependent Domains, dissertation, Heidelberg University, 1995.

[3] R. Becker and R. Rannacher, A feed-back approach to error control in finite element methods: Basic analysis and examples, *East-West J. Numer. Math.*, 4 (1996), pp. 237–264.

[4] ———, Weighted a posteriori error control in FE methods, in Proc. ENUMATH 1995, Paris Sept., 18-22, P. E. 1997, ed., Enumath, 1998.

[5] M. Braack, An Adaptive Finite Element Method for Reactive Flow Problems, dissertation, Heidelberg University, 1998.

[6] M. Braack, R. Becker, and R. Rannacher, Adaptive Finite Elements for Reactive Flows, in Enumath, Second European Conference on Numerical Mathematics and Advanced Applications, Enumath, 1997.

[7] S. C. Brenner and L. R. Scott, *The Mathematical Theory of Finite Element Methods*, Springer-Verlag, New York, 1994.

[8] W. Dörfler, A convergent adaptive algorithm for Poisson's equation, SIAM J. Num. Anal., 15 (1996), pp. 736–754.

[9] K. Eriksson and C. Johnson, An adaptive finite element method for linear elliptic problems, *Math. Comp.*, 50 (1988), pp. 361–383.

[10] T. H. F. Brezzi, L.P. Franca and A. Russo, Stabilization techniques and subgrid scales capturing, Proc. Conf. State of the Art in Num. An., York, April 1996, (to appear).

[11] L. Franca and S. Frey, Stabilized finite element methods: II. The incompressible Navier-Stokes equations, *Comp. Meth. Appl. Mech. Eng.*, 99 (1992), pp. 209–233.

[12] C. Johnson and P. Hansbo, Adaptive finite elements in computational mechanics, Comp. Meth. Appl. Mech. Eng., 101 (1992).

[13] M. L. M. Giles, M. Larson and E. Süli, Adaptive error control for finite element approximations of the lift and drag coefficients in viscous flow, SIAM J. Num. Anal., (to appear).

[14] C. J. R. Becker and R. Rannacher, Adaptive error control for multigrid finite element methods, *Computing*, 55 (1995), pp. 271–288.

[15] R. Rannacher, A posteriori error estimation in least-squares stabilized finite element schemes, in *Special Issue on Advances in Stabilised Meth. in Comp. Mec.*, L. P. Franca, ed., 1998. to appear.

[16] R. Rannacher and F.-T. Suttmeier, A posteriori error control and adaptive mesh adaptation for FE models in elasticity and elasto-plasticity, in Proc. Workshop On New Advances in Adaptive Comp. Meth. in Mech., Paris, Sept. 1997, Enumath, Elsevier, to appear.

[17] A. Russo, Bubble stabilization of finite element methods for the linearized Navier-Stokes equations, tech. rep. 930, Inst. di Analisi Numerica del CNR, Pavia, 1994, 1994.

[18] M. Schäfer and S. Turek, The benchmark problem "flow around a cylinder", sfb–preprint, IWR, Heidelberg University, May 1996.

[19] G. Shishkin, Grid approximation of singularly perturbed boundary value problems with convective terms, Sov. J. Numer. Anal. Math. Modelling, 5 (1990), pp. 173–187.

[20] R. Verfürth, *Review of A Posteriori Error Estimation and Adaptive Mesh- Refinement Techniques*, Wiley-Teubner, New York-Stuttgart, 1996.

[21] F. White, *Viscous Fluid Flow, second edition*, McGraw-Hill, New-York, 1991.

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - B.P. 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Lorraine : Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - B.P. 101 - 54602 Villers lès Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot St Martin (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, B.P. 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399